Cooperative Control of a Heterogeneous Multi-Robot System based on Relative Localization

Marco Cognetti, Giuseppe Oriolo, Pietro Peliti, Lorenzo Rosa, Paolo Stegagno

***Abstract*— We propose a cooperative control scheme for a heterogeneous multi-robot system, consisting of an Unmanned Aerial Vehicle (UAV) equipped with a camera and multiple identical Unmanned Ground Vehicles (UGVs). Our control scheme takes advantage of the different capabilities of the robots. Since the system is highly redundant, the execution of multiple different tasks is possible. The primary task is aimed at keeping the UGVs well inside the camera field of view, so as to allow our localization system to reconstruct the identity and relative pose of each UGV with respect to the UAV. Additional tasks include formation control, navigation and obstacle avoidance. We thoroughly discuss the feasibility of each task, proving convergence when possible. Simulation results are presented to validate the proposed method.**

I. INTRODUCTION

Heterogeneous multi-robot systems are appealing due to their high versatility. In fact, by taking advantage of the individual abilities of the robots, one may address multiple tasks, possibly of different nature.

The first step to achieve a coordinate motion of several agents, is to recognize and localize the team mates. In addition, the fact that the robots may collide with objects in the environment or among themselves must be taken into account. Once those issues are solved, the system is ready to perform advanced tasks, like controlling the formation shape or navigating in unknown environments.

Here, we suppose that one of the members of the multi-robot team is a Vertical Take Off and Landing Unmanned Aerial Vehicle (VTOL-UAV), equipped with a downlooking monocular camera and a range-finder sensor. The other team members are Unmanned Ground Vehicles (UGVs), capable of rough odometric localization.

The UAV is able to provide a comprehensive view of the environment, e.g. for mapping or sensing purposes, or to discover obstacles or dangerous zones, thus providing useful data to the ground robots. The hovering capabilities of the VTOL-UAV can be further exploited for sensing and monitoring the relative positions of the team members. In addition, if the ground robots move slow enough, the UAV can easily reconfigure its position to keep them inside the camera Field Of View (FOV).

We assume the absence of any kind of tagging, i.e. each UGV is identical. In real scenarios, robot may not be identifiable via vision only, for example when the ground robots are all similar, or when bad-lighting or fog do not allow to properly distinguish them.

M. Cognetti, G. Oriolo, P. Peliti and L. Rosa are with the Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza University of Rome, Via Ariosto 25, 00185 Rome, Italy. E-mail: {cognetti,oriolo,peliti,rosa}@diag.uniroma1.it. P. Stegagno is with the Max Planck Institute for Biological Cybernetics, Tübingen, Germany. E-mail: paolo.stegagno@tuebingen.mpg.de

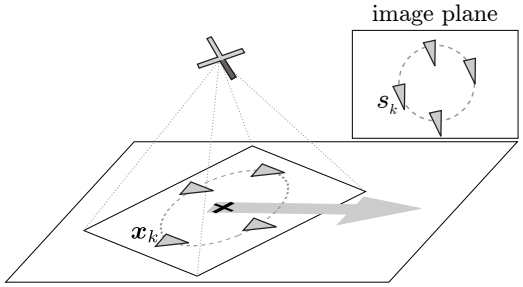

Fig. 1. Overview of the heterogeneous multi-robot system.

The idea of using a system composed by aerial and ground vehicles is not new in literature. As example, the coordination of the heterogeneous system is used to localize fixed [1] and moving [2], [3] targets on the ground. In [4], the idea that a multi-robot system can be modeled and controlled as a redundant robotic system was introduced, and a solution based on the task-priority paradigm was proposed in [5] to perform different control actions simultaneously. The same concept was used in [6] to design specific tasks for a platoon of autonomous vehicles, which was successfully tested in a real scenario in [7]. The specific problem of keeping multiple point inside a camera FOV has already been addressed. In [8], a switching approach is used to control an actuated camera by alternating a position based visual servoing with a backward motion which prevents tracked points from leaving the image plane. In [9], the same problem is addressed for multiple moving targets using a task function approach. Tracked targets are not cooperative, thus the actuated camera is in charge of the whole task. A formal proof of the feasibility of the approach is given and experimental results prove the robustness to unknown disturbances. A task-priority approach has been proposed in [10] to control a team of UAVs equipped with gimbaled cameras, while simultaneously taking into account the distance between the vehicles.

The above mentioned solutions are not feasible when the knowledge of the robot identities is missing, since the computed control inputs can not be properly mapped.

In [11], the authors proposed a control scheme for a mixed UAV-UGV team in which one of the robots is specially equipped to navigate in the environment and acts as a leader. By using an accurate relative localization system [12], all the other team mates are able to follow the leader. Obstacle avoidance is also performed by using the convex hull of the formation to obtain a collective motion. Even in this case,

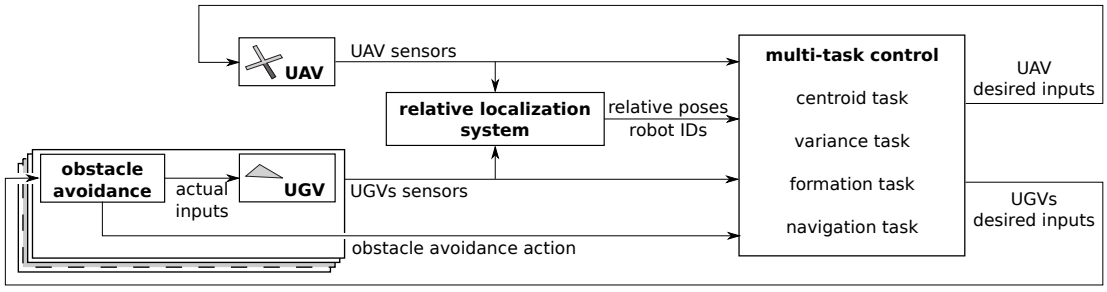

Fig. 2. General control scheme for the heterogeneous multi-robot system.

the robot IDs are supposed to be known.

