

# Scalable Target Coverage in Smart Camera Networks\*

Vikram P. Munishwar  
Computer Science  
State University of New York at Binghamton  
vmunish1@cs.binghamton.edu

Nael B. Abu-Ghazaleh<sup>†</sup>  
School of Computer Science  
Carnegie Mellon University, Qatar  
naelag@cmu.edu

## ABSTRACT

Smart camera networks are becoming increasingly popular in a number of application domains. In many applications, cameras are required to collaboratively track objects (e.g., habitat monitoring, or surveillance). In smart networks, camera coverage control is necessary to allow automatic tracking of targets without human intervention, allowing these systems to scale. In this paper, we consider the problem of automatic control of the cameras to maximize coverage of a set of targets. We formulate an optimization problem with the goal of maximizing the number of covered targets. Since the optimization problem is NP-hard, even for static targets, we propose a computationally efficient heuristic to reach near-optimal solution. Centralized solutions achieve excellent coverage, and can work well for small-scale networks, however they require significant communication cost for large scale networks. As a result, we propose an algorithm that spatially decomposes the network and computes optimal solutions for individual partitions. By decomposing the partitions in a way that minimizes dependencies between them, this approach results in coverage quality close to the centralized optimal solution, with an overhead and reaction time similar to those of distributed solutions.

## Categories and Subject Descriptors

G.1.6 [Numerical Analysis]: Optimization—*Integer programming*; C.2.1 [Computer-Communication Networks]: Network Architecture and Design—*wireless communication, centralized networks*

\*This research work was funded by Qatar National Research Fund (QNRF) under the National Priorities Research Program (NPRP) Grant No.:08-562-1-095

<sup>†</sup>This work was conducted while Dr. Abu-Ghazaleh was on a leave of absence from the State University of New York at Binghamton.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ICDSC 2010 August 31 – September 4, 2010, Atlanta, GA, USA  
Copyright 2010 ACM 978-1-4503-0317-0/10/08 ...\$10.00.

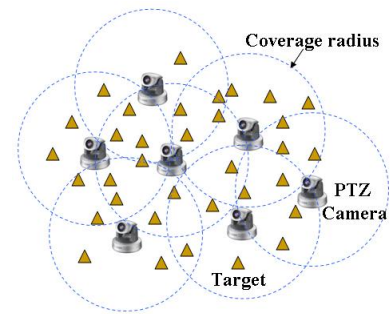


Figure 1: A general network of wireless PTZ cameras. The goal is to maximize the number of targets covered, where each camera can cover only a spherical sector of limited angle (field-of-view).

## General Terms

Algorithms, Design, Performance, Theory

## Keywords

hierarchical, coverage, optimization

## 1. INTRODUCTION

Recent technological developments in processing and imaging have created an opportunity for the development of smart camera nodes that can operate autonomously and collaboratively to meet an application's requirements. There is a wide variety of camera nodes available that can be used in camera networks deployments. The camera nodes in such networks can range from low-cost, low-resolution embedded camera sensors [10, 23, 24] to medium resolution webcams, to high-resolution, configurable PTZ (pan, tilt, and zoom) cameras.

Smart camera networks have a wide range of applications in areas such as security monitoring and surveillance, locating and tracking people, traffic management, health care and telemedicine [2]. Wireless camera networks can integrate with an existing fixed infrastructure to substantially improve the coverage and agility of these networks [6, 11]. Moreover, they can enable *ad hoc surveillance* where a group of wireless cameras are deployed in situations where infrastructure is unavailable or expensive, or quick deployment is desired. For instance, such networks can be used by emergency recovery teams to monitor a disaster area to guide the search and rescue operation [20]. They can also monitor industrial plants to quickly locate personnel in the event of accidents,

and to protect the plant from illegal intruders [29]. Camera networks also have applications in domains such as environmental monitoring and habitat monitoring for endangered animals [7, 14, 18].

The cameras must be controlled (pan, and if available, tilt and zoom) to provide the best possible coverage of events happening within the area of interest. As the scale of cameras grows from tens to hundreds of cameras, it is impractical to rely on humans to control their setting to achieve the best combined coverage. Thus, supporting autonomous configuration of cameras to maximize their collective coverage is a critical problem in smart camera networks. Figure 1 presents a general camera network scenario, with multiple PTZ cameras and a set of targets that need to be monitored. Specifically, the goal here is to maximize the number of targets covered by all cameras which can cover only a limited spherical sector (in 3-D) of the sphere of their potential coverage region at a given time. Each target may have different importance level. Moreover, the coverage for a given target may vary from different cameras depending on resolution, distance, and aspect of the target they capture.

We formulate the coverage problem in general; however, our evaluation assumes wireless camera networks to study some of the tradeoffs between delay, overhead and coverage quality in those networks. Although, a camera network would need to track multiple mobile targets in the presence of occlusions in a realistic scenario, we consider the problem of coverage of a static set of targets using a number of pre-deployed cameras as a first step in this direction. This represents the base case that we intend to later adapt to account for mobility and occlusions.

