A PSO-based Algorithm for Video Networks Planning Optimization

Jiang Peng
College of Electrical Engineering
Southwest Jiaotong University
Sichuan Chengdu

Jin WeiDong
College of Electrical Engineering
Southwest Jiaotong University
Sichuan Chengdu

Abstract— **In this paper we examine issues of deploying a camera network in a complex environment with obstacles. A camera network is composed of a distributed collection of cameras, each of which has sensing and communicating capabilities. To deploy such camera network, we present a kinetics based particle swarm optimization (PSO) approach. By introducing a kinetics-constraint factor to standard PSO, the fields are covered such that each camera is repelled by both other cameras and obstacles, thereby forcing the network to spread throughout the monitored area. The coverage enhancement is fulfilled by finding an optimal orientation for each camera, guided by PSO optimizer. Experimental results show our method is able to achieve higher coverage rate than conventional methods**.

Keywords- PSO, Potential Fields, Camera Network, Deployment.

I. INTRODUCTION

In recent years, there is increasing demand of large environments applications such as distributed video surveillance which multiple cameras' tracking correspondence and collaborative activity/events recognition are required [1,2,3] . These multiple cameras are usually connected together to form a visual sensor network to support the wide area of monitor.

An important quantifiable property of camera networks is the coverage: what visual data the camera network is physically capable of collecting. In other words, the physical area or volume of the scene has been covered by a camera network. To cover larger area, it usually requires adding more cameras. However, using an increasing number of cameras is impractical due to concerns over increased requirements in terms of computation and monetary cost, bandwidth and storage.

Coverage problems in WSN(Wireless Sensor Network) can be modeled through the well-known art gallery problem [4]. Once the FOV of the camera sensors are known, art gallery problem can be used to determine the least number of nodes and their locations in order to provide full coverage of a monitored region. It has been shown in [4] that the problem can be solved in polynomial time in two dimensional environments. However, the solution for art gallery problem cannot be used in our context as we assume a randomly deployed camera network. Optimal node placement is a very challenging problem that has been proven to be NP-hard for most of the formulations of sensor deployment. To solve these NP hard issues, it usually requests some approximate solutions.

Furthermore, in contrast to traditional sensor networks which assume omnidirectional sensors, cameras networks have its unique properties which introduce additional complexity to the sensor placement problem. In networks comprised of pan-tilt-zoom(PTZ) cameras, the covered area can be actively controlled by changing the cameras' parameters. This allows the overage area vary with this dynamic property. What is more, these cameras are usually required to install in complex scene with obstacles like walls or trees. And after installation, these cameras are usually position fix. With these properties, the cameras network cannot directly employ the most WSN deployment approaches.

Recently, some works [5-6] address the issue of PTZ configurations for coverage optimization based on expectation-maximization, or game theoretic approaches.

Bernhard et al[7] proposed a resource aware camera network coverage enhancement method which employs generic based evolutionary algorithm. In their work, the frame rate and resolution are considered as a constriction for interesting points' coverage. Jiang[8] etc proposed a potential field based algorithm to optimize the layout of camera network.

Recently, evolution algorithms gain popularity especially when apply to sensors' planning because they outperform many conventional sensors planning's approaches. Work[9] studies evolution algorithms such as the gene, ant colony and particle swarm optimization (PSO) and applies them in the camera network coverage problem. Their study indicates that PSO gains the best performance among all evolution algorithms when applying to the camera network coverage. Xu et al[10] also develop a camera coverage maximization approach which directly employs standard PSO. In their performance evaluations, PSO outperforms the conventional sensor planning methods. From this, we can see PSO is a prominent algorithm when applies to coverage optimization.

PSO uses maximized coverage to guide each camera to adjust to individuals best FOV. However, PSO based methods have some disadvantages. For example, as lack of inter-cameras communication, the convergence procedure is long and requires each camera to try many times to find the individual and global optimized FOV. It will lead to a pretty long time for the PSO based approaches to convergence. To overcome this issue, we propose a kinetics-constraint factor to the standard PSO algorithm, and the fields are covered such that each camera is repelled by both other cameras and obstacles, thereby forcing the network to spread throughout the monitored area. In this way, the PSO is guided and therefore speeds up to convergence.