In our previous paper [13] we presented a relative localization system for a single UAV–multiple UGVs team of robots. A multi-target filtering technique is used to fuse proprioceptive and exteroceptive sensor data, to reconstruct the robot relative poses and associate them to the correct identities. There, we also reported some preliminary results of a multi-task control scheme.

Here we deepen the analysis of the same heterogeneous team considered in [13] and we extend the control design to build a fully autonomous system. We propose a multi-task control framework that takes advantage of the high redundancy of the system to achieve an image-based control aimed to keep the robots close to each other, and a formation control for the UGVs. Basic navigation in the environment and obstacle avoidance are also considered. Moreover, we highlight the connection between the relative localization system and the control algorithm, which are strictly related.

Figure 1 represents a sketch of the system and the tasks described above, in order to give an overview of the heterogeneous team and the desired behavior.

The paper is organized as follows. In Sects. II and III we characterize the heterogeneous team and the kinematic models used to describe the motion of the robots. Section IV develops the design of the control laws for the visual tasks, which are needed to perform the relative localization. Section V is devoted to the design of additional tasks, such as formation control, navigation and obstacle avoidance. Section VI quickly summarizes the localization system previously presented in [13]. Simulation results are reported in Section VII. Section VIII concludes the paper.

II. OVERVIEW OF THE APPROACH

Our team is composed by an aerial vehicle with hovering capabilities and several ground robots.

In the following we use the kinematic model of an underactuated small-sized VTOL-UAV, such as a small quadrotor vehicle. Note that the proposed approach is general, and can be easily modified to consider other kinds of aircraft. Nowadays sensory equipment for aerial vehicles commonly includes an Inertial Measurement Unit (IMU), which provides measurements of the UAV attitude, angular velocities and linear accelerations. A camera is rigidly attached to the UAV, and we also suppose to have a range-finder sensor (e.g. a sonar), which provides measurements of the vehicle height.

Suppose to have N ground robots, equipped with odometry sensors, that provide rough localization capabilities. Additional exteroceptive sensors may be used for local navigation purposes, and in the following we suppose that the UGVs are equipped with a classical obstacle-avoidance algorithm that ensures collision-free motion.

We consider two main tasks that are purely visual based: the control of the centroid and variance of image features associated to the ground robots. We also consider two additional tasks: a formation control for the ground robots, and a coordinate motion which provides basic navigation capabilities to the heterogeneous team.

Figure 2 depicts the proposed framework. Sensor data are collected on-board the UAV and the UGVs and are fed to a relative localization system, which provides estimates of the relative poses between the robots, and associates to each of them an identity. All the data are used to compute the control laws needed to accomplish the desired tasks mentioned above (and thoroughly described in the rest of the paper). The UGVs are equipped with an on-board inner–loop obstacle avoidance control which modifies their control inputs on the basis of local measurements. This constitutes a perturbation of the previously computed control action. To cope with this problem, the obstacle avoidance action is used to build a feedforward compensation term in the multi-task control.

III. SYSTEM MODELING

In this section we introduce kinematic models which can be used to describe the motion of the UAV and the UGVs.

We consider a reference frame rigidly attached to the center of mass of the UAV, and we use it to describe the robot motion with respect to (w.r.t) an inertial frame. Similarly, a reference frame can be attached in the camera focus, to describe the motion of the camera. Without any loss of generality, we suppose that the camera frame and UAV body-fixed frame coincide, since any rigid transformation between them can be considered for the general case. Given that, the kinematic model of the UAV also describes the motion of the camera. A generic simplified description of the kinematics of a small-sized underactuated VTOL aircraft is given by [14]

$$\begin{aligned} \dot{\boldsymbol{x}}_c &= \boldsymbol{R}_c \boldsymbol{v}_c \\ \dot{\boldsymbol{R}}_c &= \boldsymbol{R}_c \, \boldsymbol{S}(\boldsymbol{\omega}_c) \,, \end{aligned} \tag{1}$$

where $\boldsymbol{S}(\cdot)$ denotes the skew-symmetric matrix expressing the cross product operation, $\boldsymbol{x}_c \in \mathbb{R}^3$ is the camera position in the inertial frame, $\boldsymbol{R}_c$ is the rotation matrix expressing

the orientation of the VTOL body fixed frame with respect to the world inertial frame, $\boldsymbol{v}_c \in \mathbb{R}^3$ and $\boldsymbol{\omega}_c \in \mathbb{R}^3$ are the translational and rotational velocities respectively, both expressed in the body-fixed frame.

When dealing with this sort of vehicles it is possible to separate the analysis of the translational and rotational motion, a common practice [15] which can be fully justified by means of control theory arguments, such as the time scale separation principle [16]. Due to the underactuation of the vehicle, the rotational motion cannot be assigned arbitrarily if a desired translational motion is required. For this reason, in the following we suppose that the Cartesian velocity $\boldsymbol{v}_c$ is an available control input for the UAV, and we consider the rotational velocities $\boldsymbol{\omega}_c$ of the UAV as a drift term.

The motion of the generic k-th UGV takes place on the ground plane. Denote with $\boldsymbol{x}_k \in \mathbb{R}^3$ the position in the inertial frame of a representative point on the robot body, and with $\boldsymbol{R}_k$ the rotation matrix describing the robot orientation w.r.t. the world frame. Here we suppose that an on-board controller is able to drive the representative point with any desired Cartesian velocity, and that each ground robot projects on the image a *point feature* [17] corresponding to its representative point.

For example, for differentially-driven wheeled mobile robots, it is possible to assign an arbitrary velocity to any point of the robot, with the exception of the mid point between the wheels. The control inputs that realize such velocity are easily computed by input–output linearization [18].