We first present an Integer Linear Programming (ILP) based formulation of the camera coverage optimization problem, which is based on the similar problem in the context of directional sensor networks [1]. This is an NP-hard problem; the decision version of this problem is based on the classical MAX-COVER [12] problem, which is NP-complete. Due to the high computational complexity of this problem for large-scale networks, we present a novel centralized force-based approach (CFA) which can compute close-to-optimal solution in polynomial time. We show that the solution obtained by using CFA is very close to the optimal compared with the existing heuristics (centralized as well as distributed) for maximizing static targets coverage.

The centralized approach to compute optimal solution is useful for small-scale networks, since the computation time and the network delay for collecting the current states of all cameras and targets at a central node, and disseminating optimal configurations back to all cameras is small. As the size of the network increases, the overhead of centralized approaches becomes prohibitive. On the other hand, as we validate in the evaluation section, the quality of the solutions obtained from purely distributed protocols is significantly worse than optimal. To address this problem, we propose to use a semi-centralized approach—a hierarchical solution—that attempts to combine the strengths of both approaches: the coverage quality obtained via centralized solutions and the low overhead and fast reaction time of distributed solutions. This intuition is borne by our simulation studies.

The main contributions of this paper are:

1. An Integer Linear Programming (ILP) based formulation of the camera coverage maximization problem, based on the similar problem in the context of direc-

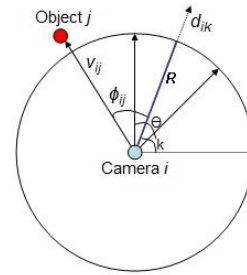


Figure 2: Camera coverage parameters

tional sensor networks [1].

2. A new centralized heuristic (CFA) to obtain a close-to-optimal solution for the camera coverage maximization problem in polynomial time.
3. A hierarchical approach to coverage maximization that addresses the scalability problem associated with centralized (optimal and CFA) solutions.

The remainder of the paper is organized as follows. Section 2 presents an Integer Linear Programming (ILP) formulation for solving coverage maximization problem for static targets, followed by the polynomial time, centralized force-directed approach (CFA) presented in Section 3. Section 4 presents the hierarchical approach for addressing the scalability problem. In Section 5, we present performance evaluation of centralized, distributed, and hierarchical approaches for maximizing the coverage gain. Section 6 presents the related work in this area. The concluding remarks and directions for future work are present in the Section 7.

## 2. BACKGROUND

We consider the general problem of maximizing target coverage by a pre-deployed camera network. Without loss of generality, we consider coverage as the total number of unique targets covered by the cameras. Thus, all targets are considered of equal importance and coverage of targets by cameras is boolean. In this section, we first present a brief overview of camera's field-of-view, and then give an overview of the ILP based problem formulation for maximizing the number of targets covered by cameras. For this work, we consider a camera node as a PTZ camera attached with an embedded board having sufficient processing capacity to carry out the simple processing required for the proposed algorithms.

### 2.1 Field-of-view of a camera

A field-of-view (FoV) of a camera represents the extent of its visible field. Typically, FoV is represented by its *angle-of-view* ( $\theta$ ) and *depth-of-field* ( $R$ ). Angle-of-view is represented in terms of the horizontal angle-of-view ( $\theta_h$ ) and vertical angle-of-view ( $\theta_v$ ). The depth-of-field represents the area of the visual scene that is acceptably sharp. Thus, for a given target, and a given level of acceptable sharpness, the depth-of-field of a camera spans from  $R_{min}$  to  $R_{max}$ . In order to simplify and abstract away the details, we consider the FoV of a camera as a spherical sector, which can be described by its angle-of-view ( $\theta$ ), and depth-of-field ( $R_{min}$  and  $R_{max}$ ).

## 2.2 ILP Problem Formulation

We make the following assumptions in the formulation. We assume that the positions of uncovered targets can be obtained through an external localization mechanism, such as using stereo cameras or a set of wide FoV cameras that can report approximate positions of targets. This assumption is consistent with those made by other works [17, 25]. PTZ cameras send their FoV and nearby targets' positions to a central node for computing optimal FoV parameters for all cameras. We consider that a target is either covered or not covered by a camera [19]; it should be possible to generalize to consider a more continuous measure of coverage in the future. Moreover, the model can generalize to consider occlusions as uncovered targets even when they are in the camera's FOV. For our experiments, we assume that the cameras are homogeneous but the formulation can naturally incorporate heterogeneous cameras. Thus, PTZ cameras can choose from a discrete set of pans (we do not consider tilt and zoom).

Consider a set of  $N$  static cameras trying to monitor  $M$  static targets (or objects) present in an area free of obstacles. Each camera is capable of panning to one of a discrete set of pans,  $P$ . The output of the optimization program is the list of  $camera \in N$ ,  $pan \in P$  pairs that collectively maximize the number of  $targets \in M$  covered.