II. MODELLING THE VIDEO NETWORK COVERAGE PROBLEM

Usually, a PTZ camera's parameters can be defined as pan, tilt angle and zoom distance. In our application, the 3D visual is casted to 2D plane and the maximum zoom is limited to a given distance by considering the minimal resolution required. Thus, the camera's FOV is modeled as a simple 2D disc. Fig 1 illustrates a simple camera network consisted by two cameras. The physical volume covered by a camera is FOV and the FOV is a planar 2D sector shape structure. The viewpoint of each camera is γ and maximized imaging distance is R. The orientation of camera i is defined as η_i .

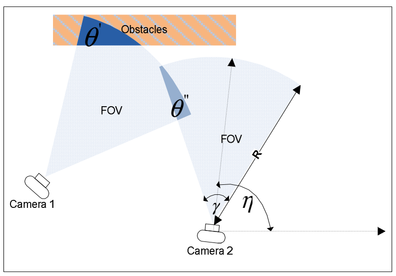

Figure 1 two cameras and one obstacle's camera network

Comparing with traditional sensor network, the camera sensor network has below two unique characteristics:

1) There are a large number of cameras in the camera network and they are mounted at random locations. The original positions are fixed but each camera is PTZ type and able to adjust the orientation freely $[0-2\pi)$.
2) The camera's FOV is a sector shape and camera can only sense the targets within the FOV. Thus camera is not a conventional omnidirectional sensor.

Therefore, for each camera i can be modeled as position and orientation on the plane $[x_i, y_i, \eta_i]$ with the perpendicular orientation.

To align with PSO, assume there are N particles, and their current positions are $X(x_1, x_2, ..., x_n)$ and the velocities are defined as $V(v_1, v_2, ..., v_n)$. The best previous position encountered by particle (i) is denoted as:

$$P_i = \begin{cases} P_i(t), f(X(t+1)) > f(X(t)) \\ p_i(t+1), f(X(t+1)) < f(X(t)) \end{cases} \quad (1)$$

where $f(X(t))$ is the optimized goal for the whole smarm best positions are denoted as $P_g(t)$ defined as below:

$$P_g(t) = Min(f(x1), f(x2),..., f(xn)) \quad (2)$$

From time t to time t+1, the particles' positions and velocities are defined as:

$$\begin{aligned} &V_i(t+1) = \delta(t)\cdot V_i(t) + C_1\cdot r_1(t)\cdot(P_i(t) - X_i(t)) + C_2\cdot r_2(t)\cdot(P_g(t) - X_i(t)) \\ &X_i(t+1) = X_i(t) + V_i(t+1) \end{aligned} \quad (3)$$

where $\delta(t)$ is the constriction coefficient, used to determine the convergence speed; C_1 and C_2 are the cognitive and social parameter, respectively, to decide local and global optimization weights; and $r_1(t)$, $r_2(t)$, are random variables uniformly distributed in [0, 1], using for the evolution purpose.

If each camera is treated as a particle, then the coverage rate of monitor area is used as PSO smarm best positions and the maximized coverage area of each camera's orientation to achieve maximized non-overlapped FOV is used as PSO individual best position. The PSO can apply on the camera coverage development problem.

However, the cameras are not 'free particle'. They must follow kinematic constraints in the context of application which will be addressed in below section.

III. KINETICS AND THE CONTROL LAW

According to the properties of fixed position and semi-directional sensing, cameras have both kinematic and dynamic constraints. They are only able to adjust the orientations to enlarge the view point. In this section, we examine the control law for individual camera and how cameras interact with each other.

Potential fields are a commonly used and well understood method in mobile robotics, where they are typically applied to tasks such as local navigation and obstacle avoidance.