The apparent motion on the image of the point feature associated to the k-th ground robot is influenced by the motion of both camera and ground robot. The kinematic model which describes this apparent motion is particularly important for us, since many parts of the control scheme we propose in this paper are based on direct visual feedback.

Here, we model the camera sensor using the *planar perspective* projection [17]. Denote the projection of the point $\boldsymbol{x}_k$ on the image plane with $\boldsymbol{s}_k = (s_{k1}, s_{k2})^T$, and let $\hat{\boldsymbol{s}}_k = (s_{k1}, s_{k2}, 1)^T$ be the vector of *homogeneous* coordinates obtained from $\boldsymbol{s}_k$. Denote with $\boldsymbol{x}_k^c$ the vector representing $\boldsymbol{x}_k$ in the camera frame. The planar perspective projection for a *calibrated* camera gives

$$\hat{\boldsymbol{s}}_k = \frac{\boldsymbol{x}_k^c}{z_k}\,, \tag{2}$$

where z_k is the depth associated to $\boldsymbol{s}_k$, i.e. the third coordinate of $\boldsymbol{x}_k^c$. The relation between $\boldsymbol{x}_k$ and $\boldsymbol{x}_k^c$ is

$$\boldsymbol{x}_k = \boldsymbol{x}_c + \boldsymbol{R}_c \boldsymbol{x}_k^c\,. \tag{3}$$

Using eq. (1), its derivative can be rearranged for $\dot{\boldsymbol{x}}_k^c$, obtaining

$$\dot{\boldsymbol{x}}_k^c = -\boldsymbol{R}_c^T \boldsymbol{v}_c - \boldsymbol{S}(\boldsymbol{\omega}_c)\boldsymbol{x}_k^c + \boldsymbol{R}_c^T\, \hat{\boldsymbol{v}}_{g,k}\,, \tag{4}$$

where $\boldsymbol{v}_c = \dot{\boldsymbol{x}}_c$ and $\boldsymbol{v}_{g,k} = \dot{\boldsymbol{x}}_k$ All the UGVs lie on a planar surface, hence the depth of each feature is geometrically related to the camera height, and can be easily reconstructed if estimates of the UAV height and attitude are available. Furthermore, the derivative of the depth z_k is directly related to the velocity of the UAV. To show this fact, consider that $z_k = \boldsymbol{e}_3^T \boldsymbol{x}_k^c$, where $\boldsymbol{e}_3 = (0\,0\,1)^T$. Its derivative is

$$\dot{z}_k = -\boldsymbol{e}_3^T \boldsymbol{R}_c^T \boldsymbol{v}_c - \boldsymbol{e}_3^T \boldsymbol{S}(\boldsymbol{\omega}_c)\boldsymbol{x}_k^c + \boldsymbol{e}_3^T \boldsymbol{R}_c^T\, \hat{\boldsymbol{v}}_{g,k}. \tag{5}$$

Using eqs. (4–5), the time derivative of eq. (2) becomes

$$\dot{\hat{\boldsymbol{s}}}_k = \hat{\boldsymbol{J}}_k^c \boldsymbol{v}_c + \hat{\boldsymbol{J}}_k^\omega\, \boldsymbol{\omega}_c + \hat{\boldsymbol{J}}_k^g \hat{\boldsymbol{v}}_{g,k}\,, \tag{6}$$

where the Jacobians $\hat{\boldsymbol{J}}_k^c$, $\hat{\boldsymbol{J}}_k^\omega$, $\hat{\boldsymbol{J}}_k^g$, are related to the camera translational and rotational motion, and to the ground robot motion, respectively. In particular,

$$\hat{\boldsymbol{J}}_k^c = -\frac{1}{z_k}\left(\boldsymbol{I} - \hat{\boldsymbol{s}}_k \boldsymbol{e}_3^T\right)\boldsymbol{R}_c^T$$
$$\hat{\boldsymbol{J}}_k^g = \frac{1}{z_k}\left(\boldsymbol{I} - \hat{\boldsymbol{s}}_k \boldsymbol{e}_3^T\right)\boldsymbol{R}_c^T$$

are related to the translational motion, while

$$\hat{\boldsymbol{J}}_k^\omega = \left(\boldsymbol{I} - \boldsymbol{s}_k \boldsymbol{e}_3^T\right)\boldsymbol{S}(\hat{\boldsymbol{s}}_k)$$

The third component of each vector $\hat{\boldsymbol{s}}_k$ is the always fixed homogeneous coordinate. Hence only the dynamics of $\boldsymbol{s}_k$ are relevant for the system. Thus we neglect the last row in eq. (6). As a consequence, the last row of the matrices $\hat{\boldsymbol{J}}_k^c$ and $\hat{\boldsymbol{J}}_k^\omega$ can be removed, and we denote the corresponding new matrices of dimensions 2×3 with $\boldsymbol{J}_k^c$ and $\boldsymbol{J}_k^\omega$. Analogously we can remove the last row of the matrix $\hat{\boldsymbol{J}}_k^g$, and also its third column, since the UGV motion is confined on the ground plane. Let $\boldsymbol{J}_k^g$ be the corresponding new 2×2 Jacobian matrix for the k-th ground robot.

Denoting by $\boldsymbol{v}_{g,k}$ the vector embedding the first two components of $\hat{\boldsymbol{v}}_{g,k}$, one can rewrite eq. (6) as

$$\dot{\boldsymbol{s}}_k = \boldsymbol{J}_k^\omega\, \boldsymbol{\omega}_c + \begin{pmatrix} \boldsymbol{J}_k^c & \boldsymbol{J}_k^g \end{pmatrix} \begin{pmatrix} \boldsymbol{v}_c \\ \boldsymbol{v}_{g,k} \end{pmatrix}. \tag{7}$$

Note that the right hand side of above equation is the sum of two terms. The first is a perturbation due to the rotational velocity of the camera, related to the underactuation of the VTOL-UAV. The second term is related to the Cartesian velocities of the robots, which are treated as control inputs.

