Interface for Multi-robots Based Video Coverage

Ryad Chellali[1], Emmanuel Bernier [1,2], Khelifa Baizid[1,2], and Mohamed Zaoui[3]

[1] Italian Institute of Technology, Via Morego, 30, Genova 16163, Italy
[2] University of Genova, Genova, Italy
[3] Collège de France, France
{ryad.chellali,emmanuel.bernier}@iit.it,
Baizid.khelifa@gmail.com

Abstract. In this paper we address the problem of coverage area with multi-robot system by giving an autonomy level to single robot as well as the team. The coverage concept is based on Voronoi algorithm and it applied to extract the minimum number of views from the supervised environment. Human Robot Interface was developed to allow user to selects minimum points' locations where robots have to achieve the supervision task. Moreover, user can access at any time through the interface watching video streams feedback coming from robots and semi-autonomous robots can locally correct their paths if probability collusion, with the environment, is predicted based on new adapted Active Counter Model. However, failed robots can easily request help from user, where, this last acts overcoming the robot from its unlikely situation. The coverage strategy that we proposed combines the real robot motion (by consideration motion error and process the correction method) and the human robot interaction as well as the supervision paradigms.

Keywords: coverage, Active counter model, multi-robots system, semi-autonomous robot.

1 Introduction

When dealing with urban search and rescue operations, time is the critical parameter has to cope with: the faster one finds victims the more chance has to save lives. We are developing a multi-robots system for USAR (Urban Search And Rescue) purposes. One of the main topics we are tackling is dealing with the shared control to capture video streams about specific areas operators aim to cover. In order to reduce the workload of operators, our system supports as inputs the areas to cover. After that, the system determines the whole strategy for robots to find first the optimal position for each one. In a second stage, each robot achieves its safe movement toward the specified point derived from the previous phase.

Unfortunately, the semi-autonomous mode does not correspond to real conditions. Indeed the context and the working environments are at best partially known: after catastrophes or fires, these environments are no more conformal to plans rescuers have. Following that, operators' intervention is required to modify on the fly one or more robots' strategies in order to fulfill the requirements.

A. Marcus (Ed.): Design, User Experience, and Usability, Pt I, HCII 2011, LNCS 6769, pp. 203–210, 2011.

In this contribution, we detail the components of the interface we developed allowing operators to interact intuitively with robots both to specify the areas to cover as well as to support direct robots reallocation and redistribution in order to cover given areas.

2 Related Works

Several researches works have been carried out demonstrating the coverage problem. Where, the original research works has introduced by Gase [1]. This approach is an analytical technique to optimize the spread of a mobile robot group for the coverage area in a convex environment based on the Voronoi algorithm. However, this approach may not consider the challenge of environments with obstacles.

A similar approach studied the problem of optimal placement for mobile robots team in surveillance tasks for non-convex environment implementing the Voronoi method [2], robots displacement is based on the importance area estimated by sensor information. Heterogeneous robots group method [3] as extension of location optimization concept [4]. This method takes into consideration flying and ground robots. Another extension for the non-convex environment case with unknown obstacles using the combination of Voronoi and partition and potential field is also proposed in [5].

In [6] a method was proposed based on the potential field for convex environment coverage. Each robot must have a connectivity feature of the communication range of its neighbors to ensure the algorithm process. In [7] the authors introduced a method for non-convex environment based on the gradient algorithm to locate the mobile robot observer in the environment. However, this method works for single robot case.

Considering the visibility concept, an approach using multi-robot system and assumes that the environment is completely connected is presented in [8]. The visibility problem could be related to the Art Gallery Problem. This method looks for the optimum number of guards in a non-convex environment for which, each segment of the environment is visible at least from one guard [9], [10], [11].

In [12] a coverage method based on the cognitive adaptive optimization algorithm (CAO), it is dedicated for heterogeneous robots and unknown environment having obstacles.

However, most of the cited techniques give solutions on the basis of an a priori knowledge (of the robot motion and the environment), which may not correspond to reality. To cope with local uncertainties and local errors (movement and sensing errors), we proposed a method which allows robots to modify and optimize locally their routes. Moreover, in case of robots fail or detect a bottleneck situation, they inform the operator, who then takes the control of the blocked robot(s) to reprogram on the fly a new path and its feasibility is simulate by moving virtual robots within the top-view map. For our system, we use the RRT [13] algorithm to generate the a priori trajectory of the robot to reach the area to be supervised. While the mission is in progress, the robot corrects its path if needed (presence of obstacles) based on new adapted Active Counter Model [14] that we developed. In addition to smoothing the executed trajectories, the SNAKE based approach allowed us to include in our path generator, different uncertainties and modeling errors produced by on-board sensors.