The following additional variables are also used in the formulation as shown in the Figure 2:

- $\vec{v}_{ij}$ : Camera-Object vector from camera  $i$  to object  $j$
  - $\vec{d}_{ik}$ : Camera direction vector defining camera  $i$ 's orientation in direction (or pan)  $k$ .
  - $\phi_{ij}$ : The angle between  $\vec{v}_{ij}$  and  $\vec{d}_{ik}$ .
  - $\theta$ : Angle of field-of-view of camera, defining the visible area for the camera.
  - $R_{max}$ : Maximum length of the field-of-view of camera.
  - $R_{min}$ : Minimum length of the field-of-view of camera.
  - $P$ : A set of possible discrete pan values for a camera.
- The camera angle  $\phi_{ij}$  is calculated by using elementary trigonometry as:

$$\phi_{ij} = \cos^{-1} \left( \frac{\vec{d}_{ik} \cdot \vec{v}_{ij}}{|\vec{d}_{ik}| |\vec{v}_{ij}|} \right). \quad (1)$$

An object  $j$  is covered by camera  $i$  if

$$\phi_{ij} \leq \frac{\theta}{2}. \quad (2)$$

$$R_{min} \leq |\vec{v}_{ij}| \cos \phi_{ij} \leq R_{max}. \quad (3)$$

Using the above coverage test, each camera can generate its coverage matrix  $A_{N \times P}^M$  as:

$$a_{ik}^j = \begin{cases} 1 & \text{if camera } i \text{ with pan } k \text{ covers object } j. \\ 0 & \text{otherwise.} \end{cases}$$

The utility of each object  $j$  is assumed to be equal; the overall objective is to maximize the overall number of covered targets. A target covered by multiple cameras only counts as one target towards the overall objective.

$$\text{Maximize} \quad \sum_{j \in M} \gamma_j. \quad (4)$$

Where,  $\gamma_j$  is a binary variable taking value 1 when target  $j$  is covered by at least one camera, and 0 otherwise.

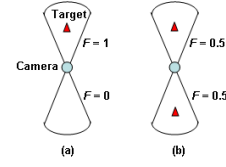


Figure 3: Illustration of Force-directed Approach

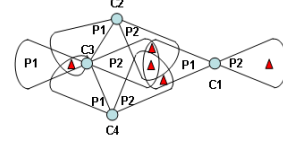


Figure 4: A counter example for CFA

The constraints of the problem can be represented as:

$$\sum_{k \in P} X_{ik} \leq 1 \quad \forall i \in N. \quad (5)$$

$$\frac{\sum_{i \in N, k \in P} a_{ik}^j X_{ik}}{L} \leq \gamma_j \leq \sum_{i \in N, k \in P} a_{ik}^j X_{ik} \quad \forall j \in M. \quad (6)$$

$$X \in \{0, 1\}, \gamma \in \{0, 1\}. \quad (7)$$

where  $X_{ik}$  is a binary variable which takes value 1 if pan  $k$  is selected for camera  $i$ ;  $L$  is an arbitrary large value ( $L \geq \|N\|$ ); and Equation 5 specifies the constraint that only a single pan can be assigned to a camera at a given time. Equation 6 allows the binary variable  $\gamma_j$  to assign itself a value 1 if target  $j$  is covered by at least one camera, and 0 otherwise.

The above problem is an Integer Linear Programming (ILP) problem. A similar problem formulation in the context of directional sensor networks has shown to be NP-hard [1]. Thus, in the following section, we present a polynomial-time, near-optimal, heuristic.

## 3. CENTRALIZED FORCE-DIRECTED ALGORITHM (CFA)

Since the optimal problem is NP-hard, we present a computationally-efficient heuristic that provides near-optimal camera configurations. We consider the objects covered in a particular pan as exerting a force proportional to the number of covered objects in that direction. More precisely, the force  $F_{ik}$  on camera  $i$  for selecting pan  $k$  is computed as:

$$F_{ik} = \frac{|M_{ik}|}{|M_i|} \quad i \in N, k \in P \quad (8)$$

where  $F_{ik}$  is a ratio of  $|M_{ik}|$ , the number of objects covered by camera  $i$  in pan  $k$ , and  $|M_i|$ , the number of objects that camera  $i$  can potentially cover (in all possible pans). Figure 3 presents an example of the computation of the force.

The proposed approach is a greedy based solution that iteratively selects camera-pan assignment pairs in decreasing order of their force magnitudes, as presented in Algorithm 1.

CFA has three loops: the outer while loop runs  $N$  times, and the inner loops run for each camera-pan pair (i.e.  $|N| \times$

---

**Algorithm 1:** Centralized Force-directed Approach

---

**input** :  $N = \{\text{set of cameras}\}$ ;  $M = \{\text{set of objects}\}$ ;  
 $P = \{\text{set of discrete pans}\}$   
**output**:  $Z$ ; Camera-pan pairs given by CFA.

```
1 while  $N \neq \{\}$  do
2   Compute  $F_{ik}$  for each  $i \in N$  and  $k \in P$ ;
3   Insert  $\{i, k\}$  with highest  $F_{ik}$  in  $Z$ ;
4   Remove selected camera from  $N$ ;
5   Remove selected targets from  $M$ ;
```

---

$|P|$  times). Thus, the time complexity of CFA is  $O(N^2P)$ . Given a large network where  $N$  is large, the complexity is  $O(N^2)$

While our experiments show that CFA gives near-optimal solutions in most cases, since it is a greedy algorithm, it can fail to produce optimal coverage. Figure 4 shows one of the cases where CFA finds a suboptimal configuration. Camera  $C1$  will first choose pan  $P1$  as it has the highest value of  $F_{1,1} = 0.75$ . Subsequently, for the remaining cameras, the value of force along pan  $P1$  becomes  $F_{2,1} = 1, F_{3,1} = 1, F_{4,1} = 1$ , while for pan  $P2$  of the remaining cameras, the force is zero. Thus, any camera can proceed given their equal force magnitude, and the algorithm will terminate since all cameras have either their pans assigned, or have zero force. This configuration only covers 3 targets, whereas the optimal configuration covers all 4 as can be seen in the figure. Nonetheless, in the experimental evaluation section we discover that in large random deployments of cameras and targets, CFA performs very close to optimal in most cases.