IV. PRIMARY TASK CONTROL

Our first objective is to regulate a composite output that consists of the centroid and the variance of the image features corresponding to the UGVs. In fact, by keeping this centroid close to the image center and controlling the dispersion of the features around it, we can keep all the UGVs in the field of view of the UAV camera. This is obviously the primary task for our system, because it guarantees that our relative localization system [13] will provide a position estimate for each UGV throughout the mission.

The centroid of the image features corresponding to the UGVs is $\boldsymbol{\sigma} = \sum_{k=1}^N \boldsymbol{s}_k / N$, so that

$$\dot{\boldsymbol{\sigma}} = \sum_{k=1}^N \boldsymbol{J}_k^\omega\, \boldsymbol{\omega}_c + \frac{1}{N}\sum_{k=1}^N \begin{pmatrix} \boldsymbol{J}_k^c & \boldsymbol{J}_k^g \end{pmatrix} \begin{pmatrix} \boldsymbol{v}_c \\ \boldsymbol{v}_{g,k} \end{pmatrix} = \boldsymbol{J}_\sigma^\omega\, \boldsymbol{\omega}_c + \boldsymbol{J}_\sigma^v\, \boldsymbol{v}, \tag{8}$$

where $\boldsymbol{v} = (\boldsymbol{v}_c \;\; \boldsymbol{v}_{g,1} \;\; \ldots \;\; \boldsymbol{v}_{g,N})^T = (\boldsymbol{v}_c \;\; \boldsymbol{v}_g)^T$ are the Cartesian velocities of the robots, i.e., the available control

inputs. Denoting by $\boldsymbol{\sigma}_d$ the constant desired position for the centroid and by $\boldsymbol{e}_\sigma = \boldsymbol{\sigma}_d - \boldsymbol{\sigma}$ the corresponding vector error, we have

$$\dot{\boldsymbol{e}}_\sigma = -\boldsymbol{J}_\sigma^v \boldsymbol{v} - \boldsymbol{J}_\sigma^\omega \boldsymbol{\omega}_c \,. \tag{9}$$

The variance of the image features around their centroid is instead expressed as $\gamma = \sum_{k=1}^N (\boldsymbol{s}_k - \boldsymbol{\sigma})^T (\boldsymbol{s}_k - \boldsymbol{\sigma})/2N$, so that

$$\begin{aligned}\dot{\gamma} &= \frac{1}{N} \sum_{k=1}^N (\boldsymbol{s}_k - \boldsymbol{\sigma})^T \left(\boldsymbol{J}_k^\omega \boldsymbol{\omega}_c + (\boldsymbol{J}_k^c \quad \boldsymbol{J}_k^g) \begin{pmatrix} \boldsymbol{v}_c \\ \boldsymbol{v}_{g,k} \end{pmatrix} \right) \\ &= \boldsymbol{J}_\gamma^\omega \boldsymbol{\omega}_c + \boldsymbol{J}_\gamma^v \boldsymbol{v} \,. \end{aligned} \tag{10}$$

having used $\sum_{k=1}^N (\boldsymbol{s}_k - \boldsymbol{\sigma})^T \dot{\boldsymbol{\sigma}} = 0$. Let $e_\gamma = \gamma_d - \gamma$ be the scalar error w.r.t. the desired value γ_d for the variance. Using eq. (10), we obtain

$$\dot{e}_\gamma = -\boldsymbol{J}_\gamma^v \boldsymbol{v} - \boldsymbol{J}_\gamma^\omega \boldsymbol{\omega}_c \,. \tag{11}$$

The overall error dynamics is then

$$\begin{pmatrix} \dot{\boldsymbol{e}}_\sigma \\ \dot{e}_\gamma \end{pmatrix} = -\boldsymbol{J}\, \boldsymbol{v} - \begin{pmatrix} \boldsymbol{J}_\sigma^\omega \\ \boldsymbol{J}_\gamma^\omega \end{pmatrix} \boldsymbol{\omega}_c, \quad \text{with } \boldsymbol{J} = \begin{pmatrix} \boldsymbol{J}_\sigma^v \\ \boldsymbol{J}_\gamma^v \end{pmatrix}. \tag{12}$$

Choose the control input $\boldsymbol{v}$ as

$$\boldsymbol{v} = \boldsymbol{G} \left(\begin{pmatrix} \boldsymbol{K}_\sigma & 0 \\ 0 & K_\gamma \end{pmatrix} \begin{pmatrix} \boldsymbol{e}_\sigma \\ e_\gamma \end{pmatrix} - \begin{pmatrix} \boldsymbol{J}_\sigma^\omega \\ \boldsymbol{J}_\gamma^\omega \end{pmatrix} \boldsymbol{\omega}_c \right) = \boldsymbol{G}\, \boldsymbol{f}, \tag{13}$$

where $\boldsymbol{G}$ is a right inverse of $\boldsymbol{J}$ and $\boldsymbol{K}_\sigma$, K_γ are positive-definite gains. We have the following result.

Proposition 1: The error dynamics under control law (13) is decoupled and globally exponentially stable.

