Non Destructive Testing based on a Scanning-From-Heating approach: Application to non-through defect detection and fiber orientation assessment

Mohamed Belkacemi, Christophe Stolz, Alexandre Mathieu, Guillaume Lemaître,
Joan Massich, Olivier Aubreton
Université de Bourgogne Franche-Comté
12 rue de la Fonderie, 71200 Le Creusot

Abstract

A fundamental issue in video surveillance is the coverage problem of the monitored area with respect to the positioning of the camera network. This is a tricky optimization problem which depends on many parameters and which is underconstrained in the general case leading to sub-optimal solutions. In this paper, we propose to adapt a Particle Swarm Optimization algorithm to solve it and also to analyze in depth its performances to assess is employability and efficiency. Experiments are performed in simulation in order to provide quantitative results.

1. Introduction

The problem of the optimal positioning of a camera network is complex and no efficient solution exits to date to solve it. The aim of this work is to provide a flexible and tunable solution to this problem and, doing so, to analyze the performances of the current state-of-the-art techniques. The final objective of our research is to design a global optimization scheme allowing a camera network to self-organize and self-reconfigure, according to priorily fixed constraints, in order to ensure a full coverage of a given scene. The scheme must suit with any kind of cameras, such as perspective, fish-eye, catadioptric and, possibly RGB-D and ToF. As a perspective, self-organization and self-reconfiguration should be performed in real time. Within this context, it is important to assess the actual performances and limits of the state-of-the-art algorithms. First step, which will be detailed in this paper, is to study and compare two standard algorithms, namely the Random Walk (RW) and the Particle Swarm Optimization (PSO), in terms of speed (i.e. running time) and quality (i.e. coverage rate).

1.1. Related works

Sensor positioning problem has been investigated since a few decades, mainly for videosurveillance [1]. Without any additional constraint, this problem is NP-Hard as stated in [1], [2] for the Watchman Route Problem (which is very similar to the optimal positioning of a camera network). Two solutions have been proposed. The first one is based on Art Gallery Problem (AGP) [3], [4]. The second way is using the hypothesis of work from Wireless Sensor Networks [5], [6] and try to find the best position for creating an efficient network for collecting data with any kind of sensor. However, the solution propose of this problem is working only with some constraint like if the sensor has 360 degrees field of view, no obstacle. One of the most efficient algorithm used is PSO as detailed in [7], [8]. In [7], some experimental results are provided and one solution running in real time is proposed. However, the scene used for the experiments is rather small and many cameras are employed to fully cover it. On the other hand, [8] uses a cost function but the cost function is not only focused on the position for surveillance, but also handling resolution and lighting. This paper also introduces the concept of acceptable response, allowing non-optimal/sub-optimal solutions. If the coverage score is good enough, the solution is accepted and not locked by the research of an optimal. The article uses the paradigm of the AGP and PSO algorithm. The main drawback of this paper is the use of a greedy implementation of PSO and greed algorithm cannot adapt to the environment at time. Our paper is directly based on [7], [8], attempting to extend it by adding degrees of freedom and a new optimization scheme.

1.2. Objectives

The objective is the creation of a video surveillance system that can be extremely adaptable and dynamic to meet the current requirements including monitoring in key areas. The first goal is to position the cameras to get the best coverage. Although coverage is important, optimizing the cover-

```
Covered=0
Uncovered=Pi
For all the point of the grid uncovered
    For all camera Cj
    If (Pi is visible by Cj)
        Covered=Pi
    Else
        unCovered=Pi
            End
    End
End
```

Figure 1. Algorithm for calculate the cover of one camera with grid of the floor.

age may be detrimental to the image quality. The objectives: 1) Optimization of the camera positioning depending on a dynamic environment, the coverage area and image quality 2) Real time optimization of camera motion depending on the freedom of movement for each camera of the network But the aim of this paper is to find the best solution to cover a room with a fix number of cameras. Its the first step of the video surveillance project.

2. Problem statement

To find the best position of a set of cameras there are two key points. The first key points is to develop an efficient algorithm to find best position for a rectangular room without any obstacle (like a wall or pylons) and all the cameras must be fixed on the ceiling toward the ground. PSO is a computation method and try to optimize the problem by iteratively test. All step the algorithm takes the previous best solution and search a new set of solutions near to the best previous solution despite to the inertia. The inertia is a parameter to PSO using to limit the field of search on every step. The choice of PSO is appropriate to find the position of a camera for two main reasons.