3 Multi-robots Platform

We developed a multi-robots platform [15] composed of wheeled and legged robots. Each of them is endowed with a video camera and a laser-scanner. The embedded processing/sensing capabilities allow to avoid obstacles through the proximity sensors and to generate a free path given a global map. On the other hand, all robots can communicate with the central system and with each other through the wireless networking infrastructure. Last, the robots can be fully tele-operated by operators: a joystick allows controlling movements regardless to the mobility of the considered robot. As well, the same joystick allows the control of the embedded camera pan-tilt movements.

4 Supervision and Control System

To manage the robotics platform we developed a central system composed of I/O, control and supervision tools. For specifying the task, namely the coverage area, a top view map and a simple 2D input tablet are used. Operators can draw directly the concerned area; which is handled by the task generator. This later considers the a priori map to derive the free path and the final poses for each robot. We used the RRT algorithm [13] and for poses, we use Voronoi distribution an extension of the art gallery problem heuristics. Both poses and paths are sent to robots. Each of them works then in a semi-autonomous mode while reaching the goal. Operator follows the execution of the task (robots motion) on the top view map. As well, they can access to video streams sent by each robot.

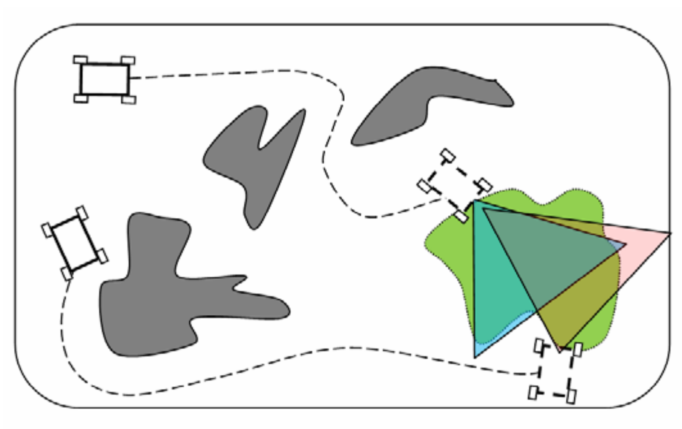

Fig. 1. An example of area coverage

4.1 Path Planning and Robots Autonomy

The path of the robots is generated using RRT algorithm, this algorithm is based on random note selection which process finally to a theoretically feasible path (without collusion). However, the generated path can work only if we consider no error on the robot motion, which is not almost the real case. So the RRT path generation method serves our platform, only, for a preliminary path definition, which could not be safe due to motion error of the real robot as well as sensor uncertainties. Considering real robot case, this last never follow theoretical path which makes the challenge to cope with this kind of problem very hard [16]. Our objective, is *first*, to unable the robot to reach a desired location without any collusion with the environment, where, the robot can modify its own path. if it predicts a collusion probability with the environment. *Second*, optimize the trajectory of the robot by minimizing the RRT path.

We proposed a new concept based on the Active Counter Model [14]. Unlike previous research used the Snake for Image Processing [17] we adapt this concept to cope with the path generation and correction. Begin; based on the force given by the obstacle the Snake modifies the path to be much safe. We build the energy of the Snake model based on the obstacles and the robot path; both are combined within a defined process. Where, the robot motion errors and the sensor uncertainties are merged in which the result led to a safe trajectory, by modifying this last.

Based on the energy minimization our Snake model will behave under the influence of the environment and the internal trajectory forces. Where, the external force of the environment (Obstacle and trajectory) push the counter outside zones having big collusion risk probability. The internal force keeps the smoothness of the counter as it possible. While, the optimization force reduces bends generated by the RRT. Assuming that the trajectory parameters are written by $v(s) = (x(s), y(s))$ The Snake energy model is defined as follow:

$$E_{Snake} = \int_0^1 E_{IntFrc}(v(s)) + \int_0^1 E_{ExtFrc}(v(s)) + \int_0^1 E_{OpFrc}(v(s)) \tag{1}$$

Where, E_{IntFrc} represents the internal energy of the counter, Where, E_{ExtFrc} represents the external energy of the counter (obstacle + trajectory) and E_{OpFrc} represents the energy to optimize the counter.

