UNIVERSITAT POLITÈCNICA DE CATALUNYA

Doctoral programme:

AUTOMATIC CONTROL, ROBOTICS AND COMPUTER VISION

Ph.D. thesis proposal

VISUAL GUIDANCE OF AUTONOMOUS MICRO AERIAL VEHICLES

Àngel Santamaria-Navarro

Thesis director: Juan Andrade-Cetto

July 16, 2014

Contents

1 Introduction and motivation

Autonomous mobile robots are expected to navigate and move efficiently on different scenarios depending on their assigned tasks. With this purpose, several perception techniques have been developed in recent years. In planar indoor and outdoor domains, 2D range sensing suffices to guarantee adequate guidance, and a large number of deployed systems rely purely on 2D range sensing for safe navigation. In less structured 3D settings however, 2D sensing does not suffice and other sensing alternatives are favored, such as 3D laser scanning [39, 14, 33, 40] or visual perception [25, 8, 9, 44]. Regarding micro aerial vehicles (MAV), where restrictions like payload are challenging, it is interesting the use of onboard and light-weight sensors such as cameras or inertial measurement units (IMU) to drive autonomously the robot. Our research concentrates on MAVs, and in particular, in solving such perception challenges for the accurate localization and guidance of MAVs in GPS-denied environments.

Micro aerial vehicles, and in particular quadrotor systems, have substantially gained popularity in the research community in recent years, motivated by the significant increase in maneuverability, together with a decrease in weight and cost. MAVs are not usually required to interact physically with the environment, but only to perform tracking, surveillance or inspection tasks. Applications however are now appearing for cases in which physical interaction is needed [26, 27, 31, 32, 38].

The development of autonomous MAVs requires various functionalities that exploit environment related information, e.g to achieve mapping tasks, mobile ground target tracking, or robot and target localization. These functionalities in turn, call for the development of new perception techniques to model, identify and recognize the scenario as well as new control techniques specifically designed to consider the vehicle constraints.

Quadrotors are equipped with four aligned coplanar propellers. Due to their symmetric design, motion control is achieved by altering the rotation rate of one or more of these propellers, thereby changing its torque load and thrust lift characteristics. These vehicles use a low-level electronic control system and electronic sensors to stabilize the aircraft during flight (attitude controller). With this actuation technique, a quadrotor becomes an underactuated vehicle with only 4 DOF at a high-level of control (3 linear and 1 angular as shown in Fig. 1). With their small size and agile maneuverability, quadcopters can be flown indoors as well as outdoors. For indoor quadrotor systems, accurate fast-rate localization is usually obtained from infrared multicamera systems such as the Vicon [2] or the Optitrack [1]. However, for outdoor tasks, such infrared-based localization devices do not work and to accurately localize the platform in real time, other means should be used, such as the integration of odometry estimates, using onobard sensors or visual servoing to control the motion of the robot using computer vision techniques.

The ability for MAVs to manipulate or carry objects could greatly expand the types of missions achievable by such unmanned systems. High perfor-

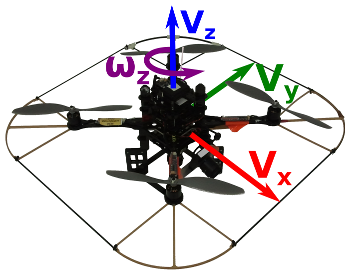

Figure 1: Quadrotor DOF.

mance arms with end-effectors typically weigh more than 10 kg, which cannot be supported by most commercially available small-sized MAVs. Recent developments however suggest a trend change with MAV payload capabilities increasing and arm weights getting smaller [20, 32]. In this thesis we will explore the perception and control issues that arise when a MAV is equipped with a low weight arm.

2 Objectives

As stated before, the main motivation of this Ph.D. thesis will be the study and development of new perception techniques to model, identify and recognize the scenario and their use in the guidance of MAVs. Specifically some of these techniques will be tested in assembly operations by means of flying robots. These assembly operations will consist on creating a structure autonomously formed by several bars and joints that initially should be picked up, transported to the assembly area and assembled into the current structure. Our work concentrates on all perception issues pertaining this task, but does not include the issues related to the actual grasping and manipulation. These items are also part of the EU project ARCAS to our work belongs, but fall out of the scope of the thesis.

The tasks that we will solve include the self-positioning of the MAV in front of a target. By means of this, the robot should be able to detect the object and precisely hover in order to grasp or release it. We will contribute with new approaches to drive the flying robot towards a target using visual servoing techniques. Thus, we expect not only to contribute with a new robust technique for uncalibrated cameras but with new control law proposals using that visual information.