For camera development, we need redefine the motion and control law which fit for this semi-directional sensor. We divide the potential field into two components: force due to obstacle and force due to other cameras. To illustrate control law for example we show in Fig 2.

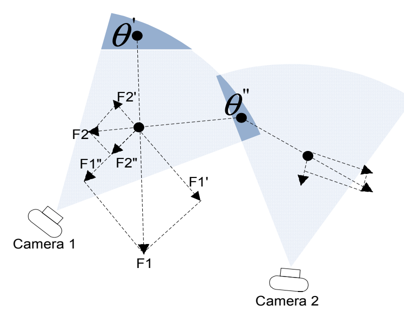

Figure 2 control law demonstration for two cameras' case

Taking fig 2 for example, obstacle repels camera 1 by force $F1$, while $F1$ decomposes into two vectors named $F1''$ and $F1'$, representing the parallel and orthogonal force to camera1's orientation. Overlapped area θ'' with camera2 also gives a repulsive force $F2$ to camera1, decomposing into two forces as $F2^{'}$ and $F2^{''}$, similar with $F1$.

If a camera has received multiple K forces, the total forces are summary of each force, denoted as: $F = \sum_{i=1}^{K} F_i$.

The parallel and orthogonal forces are donated as $F^{''} = \sum_{i=1}^{K} F_i^{''}$ and $F' = \sum_{i=1}^{K} F_i^{'}$ respectively.

According to control law, all cameras are fixed. The all forces parallel to camera's orientation have been eliminated by the kinematic constraints. Thus, we can ignore $F^{''}$. In contrast, the orthogonal force F' is key force to make camera adjust angle to enlarge coverage. In Fig 2, $F1^{'}$ is stronger than $F2^{'}$. As a result, camera1 tends to rotate to clockwise to avoid the repulsive force.

The strength of each virtual force is determined by size of overlapping area. The larger of the overlapping area, the stronger repulsive force a camera will receive from this area and vice versa. Thus, we define the strength of virtual force F_i^{stren} as equation:

$$F_i^{stren} = \lambda \cdot \gamma \cdot \frac{\sum \theta_{ij}^{'} + \sum \theta_{ji}^{''}}{R^2 \cdot \pi} \tag{4}$$

where λ is const parameter used to adjust with the size of FOV. If a camera receives virtual force strength is smaller than a const parameter ζ, then we treat this camera as standstill.

The strength of virtual force of individual camera can be treated as criterion of whether a camera has reached an optimized orientation with their neighbors. The individual strength could not be virtual zero while the orthogonal forces should reach a virtual zero which means it gains balances among its neighbors.

IV. VIRTUAL FORCE GUIDED PSO FOR VIDEO NETWORK DEPLOYMENT

With this new criterion, PSO needs to be modified and a new guide factor has been added to standard PSO. The equation 4 is modified as below:

$$\begin{aligned} V_i(t+1) &= \psi(t) \cdot F_i^{stren} + (\delta(t) \cdot V_i(t) + C_1 \cdot r_1(t) \cdot (P_i(t) - X_i(t)) + C_2 \cdot r_2(t) \cdot (P_g(t) - X_i(t))) \\ X_i(t+1) &= X_i(t) + V_i(t+1) \end{aligned} \tag{5}$$

where $\psi(t) \cdot F'^{stren}_i$ is the new added guide factor to PSO which depends on the orthogonal force and a time varying factor $\psi(t)$ which defined as below:

$$\psi(t) = 1 - \frac{t}{MAXSTEP} \tag{6}$$

The factor $\psi(t)$ is defined as descending with time which controls the weight of guide factor. With the time descending guide factor, in the begging of PSO iteration, the guide factor acts as main contributor to spread out the camera network and in the end phase of optimization, the PSO takes advantage to global optimization goal to adjust cameras' orientations until meeting convergence criteria.

In standard PSO, the camera will rotate the orientation randomly to seek the best position. The seeking optimization progress depends on the randomly behaviors which is not straightforward and cost lots of time.