Proof: Partition matrix $\boldsymbol{J}$ as $\boldsymbol{J} = (\boldsymbol{J}^c\ \boldsymbol{J}^g)$, where $\boldsymbol{J}^c$ is the 3×3 submatrix relative to the UAV velocity and $\boldsymbol{J}^g$ is the $3 \times 2N$ submatrix relative to the UGVs velocities. The determinant of $\boldsymbol{J}^c$ is easily computed as $\sum_{k=1}^N \|\boldsymbol{s}_k - \boldsymbol{\sigma}\|^2$. Since the UGVs are on the ground plane and the UAV is in flight above them, it is always $s_k \neq s_j, \forall k \neq j$. As a consequence, the above determinant is always positive, and matrix $\boldsymbol{J}$ is certainly full row rank. In turn, this means that $\boldsymbol{G}\boldsymbol{J} = \boldsymbol{I}$, and the closed-loop error dynamics become

$$\begin{pmatrix} \dot{\boldsymbol{e}}_\sigma \\ \dot{e}_\gamma \end{pmatrix} = - \begin{pmatrix} \boldsymbol{K}_\sigma & 0 \\ 0 & K_\gamma \end{pmatrix} \begin{pmatrix} \boldsymbol{e}_\sigma \\ e_\gamma \end{pmatrix}, \tag{14}$$

which proves the thesis. ∎

A possible choice of $\boldsymbol{G}$ in eq. (13) is $\boldsymbol{J}^\dagger = \boldsymbol{J}^T (\boldsymbol{J}\boldsymbol{J}^T)^{-1}$, i.e., the pseudoinverse of $\boldsymbol{J}$. This distributes the control effort among all the robots (the UAV and the UGV) in such a way that its norm is minimized, realizing a full cooperation among the UAV and the UGVs in executing the task. However, the previous proof shows that the antithetical choice is also feasible: i.e., one may set

$$\boldsymbol{G} = \begin{pmatrix} (\boldsymbol{J}^c)^{-1} \\ \boldsymbol{0} \end{pmatrix}, \tag{15}$$

corresponding to the UAV taking full responsibility of the primary task, while the UGVs are free to pursue other tasks.

V. ADDITIONAL TASKS CONTROL

The redundancy of the multi-robot system makes it possible to consider other tasks in addition to the primary objective; in particular, we will assume that such tasks concern the UGVs only. Denote by $\boldsymbol{e}_a$ the error in the execution of these additional tasks. The corresponding error dynamics are $\dot{\boldsymbol{e}}_a = \boldsymbol{J}_a \boldsymbol{v}_g$, where $\boldsymbol{J}_a$ is the corresponding Jacobian matrix. If $\boldsymbol{J}_a$ is full row rank, the control input

$$\tilde{\boldsymbol{v}}_g = -\boldsymbol{J}_a^\dagger \boldsymbol{K}_a \boldsymbol{e}_a, \tag{16}$$

where $\boldsymbol{K}_a$ would by itself ensure exponential convergence of $\boldsymbol{e}_a$ to zero.

Inclusion of the additional control term $\tilde{\boldsymbol{v}}_g$ in the UGV velocities in our proposed control scheme generates a perturbation of the primary task error dynamics. Therefore, we shall add a compensation velocity $\tilde{\boldsymbol{v}}_c$ to the UAV velocity as well. Equation (14) becomes

$$\begin{pmatrix} \dot{\boldsymbol{e}}_\sigma \\ \dot{e}_\gamma \end{pmatrix} = - \begin{pmatrix} \boldsymbol{K}_\sigma & 0 \\ 0 & K_\gamma \end{pmatrix} \begin{pmatrix} \boldsymbol{e}_\sigma \\ e_\gamma \end{pmatrix} + \boldsymbol{J}^g \tilde{\boldsymbol{v}}_g + \boldsymbol{J}^c \tilde{\boldsymbol{v}}_c, \tag{17}$$

and its exponential stability can be reserved by letting

$$\tilde{\boldsymbol{v}}_c = (\boldsymbol{J}^c)^{-1}\, \tilde{\boldsymbol{v}}_g = (\boldsymbol{J}^c)^{-1} \boldsymbol{J}^g \boldsymbol{J}_a^\dagger \boldsymbol{K}_a \boldsymbol{e}_a. \tag{18}$$

Based on this, we can formulate two different schemes for controlling both the primary and the additional tasks.

The first is obtained by choosing the right inverse $\boldsymbol{G}$ as in eq. (15). Since the execution of the primary task is delegated to the UAV, the UGV motion can be fully specified as in eq. (16), thus guaranteeing execution of the additional tasks as well (as long as $\boldsymbol{J}_a$ is full row rank). The resulting scheme (*Scheme 1*) is

$$\boldsymbol{v} = \begin{pmatrix} \boldsymbol{v}_c \\ \boldsymbol{v}_g \end{pmatrix} = \begin{pmatrix} (\boldsymbol{J}^c)^{-1} \boldsymbol{f} + \tilde{\boldsymbol{v}}_c \\ \tilde{\boldsymbol{v}}_g \end{pmatrix},$$

(note the definition of $\boldsymbol{f}$ in eq. (13)). This scheme may require a high control effort for the UAV since the UGVs are not cooperating to the primary task.

The second scheme retains the cooperative control action of the UGVs. This is achieved by choosing the right inverse $\boldsymbol{G}$ as the pseudoinverse $\boldsymbol{J}^\dagger$. We obtain (*Scheme 2*)

$$\boldsymbol{v} = \begin{pmatrix} \boldsymbol{v}_c \\ \boldsymbol{v}_g \end{pmatrix} = \boldsymbol{J}^\dagger \boldsymbol{f} + \begin{pmatrix} \tilde{\boldsymbol{v}}_c \\ \tilde{\boldsymbol{v}}_g \end{pmatrix},$$

This scheme will typically distribute the control effort more evenly between the UAV and the UGVs. However, the additional task dynamics is no more guaranteed to be stable because the UGV velocity also include the primary control action.

In the remainder of this section, we consider two specific additional tasks: formation control and navigation.

A. Formation control

The first additional task we introduce is the control of the formation of the UGVs. Clearly, some particular formations are more compatible than others with the primary task (image centroid and variance) of Sect. IV.