An important aspect to be considered while guiding autonomously a vehicle is the robot odometry estimation. During the task, the robot should be able to estimate its velocity and position state using onboard sensors. Considering the MAV characteristics. This thesis aims to provide new sensor fusion techniques to efficiently estimate the vehicle odometry by filtering the data obtained from the cameras together with other onboard sensors such as an IMU.

In addition, flying with a suspended load is a challenging task because the load significantly changes the flight characteristics of the aerial vehicle, and the stability of the vehicle-load system must be preserved. Therefore, it is essential that the flying robot has the ability to minimize the effects of the arm on the flying system during the assigned maneuvers. Specifically our objective is to research and develop new techniques to minimize those effects while a visual guidance is performed in order to improve flight behaviours.

Considering a task oriented autonomous guidance, the robot must be localized in the scenario. To this end, two different solutions will be proposed. Firstly, a coarse localization and mapping technique based on range readings provided by radio beacons. Secondly, a fine localization technique using all available oboard sensors toghether with visual mark detections.

Driving autonomously a free-flying aerial manipulator entails a lot of challenging perception aspects. The objectives of this work is to provide working solutions to some of them, namely

- MAV odometry estimation using onboard sensors.
- New visual servoing techniques to drive the flying robot towards a target.
- Robot localization and mapping methods in order to guide the robot in the scenario.
- Control law proposals specifically designed for kinematically augmented MAVs.

3 Expected contributions

This Ph.D. work seeks to contribute to the visual guidance of autonomous micro aerial vehicles attached with moving parts, such as a robotic arm.

To this end, we requiere robust visual servoing methods that drive the robot towards the target specially considering the change in the robot arquitecture as well as the task specifications. To date, we have already contributed in this regard with a new uncalibrated image-based visual servoing approach with mild assumptions about principal point and skew values of the camera intrinsic parameters, without requiring a priori knowledge of camera focal length. The technique will be used to stabilize an MAV during hovering in front of the target or the landing approach maneuver. In order to expand the visual servoing approach, the simple linear controller developed so far will be extended with a new control law techniques using non-linear model predictive control (NMPC).

In the present thesis project, we also expect to propose an incremental approach of the above mentioned visual servoing technique by considering the aerial manipulator specific arquitecture and taking advantage of the redundacy in DOFs given by the attached manipulator. This DOF redundancy will be exploited not only to achieve a desired visual servo task, but to do so whilst attaining secondary tasks during the mission, such as to reduce the dynamical effects of the suspended load or to reach arm poses with high manipulability ratios.

Current research on MAV vehicles relies on indoor testbeds where an external positioning system can be used, usually these systems are characterized by a high frame rate and accuracy and use infrared cameras to track the objects in the testbed. These external localization systems provide very fine pose estimates almost to be considered as a ground truth in most of the literature. However, these external localization systems can only be used in indoor laboratories. In this thesis we will develop a sensor fusion approach using an optical flow sensor ([16]) together with an IMU and a 3D compass to obtain an accurate odometry of the vehicle. Those sensors can work at a very high frame rate (up to 200 Hz in outdoor scenarios).

Moreover, we will collaborate with researchers from the University of Seville to develop a new coarse localization and mapping technique using radio beacons. Then, a new fine localization approach that uses such low quality localization estimates will be presented in order to autonomously servo the robot to a desired target.

During this thesis we also planned the attendance to the EuRathlon/ARCAS Workshop and International Summer School on Field Robotics 2014 and envisaged a four months stage with an international research group.

We finally note that contributions are not restricted to the scope of the EU project ARCAS for building structures. Aerial manipulators are expected to end up working in many other tasks where a high precision in terms of visual guidance and control is required, together with physical interaction with the environment.

4 State of the art

A review of the state of the art on the tackled problems is provided. This section is divided in two parts: visual servoing and task priority control. The state of the third objective of this thesis on accurate robot localization in gps-denied environments is still work in progress.

4.1 Visual servoing

Vision-based robot control systems are usually classified in three groups: position-based visual servo, image-based visual servo, and hybrid control systems [8, 9]. In position-based visual servo, the geometric model of the target is used in conjunction with visual features extracted from the image to estimate the pose of the target with respect to the camera frame. The control law is designed to reduce such pose error. For this reason, the approach is also referred as 3D visual servoing [18]. Minimizing error in pose has the disadvantage that features could easily be lost from the image during the servo loop. In image-based visual servoing on the other hand, the control law is defined directly in the image plane, minimizing the error between observed and desired image feature coordinates [45, 12]. There exist however stability and convergence problems that may occur, either because the image Jacobian becomes singular during the servoing task, or because the controller falls in a local minima at points with unrealizable image motion [7].