#### 4. ADDRESSING SCALABILITY PROBLEM: HIERARCHICAL APPROACH

The centralized approaches discussed so far work well for small sized networks, however they scale poorly for large-sized networks due to: (1) *Computational complexity*: The computational complexity of solving the ILP-based optimization problem increases exponentially w.r.t the number of cameras. (2) *Large response delay*: The response delay—the time difference between the request transmission by the first camera and the response reception by the last camera—is directly proportional to the number of cameras in the network.

To address these problems, we propose to use hierarchical mechanism for coverage maximization in large-scale networks. The intuition is to spatially decompose cameras within the network into small-sized partitions to take advantage of separation among cameras due to their geographical positions or presence of obstacles. The more separated the cameras are, the more independence they get in terms of coverage.

##### 4.1 Modified Single-Linkage Algorithm (SLA)

In order to create clusters within the network, we modify the existing, well known hierarchical clustering approach, Single-Linkage Clustering [27], where distance between two cluster is considered as the minimum distance between two points from the two clusters. The modifications to this algorithm are specific to the camera-coverage problem:

1. *Termination condition for cluster merging*: Single-linkage

clustering approach is a bottom-up clustering approach, which starts with each camera as a separate cluster, and merges clusters ‘close-enough’ with each other into a single cluster at different levels of the hierarchy. We employ a limit on the merging operation if the smallest distance between two clusters is greater than  $2 * R_{max}$ , where  $R_{max}$  is maximum sensing radius of a camera, since decisions of such clusters will be completely independent of each other.

2. *Cluster size ( $S_{max}$ )*: This is an important parameter, specific to the resource-constrained camera node, which may act as a cluster-head to compute optimal parameters for its cluster-members. Since, the number of pans a camera can take is constant, the time required to compute optimal result on a camera node (e.g. PTZ camera connected with a Soekris board [28]) is directly proportional to its cluster size. Thus, cluster-merging operation is terminated if the resultant size of the merged cluster will be greater than  $S_{max}$ .

The modified Single-Linkage Algorithm (SLA) is described in Algorithm 2. Essentially, during each iteration, the *while* loop increments  $d$  and tries to merge more clusters, such that the combined size of the clusters is less than  $S_{max}$ , and the minimum Euclidian distance between them is at most  $d$ . If at least one merged-cluster is created, a new level is added to the hierarchy. Once the clusters are formed, the cluster-member that is minimum hop away from the rest of the cluster members is chosen as the cluster head.

---

**Algorithm 2:** Modified Single Linkage Algorithm

---

**input** :  $A = \{\text{Adjacency matrix of Euclidian distances among cameras}\}$ ;  $S_{max} = \text{Maximum cluster size}$ ;  $R = \text{Coverage range of a camera}$   
**output**:  $Z$ ; Set of final clusters

```
1  $\ell = 0$ ; Current level of hierarchy;
2  $d = \text{Bound on distance between two cluster members}$ ;
3  $\Delta = \text{Distance increment}$ ;
4  $Clusters_\ell = \text{List of clusters. Each camera is an individual cluster initially}$ ;
5 while  $d \leq 2 * R$  do
6    $NewClusters = \{\}$ ;
7   foreach  $C_i \in Clusters_\ell$  do
8      $NewC = \{\}$ ;
9     foreach unused  $C_j \in Clusters_\ell$  do
10      if  $\|C_i\| + \|C_j\| \leq S_{max}$  &
11          $MinDist(C_i, C_j) \leq d$  then
12         $NewC = NewC \cup C_j$ ;
13        set  $C_j$  to be a used cluster;
14      if  $\|NewC\| > 0$  then
15         $add(NewClusters, NewC)$ ;
16      if  $\|NewClusters\| \neq \|Clusters_\ell\|$  then
17         $Clusters_{\ell+1} = NewClusters$ ;
18         $\ell ++$ ;
19       $d = d + \Delta$ ;
20  $Z = Clusters_\ell$ ;
```

---

Using the clusters provided by the modified SLA algorithm, optimal camera configurations can be computed for

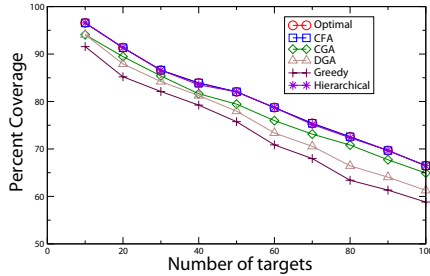


Figure 5: Study of varying number of targets.

individual clusters, instead of computing them in a global manner. Specifically, each cluster-member sends a request to its cluster-head, and the cluster-head computes and sends the optimal camera configurations back to the members.