One possible choice is to place the UGVs along a circumference of a given radius, centered at their centroid. A suitable definition of the error for this task is

$$\boldsymbol{e}_\phi = \frac{1}{2}\begin{pmatrix} R^2 \\ \vdots \\ R^2 \end{pmatrix} - \frac{1}{2}\begin{pmatrix} (\boldsymbol{x}_1^c - \boldsymbol{\mu})^T(\boldsymbol{x}_1^c - \boldsymbol{\mu}) \\ \vdots \\ (\boldsymbol{x}_N^c - \boldsymbol{\mu})^T(\boldsymbol{x}_N^c - \boldsymbol{\mu}) \end{pmatrix},$$

being $\boldsymbol{\mu}$ the centroid of the UGVs expressed w.r.t. the camera frame. Note that the computation of $\boldsymbol{e}_\phi$ only requires relative UGV positions, which are provided by our localization system (see Sect. VI).

The associated error dynamics is $\dot{\boldsymbol{e}}_\phi = \boldsymbol{J}_\phi \boldsymbol{v}_g$, where $\boldsymbol{J}_\phi$ is the corresponding Jacobian matrix. The additional control term (16) for the UGVs becomes then

$$\tilde{\boldsymbol{v}}_g = -\boldsymbol{J}_\phi^\dagger \boldsymbol{K}_\phi \boldsymbol{e}_\phi \,. \tag{19}$$

Plugging this expression in eq. (18) one is ready to implement both Scheme 1 and Scheme 2. Note the following fact.

Proposition 2: When the UAV is hovering above the UGVs, the additional UGV control term (19) does not perturb the centroid error dynamics.

Proof: First note that if the UAV is hovering above the UGVs, all the image feature depths z_k are equal to the UAV height. Consider now the part of eq. (17) related to the centroid, and recall eq. (8). Use of (19) gives

$$\dot{\boldsymbol{e}}_\sigma = -\boldsymbol{K}_\sigma \boldsymbol{e}_\sigma + \frac{1}{N}\boldsymbol{J}_\sigma^v \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{J}_\phi^\dagger \boldsymbol{K}_\phi \boldsymbol{e}_\phi \end{pmatrix} = -\boldsymbol{K}_\sigma \boldsymbol{e}_\sigma,$$

since

$$\boldsymbol{J}_\sigma^v \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{J}_\phi^\dagger \end{pmatrix} = \boldsymbol{0}.$$

under the hovering assumption. ■

The above proposition is relevant for both control schemes, as it entails that the compensation term $\tilde{\boldsymbol{v}}_c$ will only need to cancel the perturbation on the variance error dynamics, ultimately resulting in less control effort for the UAV.

B. Navigation

A translational motion of the whole multi-robot system in a certain planar direction can be easily obtained by adding the same inertial velocity $\boldsymbol{v}_\mathcal{I}$ to the Cartesian velocity of each robot (for the UAV, only to the horizontal Cartesian velocity).

None of the the previous tasks are influenced by this additional motion, since their dynamics are determined only by the *relative* motion between the robots. It is easy to prove this fact by using the modified velocities in eqs. (9) and (11) and showing that the error dynamics is unchanged.

C. Decentralized obstacle avoidance

It is reasonable to assume that each UGVs can perform obstacle avoidance. For example, artificial potential fields can be used to generate additional velocities that push the UGV away from the obstacle.

To include this further action in Schemes 1 and 2, one may simply add the repulsive velocities to the UGV control input, and devise another UAV compensation term similar in structure to (18). This will still guarantee execution of the primary task, while the formation and navigation tasks will obviously be perturbed. However, since obstacle avoidance is typically a transient event, steady-state conditions will promptly be recovered as soon as it is concluded.

Other possible strategies are *(i)* override any other UGV command and use only the obstacle avoidance action, or *(ii)* use a task-priority scheme by considering obstacle avoidance as the main task and projecting all other UGV control terms in its null space.

VI. RELATIVE LOCALIZATION SYSTEM

The relative localization system used in our approach is described in detail in [13]. In addition to estimating the relative poses of the UGVs w.r.t. the camera frame, this system is able to reconstruct the identities of the UGVs, that are needed to implement the individual task control actions specified by the above schemes. Such an information would not be provided by a typical multi-tracking algorithm.

Another key feature of our localization method is robustness to false positives and false negatives. In fact, a multi-tracking algorithm might extract some features (objects, people, ...) that do not correspond to UGVs. This would obviously negatively affect the computation of the task errors (e.g., the centroid error) and ultimately the performance of the control scheme. The same would happen in the presence of false negatives, i.e., UGVs that are not detected.

Finally, the reconstruction of the UGV relative pose (position and *orientation*) is essential for implementing the proposed approach on nonholonomic ground robots. In fact, the feedback linearization controller (static or dynamic) used to transform Cartesian velocity inputs into actual wheel commands requires the knowledge of the robot orientation.

On the other hand, the proposed control approach helps the localization system. In fact, the primary tasks are designed to keep the UGVs inside the field of view of the camera, thus providing more data to the localization system. As a consequence, the accuracy of the estimates increases and more precise control actions are computed.

