HAL
open science

3D Surveillance Coverage Using Maps Extracted by a Monocular SLAM Algorithm

Lefteris Doitsidis, Alessandro Renzaglia, Weiss Stephan, Elias Kosmatopoulos, Davide Scaramuzza, Roland Y. Siegwart

To cite this version:

Lefteris Doitsidis, Alessandro Renzaglia, Weiss Stephan, Elias Kosmatopoulos, Davide Scaramuzza, et al.. 3D Surveillance Coverage Using Maps Extracted by a Monocular SLAM Algorithm. IEEE International Conference on Robotics and Intelligent System (IROS), Sep 2011, San Francisco, United States. inria-00625534

HAL Id: inria-00625534

https://inria.hal.science/inria-00625534

Submitted on 21 Sep 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

3D Surveillance Coverage Using Maps Extracted by a Monocular SLAM Algorithm

Lefteris Doitsidis, Alessandro Renzaglia, Stephan Weiss, Elias Kosmatopoulos,
Davide Scaramuzza & Roland Siegwart

***Abstract*— This paper deals with the problem of deploying a team of flying robots to perform surveillance coverage missions over a terrain of arbitrary morphology. In such missions, a key factor for the successful completion is the knowledge of the terrain's morphology. In this paper, we introduce a two-step centralized procedure to align optimally a swarm of flying vehicles for the aforementioned task. Initially, a single robot constructs a map of the area of interest using a novel monocular-vision-based approach. A state-of-the-art visual-SLAM algorithm tracks the pose of the camera while, simultaneously, building an incremental map of the surrounding environment. The map generated is processed and serves as an input in an optimization procedure using the cognitive adaptive methodology initially introduced in [1], [2]. The output of this procedure is the optimal arrangement of the robot team, which maximizes the monitored area. The efficiency of our approach is demonstrated using real data collected from aerial robots in different outdoor areas.**

I. INTRODUCTION

The use of multi-robot teams has gained a lot of attention in the recent years. This is due to the extended capabilities that multiple robots offer with respect to a single robot for the same task. Robot teams can be used in a variety of missions including: surveillance in hostile environments (i.e. areas contaminated with biological, chemical or even nuclear wastes), environmental monitoring (i.e. air quality monitoring, forest monitoring) and law enforcement missions (i.e. border patrol), etc. In all the aforementioned tasks there are several crucial factors that affect the overall behavior of the robot teams. These include, but are not limited to, the sensors the robots might have, the size of the robot team, the type of robots used, etc. In this paper, we introduce a two-step centralized procedure to align optimally a swarm of flying vehicles. Initially, a single robot constructs a map of the area of interest using a monocular-vision-based approach. A state-of-the-art visual-SLAM algorithm tracks the pose of the camera while, simultaneously, building an incremental map of the surrounding environment. The generated map is processed and serves as an input in an optimization procedure using the cognitive, adaptive methodology initially introduced in [1], [2]. The optimization objectives are the following:

(O1) The part of the terrain that is "visible" – i.e. that is monitored – by the robots is maximized;

(O2) The team members are arranged so that for every point in the terrain, the closest robot is as close as possible to that point.

The majority of existing approaches for multi-robot surveillance coverage, which concentrate mostly on the 2D case of ground robots, deal only with one of the objectives (O1) or (O2); see e.g. [3]-[9] and the references therein. Furthermore, in most of the existing approaches the terrain morphology is considered *convex* and/or *known*. In such cases the problem of multi-robot surveillance coverage can be seen to be equivalent to a standard optimization problem where the robot trajectories are generated according to a gradient-descent or gradient-descent-like methodology. However, in the case where it is required that both of the objectives (O1) and (O2) are simultaneously addressed and the terrain's morphology is non-convex and unknown, standard optimization tools are not applicable anymore as these tools require full knowledge of an objective function that depends on the unknown terrain's morphology. To overcome the above-mentioned shortcomings of the existing approaches for multi-robot surveillance coverage, we propose a new solution that is based on the recently introduced Cognitive-based Adaptive Optimization (CAO) algorithm [11], [12]. The main advantage of CAO as compared to standard optimization tools is that it does not require that the objective function to be optimized is explicitly known; CAO instead requires that at each time instant a value (measurement) of this objective function is available. As a result, if it is possible to define an objective function which may be unknown and depend on the unknown terrain morphology but is available for measurement for every given team configuration, the CAO methodology will be directly applicable to the problem of surveillance coverage treated in this paper. By introducing an appropriate objective function, that is defined so that both objectives (O1) and (O2) are simultaneously fulfilled, we manage to make the CAO algorithm applicable to the particular problem of 3D multi-robot surveillance coverage treated in this paper. This objective function depends on the