- Because is an optimization method and try to find the cameras positions is NP-hard and in this case you can find the optimal solution
- PSO is already used on some paper for choosing the position of a set of camera in certain condition [7], [8]

The other key point is to qualify the quantity and the quality of the solution. For that its necessary to calculate the coverage of one camera. To estimate the coverage of one camera, its easier to use the pinhole model and a grid of points (presented at figure 2) and calculate what point is visible by one camera.

To do this, its easier (and fast for computation) to use the pinhole camera model. Its necessary to put a grid of point on the floor and to any point its calculated if this point of the grid (figure 2) is covered by minimum one camera. Follow this algorithm, see (figure 1)

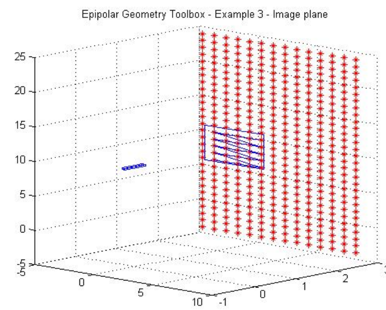

Figure 2. Visualization grid with one camera coverage.

2.1. Cost function for PSO and random walk

When you know how to calculate the coverage of one camera. Its possible to create one cost function to qualify the quality of the answer proposed by any kind of algorithm. The cost function is defined like below (1) n= number of camera Grid = is a grid represent the floor. Its used to calculate the coverage of a room
cover (n)=Coverage of the camera.

$$\text{Coverage rate of the room} = \sum_{i=1}^{n} \frac{Cover(i)}{Size(grid)} text(1)$$

3. Random walk and PSO

Random walk is compared to PSO with two degrees of freedom. Tow point was compared, one is the time commutation and the second is the coverage rate. The experimentation is implemented by using Matlab to compute the position of cameras and optimize it with the PSO algorithm (toolbox PSO [9], [10]) and random walk. The first experiment shows the efficiency of PSO compared to random walk for cover an area (see figure 3 and figure 5) but the speed (number of iterations) is much longer with PSO to Random walk (see figure 4 and figure6)

3.1. Random walk

The first experiment use a basic algorithm. The random walk work with successive selection of random solution, try it and if the solution is better to the precedent solution this is the new reference. If after 100 test (figure 4) the solution was not better, the algorithm stop and give the final solution. This basic algorithm give some result, and this result is the point of reference for the other algorithm The limit of the random walk is at the first the result proposed was not enough efficient (figure 3) and at same time, many gap appear between the results with the same input parameter (the gap vary between 2 percentages point to 5 percentages point). The other limit of the random walk is time computation (figure 4) with big gap between the min number of

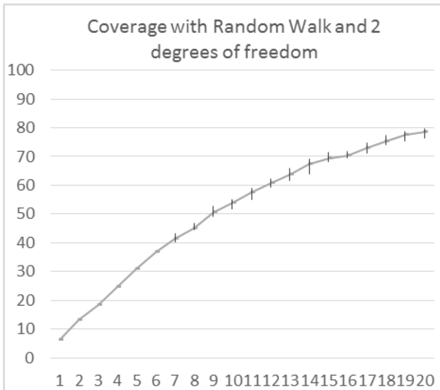

Figure 3. Percentage of coverage with n number of camera (1 to 20). The camera was positioning with basic Random Walk algorithm for optimize the position on X and Y pose

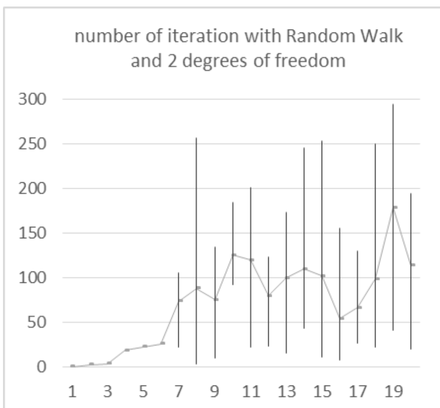

Figure 4. Number of iterations before converged solution. The algorithm implement for the random walk stop when its impossible to find best solution after 100 random set of camera pose.

iteration and the max and the time computation increase really fast despite the little number of camera. The problem caused by the time computation can be a curb for the future deployment for an adaptive camera network. Now the limit of random walk was clearly identify its important to find a better algorithm.