Hierarchical approach may produce sub-optimal results when the cameras on the boundaries of different clusters end up covering the same targets due to the computations performed for individual clusters. However, such scenarios have appeared very infrequently in our experimental evaluation conducted by varying the density of targets and cameras. Thus, we do not incorporate a solution to handle this problem. However, if in certain scenarios, the coverage loss is significant, the boundary camera coverage problem can be solved by allowing border cameras that are covering redundant targets to collaborate with each other and choose different pans in a greedy manner.

## 5. EXPERIMENTAL EVALUATION

In this section, we present experimental evaluation for coverage maximization policies for static targets. We have simulated the proposed coverage approaches using QualNet simulator [26]. Nodes are categorized into camera nodes and targets, and are assigned respective functionalities. Each camera is set with a pre-defined maximum length of the FoV  $R_{max} = 100$ , minimum length of FoV  $R_{min} = 0$ , and FoV angle  $\theta = 45^\circ$ . The camera can choose from 8 discrete pan values as its orientation, however the proposed approaches can use higher number of pans, which could be overlapping. For evaluating the hierarchical approach, the maximum allowed cluster size is set to be  $S_{max} = 10$ , in order to get reasonable number of clusters. Each experiment was run for 500 seconds. We compare six different coverage policies representing centralized, distributed, and semi-centralized approaches. We first overview the existing heuristics that are used as benchmarks for our protocol evaluation, followed by the detailed evaluation of proposed approaches.

### 5.1 Existing Heuristics

Abouzeid et al [1] provide two greedy heuristic-based polynomial time algorithms to solve the similar problem in the context of directional sensor networks:

1. *Centralized Greedy Approach (CGA)*: CGA starts with making all cameras inactive, and selecting an inactive camera in each iteration that can cover maximum number of objects in a single direction.
2. *Distributed Greedy Approach (DGA)*: In DGA, all sensors assign themselves a unique priority value, and ini-

tialize their direction in a random manner. Each sensor detects the total number of targets it can cover in each direction, and selects the direction covering maximum targets. It then sends this direction information to its sensing neighbors (neighbors in  $2 \times$  sensing range radius). Overlaps in the target coverage are avoided by accepting the decision of higher priority sensor node. The priority assignment scheme of DGA ensures that the algorithm will terminate. However, the algorithm converges in iterative fashion, stabilizing from the highest priority node in the network.

*Greedy Approach*: We also use the typical greedy approach as our benchmark, where each camera autonomously assigns a pan that maximizes the number of targets covered from its potential targets.

### 5.2 Study of Varying Number of Targets

We have modeled the ILP problem formulation using AMPL-CPLEX [15]. We compare the proposed CFA approach with the optimal result obtained by solving the ILP, and also with the other centralized approach, CGA, distributed approaches DGA and Greedy, and the hierarchical approach.

Figure 5 shows a comparison of six policies, where *Optimal* policy is the answer computed by solving the ILP, while the *Hierarchical* policy results are obtained by solving ILP for individual clusters and noting the collective coverage gain. X-axis represents the varying number of targets used for the experiment, and the Y-axis represents *Percent Coverage*, which is calculated as:

$$PercentCoverage = \frac{|M_{covered}|}{|M_{potential}|} \quad (9)$$

where,  $|M_{covered}|$  are the number of targets covered by all cameras, while  $|M_{potential}|$  are the number of targets present in the sensing radius of all cameras. Thus, we exclude targets that are not placed in the potential coverage area of all cameras from the percent-coverage calculation. This experiment is run with 50 cameras deployed randomly on a 1000 sq. meter terrain. The objects are also deployed in a random fashion. The reason behind choosing the random deployment policy for cameras and targets was that the mixture of regions with dense and sparse deployment of cameras, which typically reflects the real-world settings, is possible with random deployment, compared with the uniform deployment. Essentially, in real-world settings, more cameras would need to be deployed at the places with large interesting activity, while a few cameras would suffice where the activity of interest is minimal.

We also test the proposed policies for scenarios where targets could be present in groups, by creating inhomogeneous node distributions for targets [3]. Figure 6 shows comparison of the coverage policies for covering clustered targets.

As it can be observed, the CFA not only outperforms the other heuristics, but also gives extremely close to optimal results. Moreover, the hierarchical policy also tracks the optimal very closely even when the terrain is densely populated with targets. However, since the CFA and the hierarchical policies are heuristics, they do not always behave optimally, as discussed in the respective sections. Although, it is possible to handcraft scenarios where hierarchical approach performs poorer than the optimal solution, such situations have rarely occurred in the randomly generated deployments of

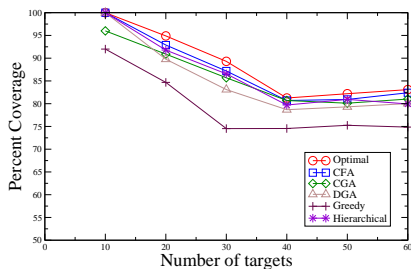


Figure 6: Study of varying number of clustered targets.

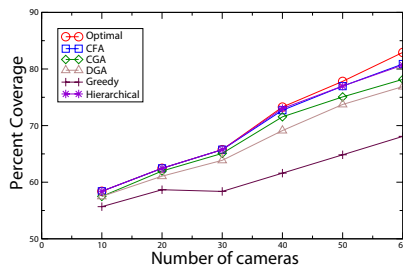


Figure 7: Study of varying number of cameras.