L. Doitsidis is with the Dept. of Electronics, Technological Educational Institute of Crete, Chania, 73133, Greece and the Informatics & Telematics Institute, Center for Research and Technology–Hellas (ITI-CERTH), Thessaloniki, Greece, ldoitsidis@chania.teicrete.gr.
Alessandro Renzaglia is with INRIA Rhône-Alpes, Grenoble, France, alessandro.rezanglia@inrialpes.fr.
Stephan Weiss is with ETH Autonomous Systems Laboratory, 8092, Zurich, Switzerland, www.asl.ethz.ch, stephan.weiss@mavt.ethz.ch.
E. B. Kosmatopoulos is with the Dept. of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi 67100, Greece and the Informatics & Telematics Institute, Center for Research and Technology–Hellas (ITI-CERTH), Thessaloniki, Greece, kosmatop@ee.duth.gr.
Davide Scaramuzza is with GRASP Lab, University of Pennsylvania, Philadelphia, USA, sdavide@seas.upenn.edu.
Roland Siegwart is with ETH Autonomous Systems Laboratory, 8092, Zurich, Switzerland, www.asl.ethz.ch, rsiegwart@ethz.ch.

unknown terrain's characteristics and thus its explicit form is not known. However, for any given team configuration the value of this objective function can be directly computed from the robots' sensor measurements, and thus the CAO algorithm can be applied to the problem at hand by using such an objective function. It has to be emphasized that, apart from rendering the problem of simultaneously addressing (O1) and (O2) for unknown terrains solvable, the CAO-based approach preserves additional attributes that make it particularly tractable: it can easily handle a variety of physical constraints and limitations and it is fast and scalable. These further attributes of the proposed CAO-based approach are detailed in the next section. It is mentioned that CAO does not create an approximation or estimation of the obstacles location and geometry; instead, it produces on-line a **local approximation** of the (unknown) cost function the robots are called to optimize. For this reason, it requires simple and thus scalable approximation schemes to be employed. A key issue for the successful implementation of the CAO proposed methodology in the case of a team of Micro Aerial Vehicles (MAVs), is the accuracy of the input it will have, which in this case is a 3D map. Since we deal with MAVs, the choice of sensors to perceive the environment to be monitored and therefore to construct the 3D maps is limited. For GPS-denied navigation and mapping, vision sensors and laser range finders are the only option. However, laser scanners are too heavy for MAVs and have a limited field of view. Therefore, cameras and inertial sensors are the only viable solution for such limited weight and calculation power budgets. For ground vehicles (cars), 3D occupancy grids built from stereo vision and GPS data have been shown to be a valid solution [13]. However, occupancy grids are not a good option for MAVs because of their limited calculation power. Lacroix [14] presented an off-line method to map a large outdoor scenario in fine resolution using low-altitude aerial stereo-vision images. Because stereo vision loses its advantage when the baseline is too small compared to the scene depth, we rely on a monocular solution in which the appropriate baseline is provided by a keyframe-based visual SLAM framework [15].

The rest of the paper is organized as follows. In section 2 we describe our visual-SLAM algorithm and how it is combined with the cognitive based adaptive optimization approach, while in section 3 we provide experimental results using data obtained by real aerial robots. Finally in section 4 we raise issues for discussion and future work.

II. PROPOSED METHODOLOGY

In order to apply our methodology a two-step procedure is applied. Initially, we use a single aerial vehicle to construct a map of the area of interest using a monocular-vision-based approach. This map is used as an input to the next step which is the optimization procedure. In this section we will initially describe the generation of the real-time 3D-map using monocular SLAM and then how we use it as an input into the cognitive optimization algorithm.

A. Real-Time 3D-Map Generation Using Monocular SLAM

To perform optimal surveillance coverage over an arbitrary terrain, we need to reconstruct the area in 3D. Note that most works on optimal coverage assume an existing map. In this work, we use an approach to build this map online and in real-time. Thus, the MAV has to be able to fly autonomously in the yet-unknown and later-mapped area. For the vision-based autonomous navigation, we use the approach described in our previous work [16]-[17]. For the 3D reconstruction we apply our meshing approach from [18]. The base for all our visual approaches, is a monocular SLAM framework ([19] in this particular case). This allows us to map and navigate in unknown environments. The SLAM algorithm consecutively builds a 3D sparse point cloud of the terrain to be observed and yields also the camera pose for controlling the MAV. We use a down-looking camera on the MAV and thus we can easily mesh the point-cloud into a 3D elevation mesh [18]. As described in [18] we first extract the principal plane of the map. Then we project the 3D point-cloud onto this plane. We have now a 2D meshing problem which we solve efficiently using a 2D delaunay approach. We reproject the obtained 2D mesh information to the 3D points in the map. Note that we only use points with a certain accuracy level given by the SLAM algorithm. After median-smoothing the mesh-map we have a good input for our optimal 3D coverage algorithms. Note that we can recover the absolute scale factor of the monocular SLAM by using an inertial sensor as we described in [20]. This way, we can reconstruct a metric 3D mesh-map of an arbitrary terrain. Figure 1 shows the initialization of the visual SLAM algorithm and the reconstruction of our test terrain.