VII. SIMULATION RESULTS

To test the effectiveness of the proposed control scheme, we built a simulation framework. The UAV is a quadrotor, simulated including rotational dynamics, aerodynamic disturbances, and rotor dynamics. The UAV sensor equipment includes: an ultrasonic rangefinder with a quantization step of 2.56 cm; a monocular camera with an image plane of 320×240 pixels, a focal length of 272.94 pixels, and a frame rate of 30 Hz; an IMU sensor, simulated by adding

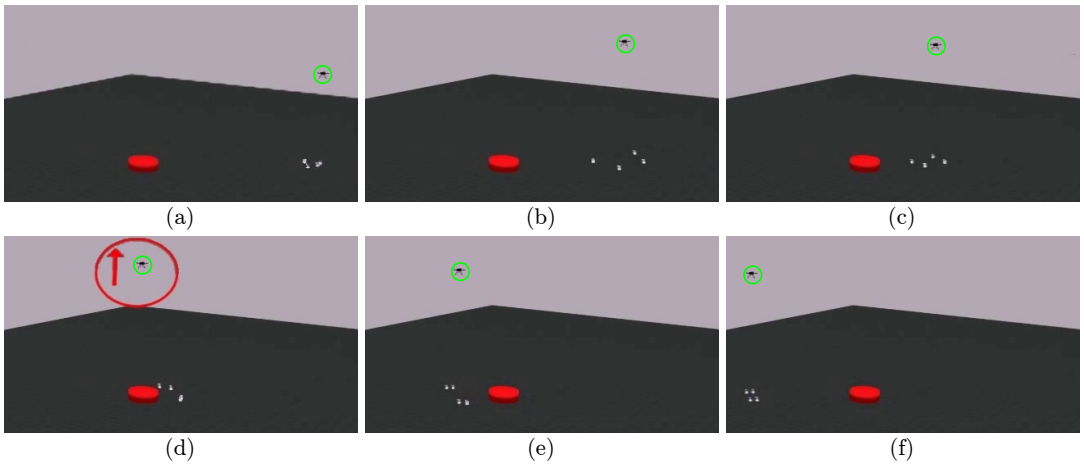

Fig. 3. Snapshots taken from the accompanying video to show the evolution of the system. The red circle with the arrow highlights the phase in which the UAV compensates the obstacle avoidance control inputs by raising its height.

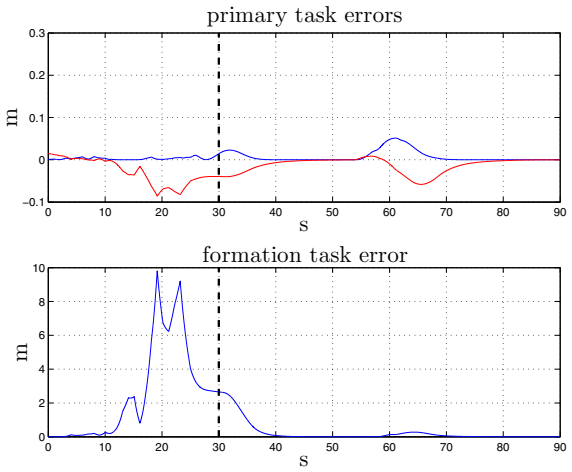

Fig. 4. Centroid and variance (top) and formation error (bottom) for known UGV identities.

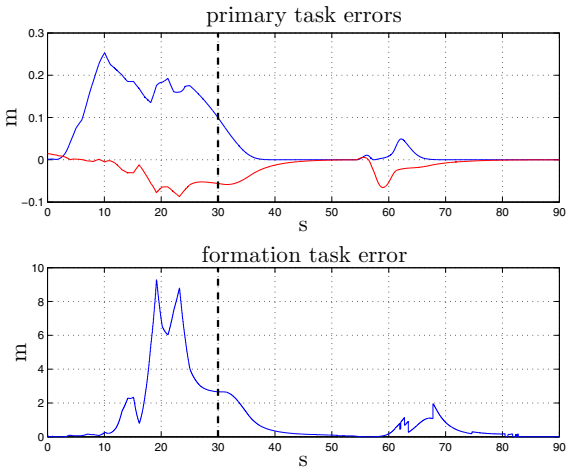

Fig. 5. Centroid and variance (top) and formation error (bottom) for reconstructed UGV identities

white Gaussian noise (zero mean, variance $5°$) to the attitude angles. The high-level control inputs for the UAV are thrust and attitude angles, as common for many vehicles of this class. The low-level control layer of the UAV includes an altitude controller, based on sonar measurements, and an attitude controller that uses the IMU data. Attitude references are generated by inverting the dynamics (1) of the UAV, and used to realize the commanded velocity inputs. The UGVs are differential drive robots equipped with wheel encoders affected by Gaussian noise. A low-level control layer based on static feedback linearization ensure that any desired velocity can be assigned to the representative point of each UGV, as discussed in Sect. III. Finally, feature extraction is simulated by geometrical computations. We added Gaussian white noise (zero mean, variance 10 pixels) to simulate camera noise, numerical approximation in the feature extraction process, and other unmodeled effects (e.g. image blur or quantization)

Figure 3 presents six snapshots from a typical simulation, fully shown in the video accompanying the paper. Fig. 3a shows the initial arrangement, with the UAV flying above the UGVs. For the first 30 seconds (Fig. 3b) the UGVs perform a preliminary random motion phase, to help the relative localization system retrieve the robot identities and compute reliable estimates of UGV relative poses. After this preliminary phase, the proposed control scheme is activated (Scheme 2). As a result, the desired centroid and variance are quickly achieved on the image; moreover, the UGVs reach the desired formation (Fig. 3c) while moving collectively in a given direction. At some point (Fig. 3d) the UGVs encounter an obstacle. As a consequence, formation is broken as the robots pass the obstacle; the UAV increases its altitude to preserve correct execution of the visual task. Once obstacle avoidance is completed, the UGVs recover their formation (Fig. 3e), and the whole system reaches the steady-state condition shown in Fig. 3f.

We tested the control algorithm in two different scenarios. In the first, the UGV identities are known. In the second, the UGV identities are reconstructed by the localization system

In the first case, which corresponds to have some sort

of tagging on the UGVs, it is not needed to reconstruct the ground-robot identities, hence the relative poses can be reconstructed by directly using camera and sonar measurements. Note that the noise present in the sensors necessarily affects those estimates. Figure 4 shows the plots of the task errors for this simulation. A vertical dashed line separates the first part of the simulation, in which the UGVs move randomly, which is here reported to better compare the two scenarios. The effect of the obstacle avoidance, acting right after 55 seconds, is clearly visible in the graphs.