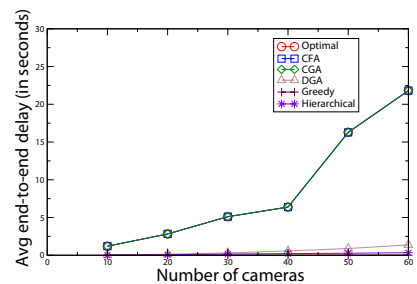


Figure 8: Study of overhead for varying number of cameras.

cameras and targets. Even when such scenarios are present, their effect is insignificant, when compared with the respective optimal solutions.

### 5.3 Study of Varying Number of Cameras

Figure 7 shows a comparison of the six policies for varying number of cameras. The number of targets were set to 100, and terrain dimension as 1000 sq. meters. As it can be noted from the graph, CFA and Hierarchical approaches outperform the other approaches and track the optimal values closely, if not overlapping. CFA and Hierarchical policies deviate slightly from the optimal for larger number of cameras deployed in a fixed sized terrain, in comparison with the smaller number of cameras. The reason behind this loss of coverage can be attributed to the existence of the counter-example case for the CFA, as described in Section 3, while the lack of coordination among cameras lying on the boundaries of different clusters, for the Hierarchical policy.

Figure 8 shows the overhead incurred by different policies in terms of the end-to-end delay. Specifically, for centralized approaches (Optimal, CFA, CGA), we define the end-to-end delay as the difference of time when the first camera sent its request to the base-station, and when the last camera received the optimal configuration from the base-station. For the DGA, end-to-end delay is the difference in time when the first camera sent the DGA packet to its neighbors, and when the last camera reconfigured itself without updating the neighbors. As it can be noted from the graph, overhead of centralized policies increases drastically as the number of cameras increases, while the increase in overhead for the distributed and semi-centralized approaches is minimal. Also, the Hierarchical approach outperforms DGA in terms of the end-to-end delay, even for networks with larger scale, since the DGA is inherently iterative in its approach. Note that the delay for centralized approaches is very high, especially for larger number of cameras. This is because the overall response time is severely affected even if the data for one camera is delayed from the base station, since the base station backs off multiplicatively on packet losses, based on the TCP's congestion control mechanism.

### 5.4 Study of Hierarchical Policy for Varying Camera Density

Figure 9 shows the comparison of the hierarchical policy with optimal in terms of the target-based coverage achieved. For this experiment, we have used 30 cameras, and 100 objects. We vary the terrain dimension from 200 sq. meters to

1000 sq. meters. Each cluster computes optimal value for its members by solving the ILP.

As it can be seen from the graph, the hierarchical policy gives close-to-optimal results. Both the approaches appear to give almost the same performance at extremes: as the terrain size increases, the cameras get placed more sparsely so that they have minimal dependence in terms of coverage on each other. On the other hand, for 200 sq. meters terrain scenario, cameras are densely packed within the small area, and thus end up covering most of the terrain area.

### 5.5 Study of the Impact of $S_{max}$

In Figure 10 and 11, we study the impact of varying the maximum allowed cluster size  $S_{max}$  on the perceived coverage gain and the overhead (in terms of end-to-end delay). For this experiment, we deployed 50 cameras and 100 targets in random fashion on a 500 sq. meter terrain. As it can be noted from the graphs, the difference between the coverage gain reduces as the value of  $S_{max}$  increases, reaching optimal when there is only a single cluster. In terms of the end-to-end delay based overhead, the hierarchical approach has an order of magnitude less overhead compared with the centralized optimal solution, when there are at least two clusters present. This represents the benefit of spatially decomposing the network. The overhead increases drastically when there is a single cluster, however it does not meet the overhead of centralized policies, since for hierarchical policy, the cluster-head itself is a camera, while for the centralized policies, the base-station is not. Thus, by carefully selecting the value of  $S_{max}$ , the hierarchical policy provides near-optimal coverage with the response time close to that of the distributed policies.

## 6. RELATED WORK

One of the classic coverage optimization problems, the Art Gallery Problem [31], focuses on placing minimum number of security guards in an art gallery so that all points in the whole gallery will be under observation. In the context of Directional Sensor Networks (DSNs), Cheng et al [5] have proposed Maximum Directional Area Coverage (MDAC) problem to maximize the total area covered by a DSN, while minimizing the number of active sensors. In another instance, Erdem et al. [9] consider the problem of determining automatic camera deployment layout to satisfy task-specific and floorplan-specific coverage requirements. Hoffmann et al [13] propose a distributed approach to maximize the total area coverage, where each camera tries to minimize cover-

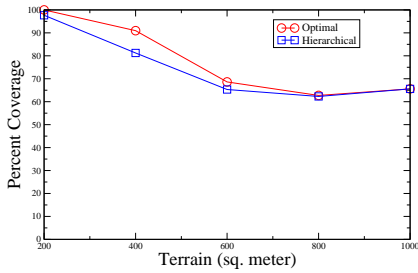


Figure 9: Hierarchical Vs Optimal for varying camera density.