B. The Cognitive-Based Optimization Approach

The Cognitive-based Adaptive Optimization (CAO) approach [10]-[12] was originally developed and analyzed for the optimization of functions for which an explicit form is unknown but their measurements are available as well as for the adaptive fine-tuning of large-scale nonlinear control systems. In this section, we will describe how the CAO approach can be appropriately adapted and extended so that it is applicable to the problem of multi-robot coverage. More explicitly, let us consider the problem where M robots are involved in a coverage task, attempting to optimize a given *coverage criterion*. Apparently, the coverage criterion is a function of the robots' positions or poses (positions and orientations), i.e.

$$J_k = \mathscr{J}\left(x_k^{(1)}, \ldots, x_k^{(M)}\right) \tag{1}$$

where $k = 0, 1, 2, \ldots$ denotes the time-index, J_k denotes the value of the coverage criterion at the k-th time-step, $x_k^{(1)}, \ldots, x_k^{(M)}$ denote the position/pose vectors of robots $1, \ldots, M$, respectively, and $\mathscr{J}$ is a nonlinear function which depends, apart from the robots' positions/poses, on the particular environment where the robots live; for instance, in the 2D case the function $\mathscr{J}$ depends on the location of the various obstacles that are present, while in the 3D case with

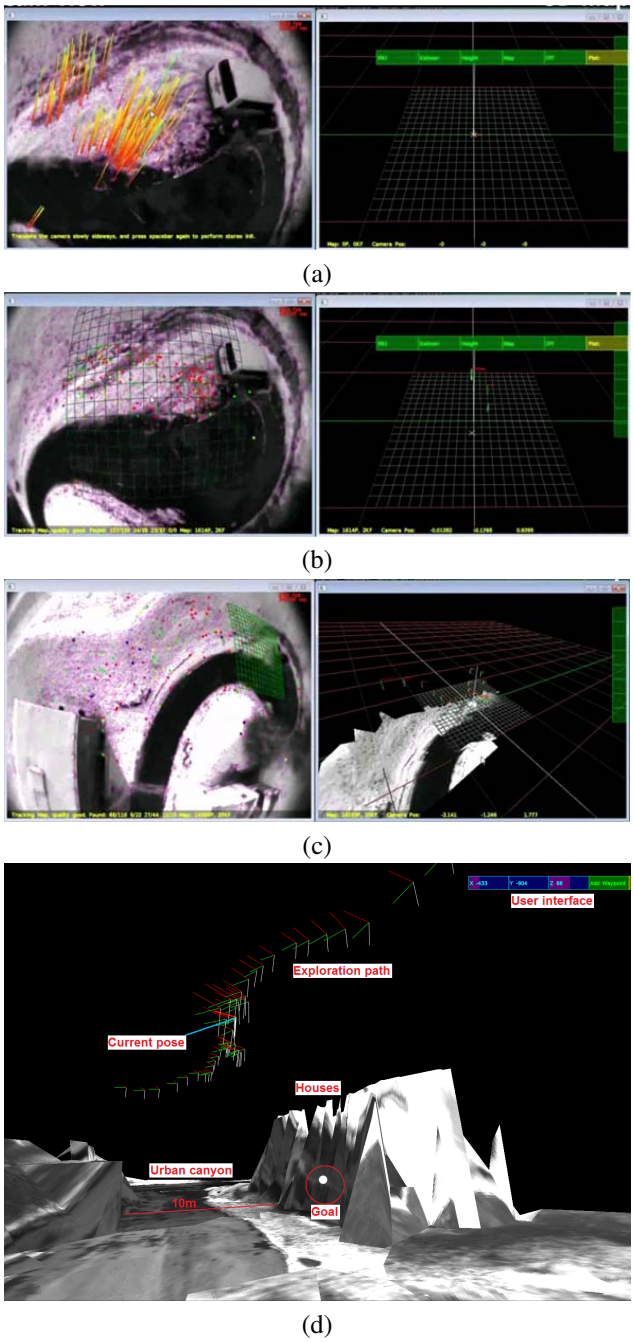

Fig. 1. (a) Initialization of the visual SLAM algorithm (on the left the tracked features used to initialize the map, on the right the reference frame). (b) The reference frame is displayed as a grid on the image (left). On the right, a few reconstructed camera poses are displayed as faint tripods. The bold tripod is the actual camera pose. This pose is used for the MAV position controller and yields the metric map scale by fusing it with the IMU measurements. (c) Generation of the textured map. (d) Sample of a meshed and also textured (snowy) outdoor environment. For the CAO approach the generated 3D mesh is sufficient, however, the texture gives the user intuitive information of where the MAV is positioned at the given time instance. Even with the texturing, this approach runs in real-time. Note that the reconstruction precision is not very high. It is, however, largely sufficient for our optimal-coverage tasks. Aid of the IMU we have a metric map and estimate here the urban canyon width to be about 10m (error is $<10\%$). The map reconstruction runs online while flying.

flying robots monitoring a terrain, the function $\mathscr{J}$ depends on the particular terrain morphology.