The internal energy is defined as follow:

$$E_{IntFrc} = \left(\alpha(s)|v(s)|^2 + \beta(s)|v(s)|^2\right)/2 \tag{2}$$

Where the first part (function of α) is a membrane force of the counter, while the second part (function of β) is a force acting to counter to be thin.

We are using the term of to 1- the only the first part of the External energy force $E_{ExtFrc} = -\left(G_\sigma * \nabla^2 I\right)^2$ defined in [16] is defined as follow. In this case our external energy is defined as follow: $1 - E_{ExtFrc}$.

The Optimization energy $E_{OpFrc} = \int_0^1 \sum \vec{f}(x, y)dt$ is defined by the:

$$F = N * \vec{n}(s) \tag{3}$$

Where $\vec{n}(s)$ represent the normal unitary vector to the curve at point $v(s)$ and N weight, for a large value of N the curve converges to straight line very quickly. The normal of each point is strongly related to all previous and next points.

The obstacle's primitives are generating using distance sensor (URG laser) the obstacles are modeled by a polygons (fig. 2. (b)). While, the sensor error is modeled by a Gaussian function, which applied to the primitives to cope with the error Δt .

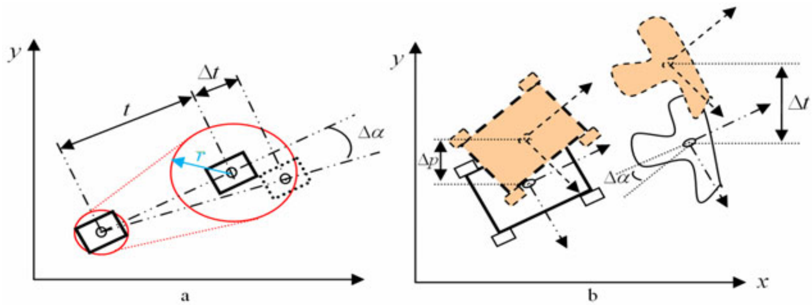

Fig. 2. Primitive concept, (a) path motion, (b) position tracking on the obstacles primitives

Trajectory primitive are represented regarding to the error motion of the robot, where, the position $(x, y)_i$ is related to the length of the executed motion from $(x, y)_{i-1}$ to $(x, y)_i$. So the robot can reach any position in the red circle. Fig. 2.(a) is showing the envelope of robot motions which can be accumulated from position *i* to *i+1*. Because of the correction strategy of the robot, that our platform allows, the errors are not accumulated in all positions, but only between the two which are successive. The radius of the red circle (fig. 2.(a)) which represents the possible reached points from the robot related to the uncertain position $(x, y)_i$ is defined as follow: $r = \max\left(\Delta t / \cos(\Delta\alpha), t * \sin(\Delta\alpha)\right)$. For the uncertainty of the distance sensor the translation Δt represented in fig. 2.(b) because of the error in the tracking system, which calculate the robot position using SLAM technique [18]. Where, the robot position uncertainty Δp affects an error on the obstacle because this last is represented in the robot scale.

4.2 Robot Placement Algorithm

The aim of this part is to find the minimal number of sensor position from which the full coverage of the targeted area is effective. The problem is similar to the classical AGP problem, where one determines the minimal number of guards and their positions to watch all gallery walls. Let us consider a camera as the used sensing system to perform the coverage. Let us as well consider that the environment is polyhedral. In this case, a classical Voronoi diagram can provide the set of points from which one is guarantied to achieve the watching task (fig.3a). Indeed, the external Voronoi vertices (fig.3b) are equally situated from objects borders and belong the respective polygons sides' bisectors. Obviously, this set is not optimal and overlaps may appear. To reduce the number of points, we developed a visibility propagation algorithm [19]. The process starts by detecting the minimal points that one object needs to be covered (parts in red in fig-3c). For the presented case, we have 4 points. These points are used to propagate the visibility to the other objects. Once a part of any polygon is seen, it is removed. We rerun the first step on a new object: for object (2) in fig-4d, we have 3 new points to fully cover the object. We propagate the visibility to the two lasting objects. In fig-3e and fig-4f we have respectively two and two new points. In total we have 12 points to cover the full area regardless to concavities, occlusions or other classical limitations.