After applying the new criterion, a camera has capability to know the neighbors and obstacles' position information and tends to avoid the overcalling when each time to adjust the orientation. The adjustment of orientation of camera is purposive and can avoid the problem that time costly optimization seeking. The algorithm is described as below.

Pseudo-code of the proposed PSO based approach

Input: N (number of cameras), W(monitor area)

Step 1. Randomly generated cameras in various positions and orientations, obstacles in various positions and shapes.

Step 2. While the predefined criteria does not match

Step 3. For each camera $i = 1$ to N

Step 3.1. $[x_i, y_i, \eta_i]$ probe its neighbors and obstacles and get the neighbors and obstacles collections S_i.

Step 3.2. For each overlapping area belong to S_i

Step 3.2.1 Measure the overlapping area and get the centroid of each overlapping area $\theta^{''}$ or $\theta^{'}$.

Step 3.2.2. if $\theta^{''}$ or $\theta^{'}$ is not empty

Step 3.2.3. Measure the repel force F1 and F2 and virtual force F

Step 3.2.4. Determine the rotate direction

Step 3.3. calculate F_i^{stren} as equation 6

Step 3.4. update the orientation with equation (8)

Step 3.5. Measure camera i coverage P_i, if $P_i > P_{i-1}$ then update η_i as η_p

Step 3.6. Measure monitor area coverage rate C_i If $C_i > C_{i-1}$ then update $[\eta_1, \eta_2, \eta_i]$ as the $[\eta_1, \eta_2, \eta_i]_g$

Step 4. End for

Step 5. End while

Step 6. Using the global best orientations to set each camera' orientation and output the coverage.

V. EXPERIMENTS RESULT

We have conducted a series of simulation experiments aimed at both validating and investigating the use of our algorithm for the camera network deployment problem. Two metrics are of particular interest: coverage(what the monitor area covered by the cameras network) and time(how long does the network take to deploy).

Our experiments were conducted in simulation software which derives from OpenCV lib on a PC with 1.7G CPU clock and 2G ram. In this scene, the simulated monitoring area is a 800X600m rectangle area and there are 100 cameras deployed randomly in this area. All cameras are same type of PTZ and have semi-directional FOV with view angle γ of 0.3π. The cameras' initialized orientations are randomly set and the maximized distance sensed is 80m. The PSO parameters are set with same setting of work[10] as c1=0.729,C2=1.49445.

We calculate the idea maximum coverage by assuming that there is no overlap among any cameras or obstacles in a camera network. The idea maximum coverage $C_{\max}$ in this scenario is: $C_{\max} = N\frac{\gamma}{2\pi}\cdot\pi\cdot R^2$.

In our setting, 69.8% is the maximum coverage that 100 cameras can cover this monitored area without any overlapping.

According to this setting, we implement the potential field based camera development (PFOFSA) in work[8] and latest PSO based camera development (STDPSO) in work [10] and we conduct the experimental comparison with these closed works. As STDPSO will take quite long iterations to reach, we set the all iteration step as 100 for all tests.

Firstly, we have randomly deployed 5 various shape obstacles in the scene, including 2 circles and 3 rectangles. The cameras will be installed to avoid the obstacle, but the FOV can overlap with these obstacles. We verify our method and compare with STDPSO and PFOFSA in this complex scene.

Fig 3 shows deployment results for all four methods when there are 5 obstacles existed in same scene. The initial deployment coverage is 51.1% due to the obstacles. The STDPSO has achieved 49.4% and 57.2% coverage. After 60 iterations of PSO, our proposed method achieves 59.2% coverage, outperforming the STDPSO and PFOFSA 2% and 1.8% on coverage perspective in 100 times iterations. We also notice that with the more obstacles adding into the scene, the coverage improvement is slow down. That is because the coverage cannot be liner increasing due to capability of the given scene.