Due to the dependence of the function $\mathscr{J}$ on the particular environment characteristics, the *explicit form of the function $\mathscr{J}$ is not known* in most practical situations; as a result, standard optimization algorithms (e.g. steepest descent) are not applicable to the problem in hand. However, in most practical cases – like the one treated in this paper – the current value of the coverage criterion can be estimated from the robots' sensor measurements. In other words, at each time-step k, an estimate of J_k is available through robots' sensor measurements,

$$J_k^n = \mathscr{J}\left(x_k^{(1)}, \ldots, x_k^{(M)}\right) + \xi_k \tag{2}$$

where J_k^n denotes the estimate of J_k and ξ_k denotes the noise introduced in the estimation of J_k due to the presence of noise in the robots' sensors. Please note that, although it is natural to assume that the noise sequence ξ_k is a stochastic *zero-mean* signal, it is not realistic to assume that it satisfies the typical Additive White Noise Gaussian (AWNG) property even if the robots' sensor noise is AWNG: as $\mathscr{J}$ is a nonlinear function of the robots' positions/poses (and thus of the robots' sensor measurements), the AWNG property is typically lost.

Apart from the problem of dealing with a criterion for which an explicit form is not known but only its noisy measurements are available at each time, efficient robot coverage algorithms have additionally to deal with the problem of *restricting the robots' positions* so that obstacle avoidance as well as robot formation constraints are met. In other words, at each time-instant k, the vectors $x_k^{(i)}, i = 1, \ldots, M$ should satisfy a set of constraints which, in general, can be represented as follows:

$$\mathscr{C}\left(x_k^{(1)}, \ldots, x_k^{(M)}\right) \leq 0 \tag{3}$$

where $\mathscr{C}$ is a set of nonlinear functions of the robots' positions/poses. As in the case of $\mathscr{J}$, the function $\mathscr{C}$ depends on the particular environment characteristics (e.g. location of obstacles, terrain morphology) and an explicit form of this function may be not known in many practical situations; however, it is natural to assume that the coverage algorithm is provided with information whether a particular selection of robots' positions/poses satisfies or violates the set of constraints (3).

Given the mathematical description presented above, the multi-robot coverage problem can be mathematically described as the problem of moving $x_k^{(1)}, \ldots, x_k^{(M)}$ to a set of positions/poses that solves the following constrained optimization problem:

$$\begin{array}{ll} \text{minimize} & (1) \\ \text{subject to} & (3)\,. \end{array} \tag{4}$$

As already noticed, the difficulty in solving in real-time and in real-life situations the constrained optimization problem (4) lies in the fact that explicit forms for the functions $\mathscr{J}$ and $\mathscr{C}$ are not available. To circumvent this difficulty, the

CAO approach, appropriately modified to be applicable to the problem in hand, is adopted. This algorithm is capable of efficiently dealing with optimization problems for which the explicit forms of the objective function and constraints are not known, but noisy measurements/estimates of these functions are available at each time-step. More details about how the CAO approach is applied to the multi-robot coverage problem described above can be found in [1]- [21]. In the specific 3D case studied here the problem can be formulated as following.

Consider a team of M flying robots that is deployed to monitor an unknown terrain $\mathscr{T}$. Let $z = \Phi(x,y)$ denote the unknown height of the terrain at the point (x,y) and assume for simplicity that the terrain $\mathscr{T}$ is rectangular along the (x,y)-axes, i.e. $x_{min} \leq x \leq x_{max}, y_{min} \leq y \leq y_{max}$. Let $\mathscr{P} = \{x^{(i)}\}_{i=1}^{M}$ denote the *configuration* of the robot team, where $x^{(i)}$ denotes the position/pose of the i-th robot.

Given a particular team configuration $\mathscr{P}$, let $\mathscr{V}$ denote the *visible area* of the terrain, i.e. $\mathscr{V}$ consists of all points $(x,y,\Phi(x,y)) \in \mathscr{T}$ that are visible from the robots. Given the robots' sensor capabilities, a point $(x,y,\Phi(x,y))$ of the terrain is said to be <u>visible</u> if there exists at least one robot so that

- the robot and the point $(x,y,\Phi(x,y))$ are connected by a line-of-sight;
- the robot and the point $(x,y,\Phi(x,y))$ are at a distance smaller than a given threshold value (defined as the maximum distance the robot's sensor can "see").

Apparently, the main objective for the robot team is to maximize the visible area $\mathscr{V}$. However, this cannot be the only objective for the robot team in a coverage task: trying to maximize the visible area will simply force the robots to "climb" as high as[1] possible. In parallel to maximizing the visible area, the robot team should make sure that it *minimizes the average distance between each of the robots and the terrain subarea the particular robot is responsible for*, where the terrain subarea a particular robot is responsible for, is defined as follows: given a team configuration $\mathscr{P}$, the subarea of the terrain the i-th robot is responsible for is defined as the part of the terrain that (a) is visible by the i-th robot and (b) each point in this subarea is closer to the i-th robot than any other robot of the team. This second, and parallel to maximizing visibility, objective for the robot team is necessary for two practical reasons: (a) firstly, the closer is the robot to a point in the terrain the better is, in general, its sensing ability to monitor this point and (b) secondly, in many multi-robot coverage applications there is the necessity of being able to intervene as fast as possible in any of the points of the terrain with at least one robot. Having in mind that the robot team has to successfully meet the two above-described objectives, we define the following combined objective function the robot team has to minimize:

$$J(\mathscr{P}) = \int_{q\in\mathscr{V}} \min_{i\in\{1,\ldots,M\}} \left|x^{(i)} - q\right|^2 dq + K\int_{q\in\mathscr{T}-\mathscr{V}} dq \qquad (5)$$