3.1.1 PSO with 2 degrees of freedom

Comparing the results obtained with the two algorithms (random walk and PSO) allows to evidence that Although the random walk is 2 time faster (see figure 4 and figure 6)for all size of the network camera, but the coverage proposed by the random walk is not acceptable and use too many camera for a complete coverage of the room, (see fig 5 and figure 6) more even if the PSO make more time the result is much better and can have the complete coverage with one smaller camera network and for any number of camera you have best result with PSO. But the number of cameras is too much important and to try to down the num-

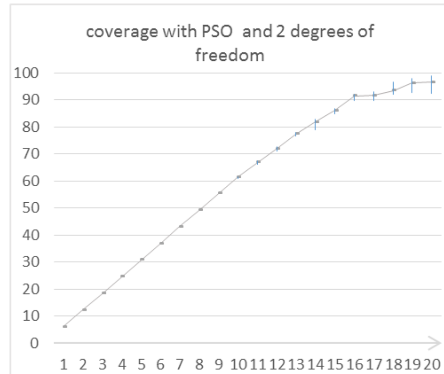

Figure 5. Percentage of coverage with n number of camera (1 to 20). The camera was positioning with PSO algorithm for optimize the position on X and Y pose.

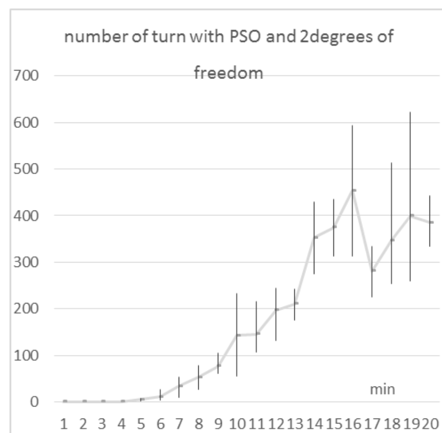

Figure 6. Number of iterations before converged solution. The algorithm implement for the PSO stop when its impossible to find best solution after 100 random set of camera pose.

ber of sensors used on the network, is possible to add one more degrees of freedom.

4. Experiments and results

For this experiment the goal is to find the best position for one cameras network. To have better result of the 1st experiment (less camera and better coverage, see figure 3 and 5) it is necessary to add one more degrees of freedom. The Pan was chosen because is a more realistic solution and interesting for snap workable image to use for video surveillance (with the adequate pan you can do face recognition [11] or object detection)[12] . For the experiment the camera accepts a pan angle θ between $-\pi/4$ to $\pi/4$

After this last experimentation, its interesting to compare the result for the speed and the quality of coverage. Add one more degrees of freedom in the system can help for have better solution of coverage, because when with 2 degrees of freedom PSO need 21 cameras for a complete coverage (figure 5) the same experience, but with 3 degrees

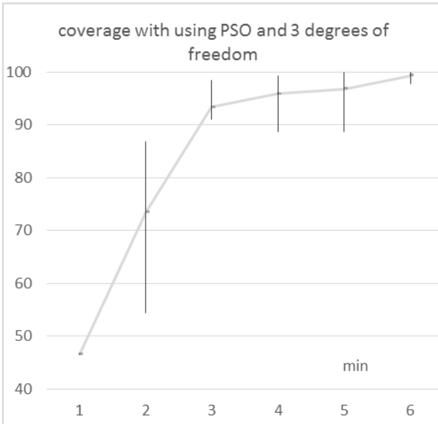

Figure 7. Percentage of coverage with n number of camera (1 to 6). The camera was positioning with PSO algorithm to optimize the position on X, Y and the pan of pose.

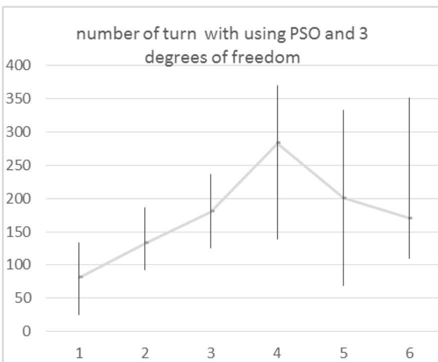

Figure 8. Number of iterations before converged solution. The algorithm implement for the PSO stop when its impossible to find best solution after 100 random set of camera pose

of freedom (x y and) find an solution with 6 cameras (figure 7). But even if the time computation is better compared to PSO with 2 degrees of freedom, its still little slower than random walk with 2 degrees of freedom (compare figure 8 and figure 4)

4.1. Visualization with simulation robot tool

To the visualization of the result, the tool is used from the robotic V-rep **verp** this tool can help to visualize the limit of PSO on the real world.