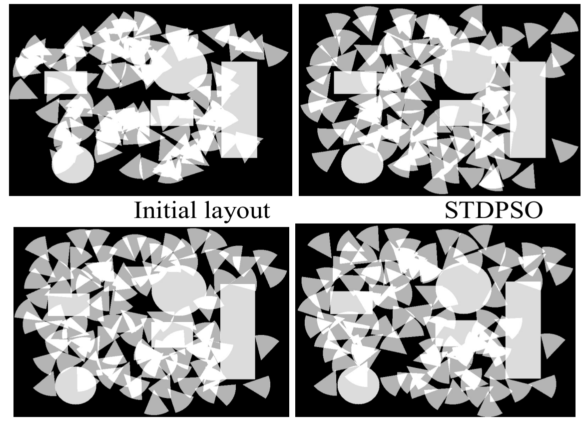

Figure 3 Experiment Result Comparison (with 5 obstacles)

To measure the result quantitatively, we test each method 100 times and get the average coverage rate and list the result in table 1. From table 1, we can see when 100 cameras are randomly deployed in the monitoring area, the coverage are averagely 42.5% and 51.1% with without obstacles and with 5 obstacles respectively. Our approach outperforms STDPSO and PFOFSA regardless obstacles existed in scene or not.

TABLE I

100 TIMES AVERAGE QUANTITATIVE RESULTS COMPARISON (100 ITERATIONS)

method	Without obstacle	With 5 obstacles
Random	42.5%	51.1%
STDPSO	49.4%	57.2%
PFOFSA	52.9%	57.4%
Our approach	54.6%	59.2%

In the following experiment, we aim to valid the deployment time with coverage rate.

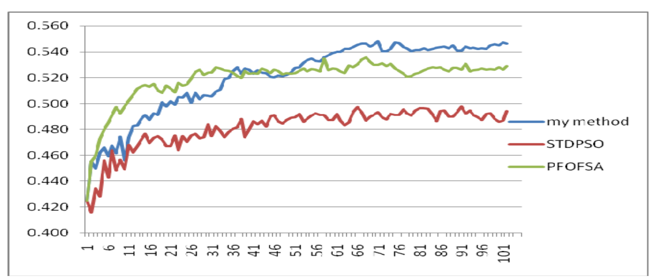

Figure 4, coverage plot with iterations

Fig 4 shows a plot of coverage versus deployment iteration. From this plot, it is apparent that the rate of coverage increases with iteration, and that the total coverage remained stable when the experiment was terminated, after 100 seconds. The potential field based PFOFSA has fastest deployment speed, not surprisingly. It is because this method is a local optimization method which considers a node and its neighbors only. However, this non-global optimized approach is easy to trap local optimization and prevent it achieving higher coverage. Thus, PFOFSA remains about 52% and stops. In contrast, PSO based methods such as STDPSO and our approach have a steady optimization trend. However, STDPSO coverage fluctuates between 0.48 and 0.49 and increasing very slowly. That is PSO usually requires quite longer time to achieve the same optimization goal because each time the camera adjusts orientation is tentative and random. As the guide factor added in PSO, our method has determined directions of adjusting orientation and with the time the guide factor will fade the impacts, thus it adjusts the initial layout and continue global optimize with PSO.

The rest experiments target to exam the relationship of coverage and the critical parameters used in our approach. Number of camera, the maximized sense length R, and initial layout are investigated for the impact to coverage.

Firstly, we address the impacts of number of camera to the coverage. Obviously, the more cameras will lead to higher coverage while we examine the coverage improvement by adding more cameras in the monitoring area.

Fig 5 illustrates the coverage with the number of cameras. From 60 to 200 cameras, we add 20 cameras each time and measure the coverage of random deployment and after applying our approach. Fig 5 indicates both random deployment and our approach have almost liner coverage increase with the cameras' number. It also implies that our approach works well both in sparse and dense camera scene.