[1]Note also that in the case where there are no limits for the robot's maximum height and the maximum sensing distance, it suffices to have a single robot at a very high position to monitor the whole terrain.

where K is a large user-defined positive constant. The first of the terms in the above equation is related to the second objective (minimize the average distance between the robots and the subarea they are responsible for) and the second term is related to the invisible area in the terrain ($\int_{q\in\mathscr{T}-\mathscr{V}} dq$ is the total part of the terrain that is not visible by any of the robots). The positive constant K is used to make sure that both objectives are met. To see this, consider the case where $K = 0$, in which case we will have that the robots, in their attempt to minimize their average distance to the subarea they are responsible for, may also seek to minimize the total visible area. On the other hand, in case where the first of the terms in (5) is absent, we will have the situation mentioned above where the robots in their attempt to maximize the visible area will have to "climb" as high as they are allowed to.

It has to be emphasized that the positive constant K should be chosen sufficiently large so that the second term in (5) dominates the first term unless no or a negligible part of the terrain remains invisible. In this way, minimization of (5) is equivalent to firstly making sure that all – or almost all – of the terrain is visible and then to locate the robots so that their average distance to the subarea they are responsible for is minimized.

A large choice for the positive term K plays another crucial role for the practical implementation of the CAO algorithm in multi-robot coverage applications: the problem with the performance index defined in (5) is that its second term $\int_{q\in\mathscr{T}-\mathscr{V}} dq$ *cannot be, in general, computed in practice*; as this term involves the part of the terrain that is not currently visible, its computation requires that the geometry this part is known or equivalently, as the invisible part changes with the evolution of the team's configuration, that the whole terrain is known. To overcome this problem, instead of minimizing (5) the following performance index is actually minimized by the CAO approach:

$$\begin{aligned}\bar{J}(\mathscr{P}) &= \int_{q\in\mathscr{V}} \min_{i\in\{1,\ldots,M\}} \left|x^{(i)} - q\right|^2 dq \\ &\quad + K\int_{(x,y,\phi(x,y))\in\mathscr{T}-\mathscr{V}} \mathscr{I}(x,y)dxdy \qquad (6)\end{aligned}$$

where $\mathscr{I}(q)$ denotes the indicator function that is equal to 1 if the point $(x,y,\phi(x,y))$ belongs to the invisible area of the terrain and is zero, otherwise. In other words, in the cost criterion $\bar{J}(\mathscr{P})$ and for the whole invisible area, the unknown terrain points $(x,y,\phi(x,y))$ are replaced by $(x,y,1)$, i.e. $\bar{J}(\mathscr{P})$ assumes that the whole invisible area is a flat subarea.

It is not difficult for someone to see that the replacement of the cost criterion (5) by the criterion (6) has a negligible implication in the team's performance: as a large choice for K corresponds to firstly making sure that the whole terrain is visible and then to minimizing the average distance between the robots and their responsible subareas, minimizing either of criteria (5) or (6) is essentially the same.

An efficient trajectory generation algorithm for optimal coverage – i.e. for minimization of the cost criteria (5) or (6) – must make sure that the physical constraints are also

met throughout the whole multi-robot coverage application. Such physical constraints include, but are not limited to, the following ones:

- The robots remain within the terrain's limits, i.e. they remain within $[x_{min}, x_{max}]$ and $[y_{min}, y_{max}]$ in the $x-$ and y-axes, respectively.
- The robots satisfy a maximum height requirement while they do not "hit" the terrain, i.e. they remain within $[\Phi(x,y)+d, z_{max}]$ along the z-axis, where d denotes the minimum safety distance (along the z-axis) the robots' should be from the terrain and z_{max} denotes the maximum allowable height for the robots.
- The robots do not come closer on to each other than a minimum allowable safety distance d_r.

It is not difficult for someone to see that all the above constraints can be easily cast in the form (3) and thus can be handled by the CAO algorithm.

III. EXPERIMENTAL RESULTS

To validate our approach in a realistic environment, we have used two different data sets which were collected with the use of a miniature quadrocopter specially designed for the needs of the European project sFLY (www.sfly.org). Our experimental platform is the quadrocopter Pelican [22] presented in Fig. 2, developed by Ascending Technologies [23]. The helicopter is driven by four rotors, symmetric to the center of mass. The control of the quadrocopter is done only by changing the rotation speed of the propellers and is described in more detail in [22]. The key features of this system are the payload of about 500 g, the flexible design enabling one to easily mount different payloads such as computer boards or cameras, and a flight autonomy of about twenty minutes. The helicopter features also a flight control unit (autopilot) for low-level data fusion and flight control. In particular, the fused data are body accelerations, body angular velocities, magnetic compass, and height measured by an air pressure sensor. As for the sensing, we equipped the helicopter with a Point-Grey USB Firefly camera with a resolution of 752x 480 pixels and global shutter. The optics provides a 150-degree field-of-view. The camera faces down since we expect the most stable, traceable features coming from the ground. The camera is used for our visual SLAM algorithm and provides real-time 6DoF pose estimates of the helicopter. For the visual SLAM and all computationally more expensive onboard tasks, we equipped the helicopter with a 1.6 GHz Intel Atom based embedded computer, also available from [23]. This computer is equipped with 1 GB RAM, a MicroSD card slot for the operating system, and a mini WiFi card (N standard) for remote access to the helicopter.