After some experimentation is interesting to use the result find with Matlab, the solver model and the PSO (for 3 degrees of freedom) with a simulation system for robots. This experiment with robot simulation (see figure 9) can proof at the same time the efficiency of the algorithm and can help to visualize the limit of the system

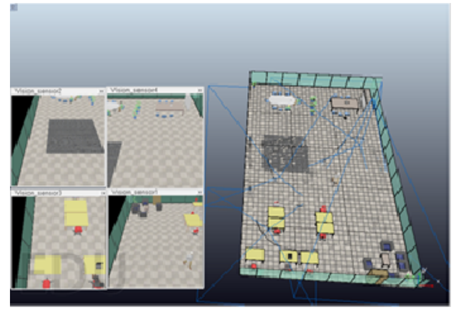

Figure 9. The result of one experimentation with four camera. The position of the camera and the pan is implement in a robot simulator for have a visualization of the final result

5. Conclusion and future works

In this paper the problem of coverage for video surveillance was show like optimization problem. Random walk and after PSO was using for find acceptable solution and add one more degrees of freedom for have more interesting result. But the experiment show the limit at PSO like the time computation. This limit can be curb and not enough fast for the original goal to create a system of video surveillance dynamic and capable to react despite to the external constraint (real time adaptation). More the time computation is important and that can limit the application moreover the actual algorithm cant take into account the obstacle. All this limitation can give some possible future progresses and one of the way to progress is to find some constraint or degrees of freedom like previously the pan. The next step can be to give at the cost function the possibility to manage the multi coverage by zone and add the possibility to zoom.

This work is supported by the Regional Council of Burgundy. We are extremely grateful to members of laboratory LE2I for day to day help and collaboration.

References

[1] W.-p. Chin and S. Ntafos, "Optimum watchman routes," *Information Processing Letters*, vol. 28, no. 1, pp. 39–44, 1988 (cit. on p. 1).

[2] J. Zhao, S.-C. Cheung, and T. Nguyen, "Optimal camera network configurations for visual tagging," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 2, no. 4, pp. 464–479, 2008 (cit. on p. 1).

[3] M. Moeini, A. Kröller, and C. Schmidt, "Une nouvelle approche pour la résolution du problème de la galerie d'art," (cit. on p. 1).

[4] U. M. Erdem and S. Sclaroff, "Automated camera layout to satisfy task-specific and floor plan-specific coverage requirements," *Computer Vision and Image Understanding*, vol. 103, no. 3, pp. 156–169, 2006 (cit. on p. 1).

[5] B. Song, C. Soto, A. K. Roy-Chowdhury, J. Farrell, *et al.*, "Decentralized camera network control using game theory," in *Distributed Smart Cameras, 2008. ICDSC 2008. Second ACM/IEEE International Conference on*, IEEE, 2008, pp. 1–8 (cit. on p. 1).

[6] Q. Wang, J. Wu, and C. Long, "On-line configuration of large scale surveillance networks using mobile smart camera," in *Distributed Smart Cameras (ICDSC), 2013 Seventh International Conference on*, IEEE, 2013, pp. 1–6 (cit. on p. 1).

[7] P. Zhou and C. Long, "Optimal coverage of camera networks using pso algorithm," in *Image and Signal Processing (CISP), 2011 4th International Congress on*, IEEE, vol. 4, 2011, pp. 2084–2088 (cit. on pp. 1, 2).

[8] K. K. Reddy and N. Conci, "Camera positioning for global and local coverage optimization," in *Distributed Smart Cameras (ICDSC), 2012 Sixth International Conference on*, IEEE, 2012, pp. 1–6 (cit. on pp. 1, 2).

[9] (). Http://www.mathworks.com/matlabcentral/fileexchange/7506-particle-swarm-optimization-toolbox/content/pso$_{t}relea_{v}ectorized.m$ (cit. on p. 2).

[10] (). Http://www.coppeliarobotics.com/ (cit. on p. 2).

[11] L. An, M. Kafai, and B. Bhanu, "Face recognition in multi-camera surveillance videos using dynamic bayesian network," in *Distributed Smart Cameras (ICDSC), 2012 Sixth International Conference on*, IEEE, 2012, pp. 1–6 (cit. on p. 3).

[12] S. Parupati, R. Bakkannagari, S. Sankar, and V. Kulathumani, "Collaborative acquisition of multi-view face images in real-time using a wireless camera network," in *Distributed Smart Cameras (ICDSC), 2011 Fifth ACM/IEEE International Conference on*, IEEE, 2011, pp. 1–6 (cit. on p. 3).