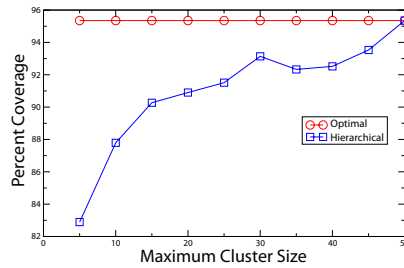


Figure 10: Study of  $S_{max}$  on the perceived coverage gain.

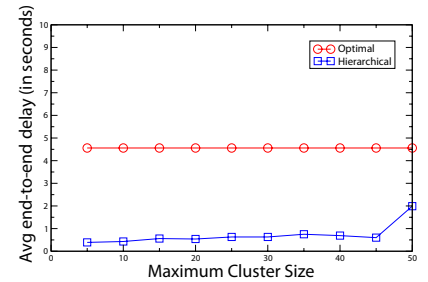


Figure 11: Study of  $S_{max}$  on the end-to-end delay.

age overlaps with its neighboring cameras. Although, these works address the problem of maximizing coverage by directional sensors, they focus mainly on maximizing the coverage of a given area, which is different from our goal of maximizing target-oriented coverage.

Significant research has been done on tracking a single or multiple targets [8, 21, 30]. However, this research is not targeted towards maximizing the number of targets under observation (the coverage maximization problem).

Some of the recent works in the context of *directional* sensor networks have provided optimization based solutions to address a basic instance of the problem of covering maximum targets [1, 4, 5]. Abouzeid et al [1] formulate the coverage optimization problem as a Maximum Coverage with Minimum Sensors (MCMS) problem, and show that MCMS is NP-complete. They formulate the MCMS problem as an Integer Linear Programming (ILP) problem to propose a centralized approach to solve the problem. However, since the centralized approach is not scalable, they provide greedy based centralized and distributed approaches for coverage maximization for DSNs, as described in Section 5.1.

Cai et al [4] focus on the problem of maximizing the lifetime of directional sensor network. Essentially, they compute Multiple Directional Cover Sets (MDCSs) for a given set of targets, where a Directional Cover Set (DCS) is a subset of directions of the sensors, which can cover all the targets present in the area under observation. Thus, by computing multiple such DCSs, they can alternately activate only one DCS at any given time, while keeping the other sensors in sleep state. This helps them increase the overall network lifetime. The authors have shown that the problem of computing DCS and MDCS is NP-complete.

Krahnstoeber et al [16], propose a solution for a typical surveillance class applications, where the goal is to cover each target for a fixed interval. Thus, this paper focuses on obtaining optimal scheduling pattern for covering targets, such that the total information gain from the overall coverage is maximum. Qureshi et al [22] present a planning strategy to achieve a close-up biometric coverage of selected pedestrians till they are present in the coverage region. These works differ from our work in terms of the objective: we focus on applications where it is important to track targets continuously as much as possible.

## 7. CONCLUSION AND FUTURE WORK

Smart camera networks are gaining increasing popularity due to their applicability in a number of real-world scenar-

ios. In smart camera networks, coverage control is necessary to allow automatic tracking of targets without human intervention, allowing these systems to scale. In this paper, we consider the problem of automatic control of the cameras to maximize coverage of a set of targets. To address this problem, we first present an ILP based formulation of the coverage optimization problem, which is based on the similar work in the context of directional sensor networks. Since, the ILP problem is NP-hard, we proposed a novel polynomial time force-directed approach (CFA), which performs very close to the optimal as shown in the experimental evaluation section. Since, the purely centralized solutions are not scalable, we proposed a hierarchical approach that spatially decomposes the network and computes optimal solutions for individual partitions. We validated experimentally that the hierarchical policy gives close to the optimal results, while maintaining the overhead close to the distributed approaches with careful selection of parameters (especially  $S_{max}$ ).

In future, we plan to extend the proposed solutions for ensuring maximum coverage in case of mobile targets, and study the effectiveness of centralized, distributed, and semi-centralized approaches for different target-mobility scenarios in simulations as well as on a real testbed. Next, we would like to focus on maximizing coverage when static/mobile obstacles are present in the deployment area.

## 8. REFERENCES

- [1] J. Ai and A. A. Abouzeid. Coverage by directional sensors in randomly deployed wireless sensor networks. *Journal of Combinatorial Optimization*, 11:21–41, 2006.
- [2] I. Akyildiz, T. Melodia, and K. Chowdhury. A survey on wireless multimedia sensor networks. *Computer Networks*, 51(4):921–960, 2007.
- [3] C. Bettstetter, M. Gyarmati, and U. Schilcher. An inhomogeneous spatial node distribution and its stochastic properties. In *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*, page 404. ACM, 2007.
- [4] Y. Cai, W. Lou, M. Li, and X. Li. Target-Oriented Scheduling in Directional Sensor Networks. *IEEE Infocom*, 2007.
- [5] W. Cheng, S. Li, X. Liao, S. Changxiang, and H. Chen. Maximal Coverage Scheduling in Randomly Deployed Directional Sensor Networks. In *Parallel Processing Workshops, 2007. ICPPW 2007. International Conference on*, pages 68–68, 2007.