The scenarios tested consider a team of four MAVs and correspond into two different areas. The first area is Birmensdorf in Switzerland and it's presented in Fig. 3, while the second area corresponds to the ETHZ's hospital area and it's presented in Fig. 4. More details about the data and the methodology used to extract them, are presented in [16] and [18].

Fig. 2. The Pelican quadrocopter

Fig. 3. Outdoor flight path through the Birmensdorf area.

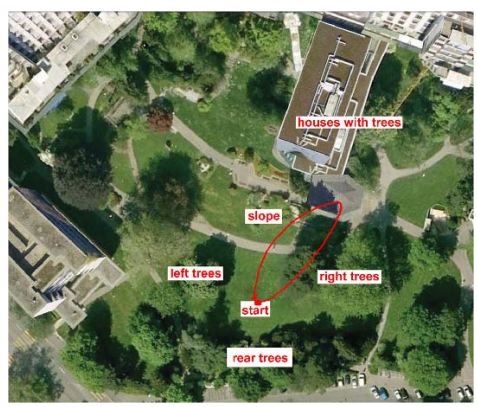

Fig. 4. Outdoor flight path through the ETHZ's hospital area.

The main constraints imposed to the robots are that they remain within the terrain's limits, i.e. within $[x_{min}, x_{max}]$ and $[y_{min}, y_{max}]$ in the $x-$ and $y-$ axes, respectively. At the same time they have to satisfy a maximum height requirement while they do not "hit" the terrain, i.e. they remain within $[\Phi(x,y)+d, z_{max}]$ along the z-axis. Several initial configurations for each scenario were tested. The values of the cost function for three different configurations, in the case of the Birmensdorf area are presented in Fig. 5. Sample trajectories for a robot team with initial coordinates for *Robot 1* $(1.34, 121.29, 22.91)$, for *Robot 2* $(2.69, 121.29, 22.91)$, for *Robot 3* $(4.04, 121.39, 22.91)$ and for *Robot 4* $(5.39, 121.29, 22.91)$ *(all units are in meters)*

are presented in Fig. 6, while in Fig. 7 the final positions of 3 robot teams starting from different initial positions are presented in a 3D view. Different marker type corresponds to different robots, while different color corresponds to a different team. The values of the cost function for three initial configurations in the case ETHZ's hospital area are presented in Fig. 8. Sample trajectories for a robot team with initial coordinates for *Robot 1* $(2.33, 95.57, 41.95)$, for *Robot 2* $(25.64, 97.90, 41.95)$, for *Robot 3* $(48.95, 100.23, 41.95)$ and for *Robot 4* $(72.26, 102.56, 41.95)$ *(all units are in meters)* are presented in Fig. 9. In Fig. 10 the final positions of 3 robot teams starting from different initial positions are presented in a 3D view.

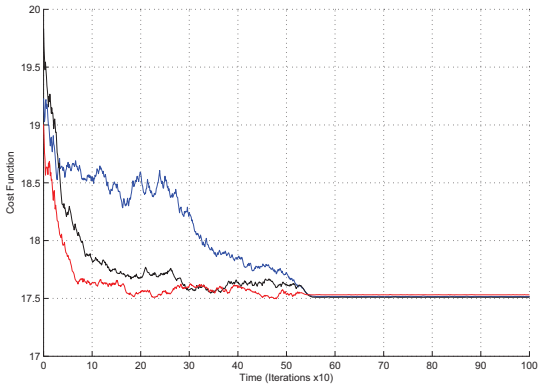

Fig. 5. Comparative cost functions for different initial robot team configurations in Birmensdorf area.

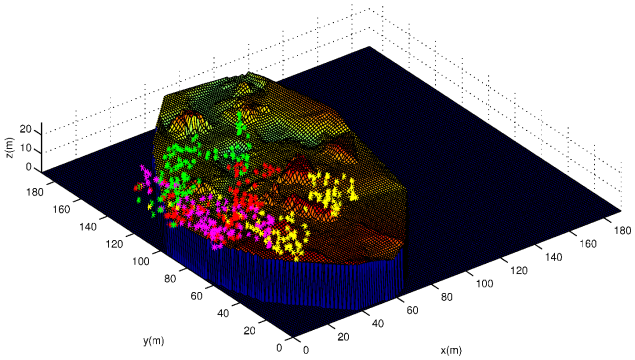

Fig. 6. 3D Path followed by a robot team in a coverage scenario in Birmensdorf area.