In the second scenario, we suppose that the robot identities are unknown, thus we rely on the localization system to reconstruct both relative poses and identities. The outcome of the simulation is shown in Fig. 5. The initial random motion of the UGVs allows the localization algorithm to reconstruct the robot identities. Here, we used the same (randomly generated) velocities of the previous case, in order to provide a fair comparison with the previous case.

Compare Fig. 4 and Fig. 5. First, a difference is visible during the phase of random motion for the UGVs: the larger error of the second scenario is due to the initial low accuracy of the relative pose estimates. Moreover, in this initial phase, the control action for the primary tasks is performed only by the UAV, since the lack of knowledge about the ground robot identities does not allow them to cooperate. Second, consider the transient phase after the UGVs pass over the obstacle in the formation error plot reported in Fig. 5. The proximity of the features on the image plane and the coordinates motion of the robots, which results in similar odometry measurements, may trick the best particle selection process which is responsible for the identity assignment, thus the relative localization system introduces some discontinuities [13]. Anyway, the resulting perturbation on the error dynamics is limited, and the system is able to recover convergence to the steady-state conditions.

VIII. CONCLUSIONS

We have presented a cooperative control scheme for a heterogeneous multi-robot system, consisting of an Unmanned Aerial Vehicle (UAV) equipped with a camera and multiple identical Unmanned Ground Vehicles (UGVs). Since the system is highly redundant, the execution of multiple different tasks is possible. The primary visual task is aimed at keeping the UGVs well inside the camera field of view, so as to allow our localization system [13] to reconstruct the identity and relative pose of each UGV with respect to the UAV. Additional tasks include formation control, navigation and obstacle avoidance. We thoroughly discuss the feasibility of each task, proving convergence when possible.

The proposed control framework performs satisfactorily in simulation. We are currently build an experimental setup to further test the robustness of the proposed approach.

REFERENCES

[1] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas, "Cooperative air and ground surveillance," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 16–26, 2006.

[2] F. Morbidi, C. Ray, and G. L. Mariottini, "Cooperative active target tracking for heterogeneous robots with application to gait monitoring," in *2011 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2011, pp. 3608–3613.

[3] S. A. P. Quintero, F. Papi, D. J. Klein, L. Chisci, and J. P. Hespanha, "Optimal uav coordination for target tracking using dynamic programming," in *49th IEEE Conf. on Decision and Control*, 2010, pp. 4541–4546.

[4] B. E. Bishop and D. J. Stilwell, "On the application of redundant manipulator techniques to the control of platoons of autonomous vehicles," in *2001 IEEE Int. Conf. on Control Applications*, 2001, pp. 823–828.

[5] S. Chiaverini, G. Oriolo, and I. Walker, "Chapter 11: Kinematically redundant manipulators," in *Handbook of Robotics*. Springer, 2009, pp. 245–268.

[6] G. Antonelli and S. Chiaverini, "Kinematic Control of Platoons of Autonomous Vehicles," *IEEE Transactions on Robotics*, vol. 22, no. 6, pp. 1285–1292, 2006.

[7] G. Antonelli, F. Arrichiello, and S. Chiaverini, "Experiments of Formation Control With Multirobot Systems Using the Null-Space-Based Behavioral Control," *IEEE Transactions on Control System Technology*, vol. 17, no. 5, pp. 1173–1182, 2009.

[8] G. Chesi, K. Hashimoto, D. Prattichizzo, and A. Vicino, "Keeping Features in the Field of View in Eye-In-Hand Visual Servoing: A Switching Approach," *IEEE Transactions on Robotics*, vol. 20, no. 5, pp. 908–913, 2004.

[9] N. R. Gans, G. Hu, K. Nagarajan, and W. E. Dixon, "Keeping Multiple Moving Targets in the Field of View of a Mobile Camera," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 822–828, 2011.

[10] N. Gans, J. Curtis, P. Barooah, J. Shea, and W. Dixon, "Balancing Mission Requirement for Networked Autonomous Rotorcrafts Performing Video Reconnaissance," in *2009 AIAA Guidance, Navigation, and Control Conference*, 2009, pp. 1–14.

[11] M. Saska, T. Krajnìk, V. Vonàsek, P. Vaněk, and L. Přeučil, "Navigation, Localization and Stabilization of Formations of Unmanned Aerial and Ground Vehicles," in *2013 Int. Conf. on Unmanned Aircraft Systems*, 2013, pp. 813–840.

[12] ——, "Cooperative μUAV-UGV Autonomous Indoor Surveillance," in *9th Int. Multi-Conference on Systems, Signals and Devices*, 2012, pp. 1–6.

[13] P. Stegagno, M. Cognetti, L. Rosa, P. Peliti, and G. Oriolo, "Relative Localization and Identification in a Heterogeneous Multi-Robot System," in *2013 IEEE Int. Conf. on Robotics and Automation*, 2013, pp. 1857–1864.

[14] N. Guenard, T. Hamel, and R. Mahony, "A practical Visual Servo Control for a Unmanned Aerial Vehicle," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 331–340, 2008.

[15] B. Hérissé, T. Hamel, R. Mahony, and F. X. Russotto, "Landing a VTOL Unmanned Aerial Vehicle on a Moving Platform Using Optical Flow," *IEEE Transactions on robotics*, vol. 28, no. 1, pp. 77–89, 2012.

[16] H. K. Khalil, *Nonlinear Systems*. Prentice-Hall, 2002.

[17] Y. Ma, S. Soatto, J. Košecká, and S. S. Sastri, *An invitation to 3-D vision*. Springer, 2004.

[18] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics*. Springer, 2009.