This situation can be palliated to some extent with the use of hybrid approaches, which entail some combination of both groups of algorithms. The 2-1/2-D hybrid visual servoing scheme [25] estimates partial camera displacement at each iteration of the control law and minimizes a functional of both, the error measures in image space typical from image-based servo and a log depth ratio accounting for the rate at which the camera moves to the target. Another hybrid approach is the partitioned visual servo scheme [11], which is based on adding to the traditional image-based error function a term decoupling the motion and rotation along the z axis. To this end two new image features are introduced, one of them related to the area of the polygon being tracked.

In all image-based and hybrid approaches however, the resulting image Jacobian or interaction matrix, which relates the camera velocity with the image feature velocities, depends on a priori knowledge of intrinsic camera parameters. To do away with this dependence, one could optimize for the parameters in the image Jacobian while error in the image plane is being minimized. This is done for instance, using Gauss-Newton to minimize squared image error and non-linear least squares optimization for the image Jacobian [36, 37]; using weighted recursive least squares (RLS), not to obtain the true parameters, but instead an approximation that still guarantees asymptotic stability of the control law in the sense of Lyaponov [17]; or using k-nearest neighbor regression to store previously estimated local models or previous movements, and estimating the Jacobian using local least squares (LLS) [13]. To provide robustness to outliers in the computation of the Jacobian, [43] proposes the use of an M-estimator.

4.2 Task priority control

Recently, MAVs have been proposed for grasping and manipulation tasks. This is a challenging issue since the vehicle is characterized by an unstable dynamics [38, 27] when interaction with the environment is needed.

Flying a quadrotor with a serial robotic arm attached is a complex task because the load distribution significantly changes the flight characteristics for different arm configurations or when the arm is grasping an object. In those situations the stability of the vehicle-load system must be preserved. Therefore, it is essential that the flying robot has the ability to minimize the effects of the arm on the flying system during the assigned maneuvers [19, 34].

To avoid this undesired behavior, the redundancy of the system obtained with the arm's extra degrees of freedom could be exploited to achieve additional tasks acting on the null space of the quadrotor-arm Jacobian [30]. More specifically it can be used to develop a secondary stabilizing tasks after the primary servoing task.

The attached arm produces undesired dynamic effects to the quadrotor, such as the change of the center of mass during flight, that can be solved designing a low-level attitude controller such as a Cartesian impedance controller [22, 23], or an adaptive controller [4]. However, in [23] the redundancy is not exploited to modulate the mobility request for the flying platform and the robotic arm. In that sense, the secondary tasks, that can also be performed within a hierarchical framework, could be designed such as to optimize some given quality indices, e.g. manipulability, joint limits, etc., [5, 10].

5 Preliminary work

We now report the current state of our research in two of the three topics addressed in this thesis: visual servoing and secondary task priority control. The third item, global localization is work in progress.

5.1 Visual Servoing

We proposed in ICRA13' a new method for uncalibrated image-based visual servoing [41]. In contrast to traditional image-based visual servo, the proposed solution does not require a known value of camera focal length for the computation of the image Jacobian. Instead, it is estimated at run time from the observation of the tracked target. The technique is shown to outperform classical visual servoing schemes in situations with noisy calibration parameters and for unexpected changes in the camera zoom.

Background

Drawing inspiration on the EPnP [28] and UPnP [35] algorithms, we can solve for the camera pose and focal length using a reference system attached to the target object, and the known distance constraints between them as follows. We define a set of four control points as a basis for this reference system. Then, one can express the 3D coordinates of each target feature as a weighted sum of the elements of this basis. Computing the pose of the object with respect to the camera resorts to computing the location of these control points with respect to the camera frame. A least squares solution for the control point coordinates albeit scale, is given by the null eigenvector of a linear system made up of all 2D to 3D perspective projection relations between the target points. Given the fact that distances between control points must be preserved, these distance constraints can be used in a second least squares computation to solve for scale and focal length.

More explicitly, the perspective projection equations for each target feature become

$$\sum_{j=1}^{4} \left(a_{ij} x_j + a_{ij}(u_0 - u_i)\frac{z_j}{\alpha} \right) = 0, \tag{1}$$

$$\sum_{j=1}^{4} \left(a_{ij} y_j + a_{ij}(v_0 - v_i)\frac{z_j}{\alpha} \right) = 0, \tag{2}$$

where $\mathbf{s}_i = [u_i, v_i]^T$ are the image coordinates of the target feature i, and $\mathbf{c}_j = [x_j, y_j, z_j]^T$ are the 3D coordinates of the j-th control point in the camera frame. The terms a_{ij} are the barycentric coordinates of the i-th target feature which are constant regardless of the location of the camera reference frame, and α is our unknown focal length.