IV. CONCLUSIONS AND FUTURE WORK

A two-step centralized procedure to align a swarm of flying vehicles to perform surveillance coverage has been presented and formally analyzed. Initially a state-of-the-art visual-SLAM algorithm tracks the pose of the camera while, simultaneously, building an incremental map of the surrounding environment, which is used as an input in an optimization procedure. Based in a cognitive based methodology an optimal alignment is produced, which maximizes the area

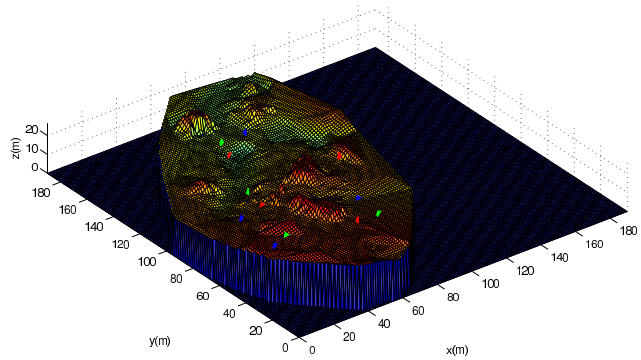

Fig. 7. Final configurations of three robot teams starting from different initial positions for the Birmensdorf area.

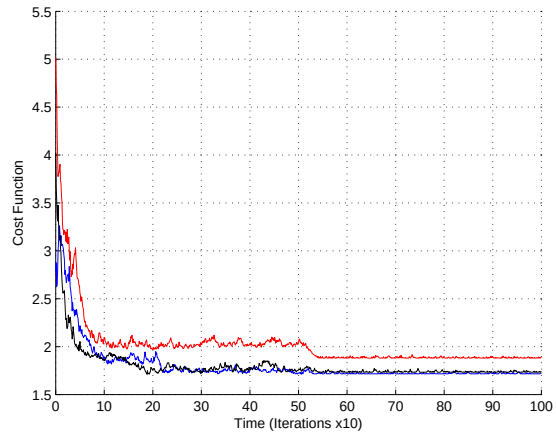

Fig. 8. Comparative cost functions for different initial robot team configurations in ETHZ's hospital area.

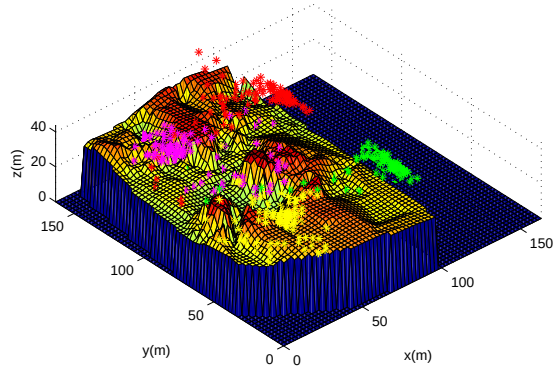

Fig. 9. 3D Path followed by a robot team in a coverage scenario in the ETHZ's hospital area.

monitored by the aerial robots. The proposed approach has the following key advantages with respect to previous works:

- it does not require any a priori knowledge on the

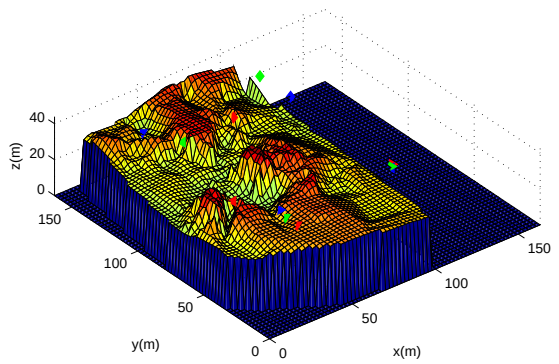

Fig. 10. Final configurations of three robot teams starting from different initial positions for the ETHZ's hospital area.

environment;

- it works in any given environment, without the necessity to make any kind of assumption about its topology;
- it can incorporate any kind of constraints;
- it does not require a knowledge about these constraints since they are learnt during the task execution;
- its complexity is low allowing real time implementations;
- it requires low weight and cost sensors, which makes it ideal for aerial robot applications.

The advantages of the proposed methodology make it suitable for real implementations and the results obtained through numerical simulations give us the motivation to adopt the CAO also in other frameworks. We are interested into formulating the same problem in a distributed manner by using different cost functions for each team. This approach is closer to real world applications since it will not depend into a centralized scheme with all the known disadvantages. Apart from that a decentralized approach will allow us to include communications constraints. We are also interested in incorporating more realistic constraints including sensor limitations. Furthermore, we expect that many important tasks in mobile robotics can be approached by CAO-based algorithms: for example coordinated exploration, optimal target tracking, multi-robot localization, etc. This is due to the fact that the CAO approach does not require an a priori knowledge of the environment and it has low complexity. Both these issues are fundamental in mobile robotics.

V. ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Communities Seventh Framework Programme (FP7/2007-2013) under grant agreement n. 231855 (sFly).

REFERENCES

[1] A. Renzaglia, L. Doitsidis, A. Martinelli and E.B. Kosmatopoulos, "Cognitive-based Adaptive Control for Cooperative Multi-Robot Coverage," *IEEE International Conference on Robotics and Intelligent System (IROS),* pp. 3314-3320, Taipei, Taiwan, 2010.