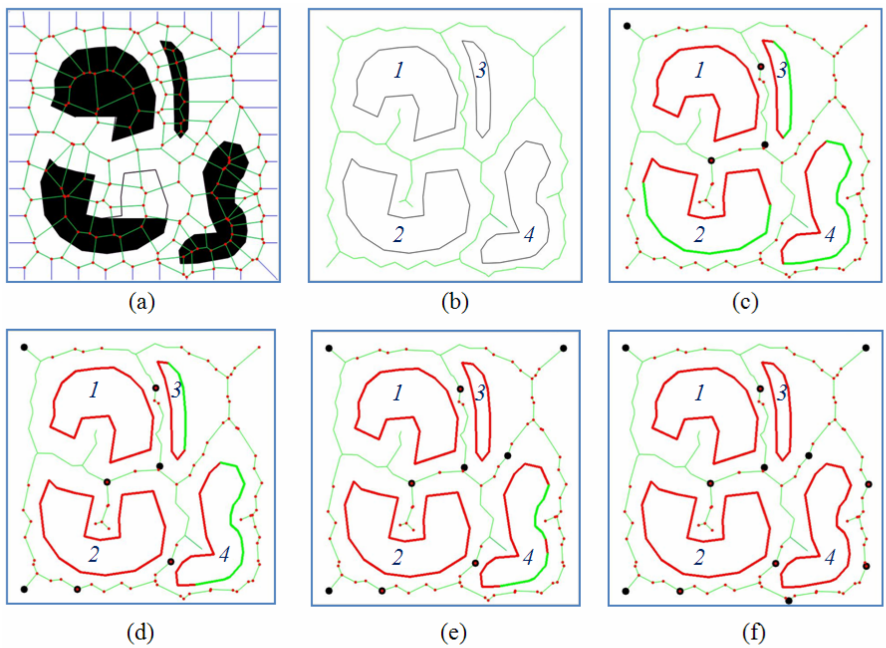

Fig. 3. Coverage area algorithm example, (a) external Voronoi vertices distribution, (b) initial points configuration, (c) to (f) the first, second, third and fourth object coverage respectively

5 Experiments Scenario

The full platform was tested in our laboratory conditions to assess and to validate all the developed blocks. Currently we are comparing between the results one can obtain through the full tele-operated mode and the semi-autonomous mode in order to show the effectiveness of the developed algorithm (fig. 4).

An example of the self-correction of the robot path is shown in fig. 4. By using the interaction interface user select the robot in (fig.4b) and then start the action in (fig.4a) by showing the target in (fig.4c). After generating the path primitive (fig.4d-right) a Gaussian process is applied on this path as well as to the obstacles. Based on this the external force will be computed (fig.3d-up right). (fig.4d-right-sown) contains the original path with black color (generated by the RRT), the Snake's optimized path with green color and the real robot path execution with the red color. In this case the Snake has pushed the path to be more save (fig.4d-right-up).

However, an unknown blue obstacle (because of the occlusion problem) makes the robot staked during the execution, where, the operator has fully tele-operates the robot (blue portion) to help it to overcame this problem, and then the robot has continues exciting the path (last read portion) until arrived to the target.

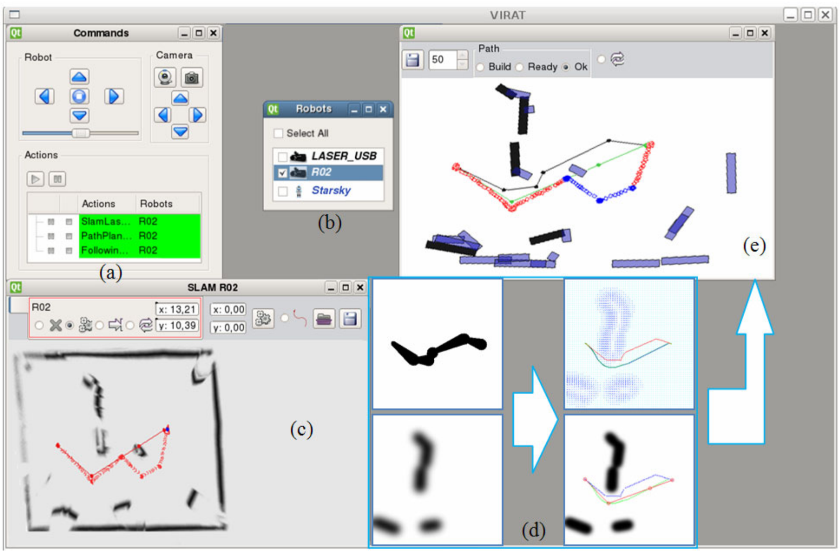

Fig. 4. Interaction interface with safe path using Snake concept, (a) command windows, (b) robots selection, (d-right) path and obstacle primitives and their Gaussian distribution, (d-left) external force, (e) original, Snake's and real paths