- [6] R. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, et al. *A System for Video Surveillance and Monitoring*. Carnegie Mellon University, the Robotics Institute, 2000.
- [7] N. Cruz, A. Matos, J. de Sousa, F. Pereira, J. Silva, E. Silva, J. Coimbra, and E. Dias. Operations with multiple autonomous underwater vehicles: the PISCIS project. In *Second Annual Symposium on Autonomous Intelligent Networks and Systems AINS*, 2003.
- [8] A. Ercan, A. El Gamal, and L. Guibas. Object tracking in the presence of occlusions via a camera network. *Proceedings of the 6th international conference on Information processing in sensor networks*, pages 509–518, 2007.
- [9] U. Erdem and S. Sclaroff. Automated camera layout to satisfy task-specific and floor plan-specific coverage requirements. *Computer Vision and Image Understanding*, 103(3):156–169, 2006.
- [10] E. Fossum and L. Photobit. CMOS image sensors: Electronic camera-on-a-chip. *IEEE transactions on electron devices*, 44(10):1689–1698, 1997.
- [11] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, S. Pankanti, I. Center, and N. Hawthorne. Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking. *Signal Processing Magazine, IEEE*, 22(2):38–51, 2005.
- [12] D. Hochbaum. Approximating covering and packing problems: set cover, vertex cover, independent set, and related problems. In *Approximation algorithms for NP-hard problems*. PWS Publishing Co., 1996.
- [13] M. Hoffmann, M. Wittke, J. Hahner, and C. Muller-Schloer. Spatial Partitioning in Self-Organizing Smart Camera Systems. *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, issue 4, pp. 480–492, 2:480–492, Aug. 2008.
- [14] R. Holman, J. Stanley, and T. Ozkan-Haller. Applying video sensor networks to nearshore environment monitoring. *IEEE Pervasive Computing*, 2(4):14–21, 2003.
- [15] I. ILOG. ILOG AMPL CPLEX System. ILOG. Inc., *Incline Village, NV*, 2001.
- [16] N. Krahnstoeber, T. Yu, S. Lim, K. Patwardhan, and P. Tu. Collaborative Real-Time Control of Active Cameras in Large Scale Surveillance Systems. In *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications*, 2008.
- [17] P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu. SensEye: a multi-tier camera sensor network. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 229–238. ACM New York, NY, USA, 2005.
- [18] T. Little, J. Konrad, and P. Ishwar. A wireless video sensor network for autonomous coastal sensing. *Proceedings of Coastal Environmental Sensing Networks 2007*, 2007.
- [19] C. Papageorgiou, M. Oren, and T. Poggio. A General Framework for Object Detection. In *Proceedings of the Sixth International Conference on Computer Vision*, page 555. IEEE Computer Society, 1998.
- [20] L. Paxton and J. Yee. The role of emerging technologies in imagery for disaster monitoring and disaster relief assistance. *Acta Astronautica*, 52(9-12):793–802, 2003.
- [21] F. Porikli and A. Divakaran. Multi-camera calibration, object tracking and query generation. In *Multimedia and Expo, 2003. ICME'03. Proceedings. 2003 International Conference on*, volume 1, 2003.
- [22] F. Qureshi and D. Terzopoulos. Planning Ahead for PTZ Camera Assignment and Handoff. In *ACM/IEEE International Conference on Distributed Smart Cameras*, pages 1–8, 2009.
- [23] M. Rahimi, R. Baer, O. Iroezzi, J. Garcia, J. Warrior, and M. Srivastava. Cyclops: in situ image sensing and interpretation in wireless sensor networks. In *Proceedings of the 3rd international conference on Embedded networked sensor systems*, pages 192–204. ACM New York, NY, USA, 2005.
- [24] A. Rowe, A. Goode, D. Goel, and I. Nourbakhsh. CMUcam3: an open programmable embedded vision sensor. See also [www.cmucam.org](http://www.cmucam.org). *CMU-RI-TR-07-13, Robotics Institute, Carnegie Mellon University*, 2007.
- [25] D. Schleicher, L. M. Bergasa, R. Barea, E. Lopez, and M. Ocana. Real-time simultaneous localization and mapping using a wide-angle stereo camera. In *DIS '06: Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications*, pages 55–60, Washington, DC, USA, 2006. IEEE Computer Society.
- [26] Q. Simulator. Scalable Network Technologies. Inc.[Online]. Available: [www.qualnet.com](http://www.qualnet.com).
- [27] Yooreeka: A library for data mining, machine learning, soft computing, and mathematical analysis. Available at <http://code.google.com/p/yooreeka/>.
- [28] Soekris engineering net5501. Available at <http://www.soekris.com/net5501.htm>.
- [29] T. Teixeira and A. Savvides. Lightweight people counting and localizing in indoor spaces using camera sensor nodes. In *Distributed Smart Cameras, 2007. ICDCS'07. First ACM/IEEE International Conference on*, pages 36–43, 2007.
- [30] N. Ukita, T. Nagao, and T. Matsuyama. Versatile Cooperative Multiple-Object Tracking by Active Vision Agents. In *MVA2000, IAPR Workshop on Machine Vision Applications*, 2000.
- [31] J. Urrutia. Art gallery and illumination problems. *Handbook of Computational Geometry*, pages 973–1027, 2000.