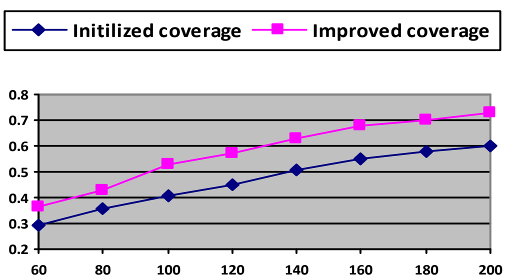

Figure 5, coverage results on different number of cameras

Furthermore, the depth of shooting is a key parameter. A constraint on the minimum required resolution translates directly into an upper limit on depth. In our simulation, we adjust the radius of sector to simulate the impacts of upper limit of depth. From 40 to 140, the radius has been updated by 20 steps. From fig 6, we can see with very short depth, the camera network almost has no overlapping. As a result, no improvement is made from the randomly deployment. However, with the increase of depth, the overlapping area increases also, and our approach is able to repel camera each other and lead to a steady increase of coverage improvement.

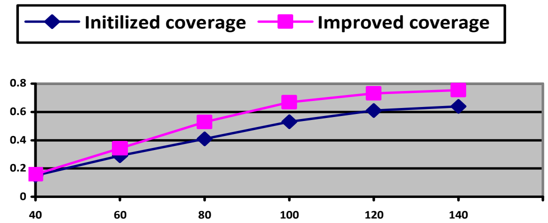

Figure 6, coverage results on different depth of shooting

VI. CONCLUSION

In this paper we present a formulation and an approximation method for the camera network coverage optimization. We consider the property of PTZ cameras and the neighboring relationship among cameras. With a modified PSO based evolution algorithm, we demonstrate that the feasible and near-optimal solutions can be found for a large number of cameras in a complex scene scenario in short time.

I. ACKNOWLEDGMENT

This research is supported by the Fundamental Research Funds for the Central Universities (SWJTU12CX027) and Natural Science Foundation of China(60971103,61134002).

REFERENCES

[1] S. Soro and W. Heinzelman, “A Survey of Visual Sensor Networks,” Advances in Multimedia, pp. 1–21, 2009.

[2] Aaron Mavrinac, Xiang Chen, “Modeling Coverage in Camera Networks: A Survey”, International Journal of Computer Vision,2012, November

[3] Bir Bhanu, Chinya V. Ravishankar, Distributed Video Sensor Networks [M] Springer-Verlag,2011.

[4] D Lee, A Lin, Computational complexity of art gallery problems[J] IEEE Transactions on Information Theory,1986,32(2): 276 - 282

[5] Bi Song ; Morye, A. Collaborative Sensing in a Distributed PTZ Camera Network[J] IEEE Transactions on Image Processing,2012,21(7):3282-3295

[6] Lobaton, E., A Distributed Topological Camera Network Representation for Tracking Applications[J] IEEE Transactions on Image Processing, 2010,19(10):2516-2529

[7] B. Dieber, C. Micheloni, and B. Rinner. “Resource-Aware Coverage and Task Assignment in Visual Sensor Networks”. IEEE Transactions on Circuits and Systems for Video Technology,21:1424–1437, 2011.

[8] Yibo, J, WangLiang W, WeiJie,C, Coverage Optimization of Occlusion-Free Surveillance for Video Sensor Networks [J]Journal of software，2012,23(2):310-322

[9] Aouf, N. ; Djouadi, M.S. ; Particle Swarm Optimization Inspired Probability Algorithm for Optimal Camera Network Placement[J]IEEE Sensors Journal,2012,12(5):1402-1412

[10] Yi-Chun Xu,Bangjun Lei,Emile A. Hendriks,Camera Network Coverage Improving by Particle Swarm Optimization[J]EURASIP Journal on Image and Video Processing,2011,1(1):3

[11] J. Zou, C. Ta, R. Zhang, and H. Xiong, “Modeling and Optimization of Network Lifetime in Wireless Video Sensor Networks,” in Proceedings of the IEEE International Communications Conference, 2010, pp. 1–6.

[12] C. Yu and G. Sharma, “Camera Scheduling and Energy Allocation for Lifetime Maximization in User-Centric Visual Sensor Networks,” IEEE Transactions on Image Processing, vol. 19, no. 8, pp. 2042–2055, 2010