[2] A. Renzaglia, L. Doitsidis, A. Martinelli and E.B. Kosmatopoulos, "Adaptive-based, Scalable Design for Autonomous Multi-Robot Surveillance," *49th IEEE Conference on Decision and Control (CDC),* pp. 4618-4624, Atlanta, Georgia, USA, 2010.

[3] J. Cortés, S. Martínez, T. Karataş and F. Bullo, "Coverage Control for Mobile Sensing Networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243-255, 2004.

[4] M. Schwager, J. McLurkin and D. Rus, "Distributed Coverage Control with Sensory Feedback for Networked Robots," in *Proceedings of Robotics: Science and Systems,* Philadelphia, USA, 2006.

[5] L. Pimenta, V. Kumar, R.C Mesquita and G. Pereira, "Sensing and coverage for a network of heterogeneous robots," in *47th IEEE Conference on Decision and Control,* Cancun, Mexico, pp. 3947-3952, 2008.

[6] A. Ganguli, J. Cortés and F. Bullo, "Maximizing visibility in nonconvex polygons: nonsmooth analysis and gradient algorithm design," in *American Control Conference*, vol. 2, pp. 792-797, 2005.

[7] A. Ganguli, J. Cortés and F. Bullo, "Visibility-based multi-agent deployment in orthogonal environments," in *American Control Conference,* New York, USA, pp. 3426-3431, 2007.

[8] A. Breitenmoser, J. Metzger, R. Siegwart and D. Rus, "Distributed Coverage Control on Surfaces in 3D Space," in *IEEE International Conference on Robotics and Intelligent System (IROS),* Taipei, Taiwan, 2010.

[9] M. Schwager, B.J. Julian and D. Rus, "Optimal Coverage for Multiple Hovering Robots with Downward Facing Camera," in *IEEE International Conference on Robotics and Automation (ICRA),* Kobe, Japan, pp. 3515-3522, 2009.

[10] E.B. Kosmatopoulos, Papageorgiou, M., Vakouli, A. Kouvelas, A., "Adaptive fine-tuning of nonlinear control systems with application to the urban traffic control strategy TUC," *IEEE Transactions on Control Systems Technology,* Vol. 15, no. 6, pp. 991-1002, 2007.

[11] E.B. Kosmatopoulos, "An adaptive optimization scheme with satisfactory transient performance," *Automatica,* Vol. 45, No. 3, pp. 716-723, 2009.

[12] E.B. Kosmatopoulos and A. Kouvelas, "Large-Scale Nonlinear Control System Fine-Tuning through Learning," *IEEE Transactions Neural Networks,* Vol. 20, No. 6, pp. 1009-1023, 2009.

[13] H. Chen and Z. Xu, "3D Map Building Based on Stereo Vision," *IEEE International Conference on Networking, Sensing and Control, ICNSC,* 2006.

[14] S. Lacroix, I. Jung, A. Mallet, "Digital Elevation Map Building from Low Altitude Stereo Imagery," *Proc. Of the 9 th Int. Symposium on Intelligent Robotic Systems,* 2001.

[15] H. Strasdat, J.M.M. Montiel and A.J. Davison, "Real-Time Monocular SLAM: Why Filter?", *IEEE International Conference on Robotics and Automation (ICRA),* 2011.

[16] M. Bloesch, S. Weiss, D. Scaramuzza, and R. Siegwart, "Vision Based MAV Navigation in Unknown and Unstructured Environments", *IEEE International Conference on Robotics and Automation (ICRA)*, 2010.

[17] M. Achtelik, M.C. Achtelik, S. Weiss and R. Siegwart, Roland, "Onboard IMU and Monocular Vision Based Control for MAVs in Unknown In- and Outdoor Environments," *IEEE International Conference on Robotics and Automation (ICRA),* 2011.

[18] S. Weiss, M. Achtelik, L. Kneip D. Scaramuzza and R. Siegwart, "Intuitive 3D Maps for MAV Terrain Exploration and Obstacle Avoidance," *Journal of Intelligent & Robotic Systems,* vol. 61, pp. 473-493, 2011.

[19] G. Klein and D. Murray, "Parallel Tracking and Mapping for Small AR Workspaces," *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07),* 2007.

[20] S. Weiss and R. Siegwart, "Real-Time Metric State Estimation for Modular Vision-Inertial Systems," *IEEE International Conference on Robotics and Automation (ICRA),* 2011.

[21] A. Renzaglia, L. Doitsidis, A. Martinelli and E.B. Kosmatopoulos, "Multi-Robot 3D Coverage of Unknown Areas," *International Journal of Robotics Research*, (under review)

[22] D. Gurdan, J. Stumpf, M. Achtelik, K.-M. Doth, G. Hirzinger, D. Rus, "Energy-efficient autonomous four-rotor flying robot controlled at 1 kHz," *IEEE International Conference on Robotics and Automation (ICRA)*, pp.361-366,2007.

[23] Ascending Technologies GmbH, website, http://www.asctec.de.