6 Conclusion

A new method for coverage area considering a shared and robust control between a real robot and human operators has been proposed. The goal was to optimize the task by extracting, the minimum points of views as a first level, while the second optimization level is done with the robots' path using new model of SNAKE concept. From the robot autonomy point of view, results show that the real robot motion and distance sensor uncertainties are well handled and the robot can modifies its own path to be much more safe by minimizing collisions probability. The proposed combination between of the human supervision and robot autonomy, allowed performing successfully the task, even with unknown situations that the real robot has found.

References

1. Douglas, W.: Gage: Command control for many-robot systems. In: Nineteenth Annual AUVS Technical Symposium, Hunstville Alabama, USA, pp. 22–24 (1992)
2. Schwager, M., McLurkin, J., Rus, D.: Distributed Coverage Control with Sensory Feedback for Networked Robots. In: Proceedings of Robotics: Science and Systems, Philadelphia, USA (2006)
3. Pimenta, L.C.A., Kumar, V., Mesquita, R.C., Pereira, G.A.S.: Sensing and Coverage for a Network of Heterogeneous Robots. In: Proceedings of the 47th IEEE Conference on Decision and Control, Mexico, pp. 3947–3952 (2008)
4. Cortes, J., Martınez, S., Karatas, T., Bullo, F.: Coverage Control for Mobile Sensing Networks. IEEE Transactions on Robotics and Automation 20, 243–255 (2004)
5. Renzaglia, A., Martinelli, A.: Distributed coverage control for a multi-robot team in a non-convex environment. In: IEEE IROS09 3rd Workshop on Planning, Perception and Navigation for Intelligent Vehicles, St. Louis, MO, USA, pp. 76–81 (2009)
6. Poduri, S., Sukhatme, G.S.: Constrained Coverage for Mobile Sensor Networks. In: IEEE International Conference on Robotics and Automation, New Orleans, LA, USA, pp. 165–172 (2004)
7. Ganguli, A., Cortes, J., Bullo, F.: Maximizing visibility in nonconvex polygons: nonsmooth analysis and gradient algorithm design. In: American Control Conference, Portland Oregon, USA, vol. 2, pp. 792–797 (2005)
8. Ganguli, A., Cortes, J., Bullo, F.: Visibility-based multi-agent deployment in orthogonal environments. In: American Control Conference, New York, USA, pp. 3426–3431 (2007)
9. Ferrari, S.: Track Coverage in Sensor Networks. In: American Control Conference, vol. 2, pp. 14–16 (2006)
10. Shermer, T.C.: Recent results in art galleries. IEEE Proceedings 80, 1384–1399 (1992)
11. O'Rourke, J.: Art Gallery Theorems and Algorithms. Oxford University Press, New York (1987)
12. Renzaglia, A., Doitsidis, L., Martinelli, A., Kosmatopoulos, E.B.: Cognitive-based adaptive control for cooperative multi-robot coverage. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3314–3320 (2010)
13. Lavalle, S.M., Kuffer, J.J.: Rapidly-exploring random trees: Progress and prospects. In: Workshop on the Algorithmic Foundations of Robotics (2000)
14. Kass, M., Witkin, A., Terzopoulos, D.: Snakes Active contour models. Journal of Computer Vision, 321–331 (1987)
15. Baizid, K., Zhao, L., Mollet, N., Chellali, R.: Human Multi-Robots Interaction with High Virtual Reality Abstraction Level. In: 2nd International Conference on Intelligent Robotics & Applications, Singapore (2009)
16. Bouilly, B., Simeon, T., Alami, R.: A numerical technique for planning motion strategies of a mobile robot in presence of uncertainty. In: IEEE International Conference on Robotics and Automation, vol. 2, pp. 1327–1332 (1995)
17. Aizu, M., Nakagawa, O., Takagi, M.: An image segmentation method based on a color space distribution model. In: International Conference on Mechatronics and Automation, China, pp. 532–533 (2006)
18. Steux, B., Oussama, H.: Oussama.: tinySLAM: A SLAM algorithm in less than 200 lines C-language program. In: International Conference on Control Automation Robotics Vision, pp. 1975–1979 (2010)
19. Chellali, R., baizid, K.: New Algorithm for Coverage Problem. Technical Report, IIT (February 2011)