These equations can be jointly expressed for all 2D-3D correspondences as a linear system

$$\mathbf{Mx} = \mathbf{0}\,, \tag{3}$$

where M is a $2n \times 12$ matrix made of the coefficients a_{ij}, the 2D points $\mathbf{s}_i$ and the principal point; and $\mathbf{x}$ is our vector of 12 unknowns containing both the 3D coordinates of the control points in the camera reference frame and the camera focal length, dividing the z terms:

$$\mathbf{x} = [x_1, y_1, z_1/\alpha, ..., x_4, y_4, z_4/\alpha]^T. \tag{4}$$

Its solution lies in the null space of $\mathbf{M}$, and can be computed as a scaled product of the null eigenvector of $\mathbf{M}^T\mathbf{M}$ via Singular Value Decomposition

$$\mathbf{x} = \beta\mathbf{v}\,, \tag{5}$$

the scale β becoming a new unknown. In the noise-free case, $\mathbf{M}^T\mathbf{M}$ is only rank deficient by one, but when image noise is severe $\mathbf{M}^T\mathbf{M}$ might loose rank, and a more accurate solution can be found as a linear combination of the basis of its null space. In this work we are not interested on recovering accurate camera pose, but on minimizing the projection error within a servo task. It is sufficient for our purposes to consider only the least squares approximation. That is, to compute the solution only using the eigenvector associated to the smallest eigenvalue.

To solve for β we add constraints that preserve the distance between control points of the form

$$||c_j - c_{j'}||^2 = d_{jj'}^2\,, \tag{6}$$

where $d_{jj'}$ is the known distance between control points c_j and $c_{j'}$ in the world coordinate system. Substituting $\mathbf{x}$ in the six distance constraints of Eq. 6, we obtain a system of the form

$$\mathbf{Lb} = \mathbf{d}, \tag{7}$$

where $\mathbf{b} = [\beta^2, \alpha^2\beta^2]^T$, $\mathbf{L}$ is a 6×2 matrix built from the known elements of $\mathbf{v}$, and $\mathbf{d}$ is the 6-vector of squared distances between the control points. We solve this overdetermined linearized system using least squares and estimate the magnitudes of α and β by back substitution

$$\alpha = \sqrt{\frac{|b_2|}{|b_1|}}\,, \tag{8}$$

$$\beta = \sqrt{b_1}\,. \tag{9}$$

Image Jacobian

As the camera moves, the velocity of each target control point $\mathbf{c}_j$ in camera coordinates can be related to the camera spatial velocity $(\mathbf{t}, \mathbf{\Omega})$ with

$$\dot{\mathbf{c}}_j = -\mathbf{t} - \mathbf{\Omega} \times \mathbf{c}_j \tag{10}$$

which corresponds to

$$\begin{bmatrix} \dot{x}_j \\ \dot{y}_j \\ \dot{z}_j \end{bmatrix} = \begin{bmatrix} -t_x - \omega_y \, z_j + \omega_z \, y_j \\ -t_y - \omega_z \, x_j + \omega_x \, z_j \\ -t_z - \omega_x \, y_j + \omega_y \, x_j \end{bmatrix} . \tag{11}$$

Injecting Eq. 5 in Eq. 11, we obtain

$$\begin{bmatrix} \dot{x}_j \\ \dot{y}_j \\ \dot{z}_j \end{bmatrix} = \begin{bmatrix} -t_x - \omega_y \, \alpha \, \beta v_z + \omega_z \, \beta v_y \\ -t_y - \omega_z \, \beta v_x + \omega_x \, \alpha \, \beta v_z \\ -t_z - \omega_x \, \beta v_y + \omega_y \, \beta v_x \end{bmatrix} , \tag{12}$$

where v_x, v_y, and v_z are the x, y, and z components of eigenvector $\mathbf{v}$ related to the control point $\mathbf{c}_j$, and whose image projection is given by

$$\begin{bmatrix} u_j \\ v_j \end{bmatrix} = \begin{bmatrix} \alpha \frac{x_j}{z_j} + u_0 \\ \alpha \frac{y_j}{z_j} + v_0 \end{bmatrix} , \tag{13}$$

and its time derivative by

$$\begin{bmatrix} \dot{u}_j \\ \dot{v}_j \end{bmatrix} = \alpha \begin{bmatrix} \frac{\dot{x}_j}{z_j} - \frac{x_j \dot{z}_j}{z_j^2} \\ \frac{\dot{y}_j}{z_j} - \frac{y_j \dot{z}_j}{z_j^2} \end{bmatrix} . \tag{14}$$

Substituting Eqs. 5 and 12 in Eq. 14 we obtain

$$\dot{u}_j = \frac{-t_x - \alpha\beta v_z \omega_y + \beta v_y \omega_z}{\beta v_z} - \frac{v_x(-t_z - \beta v_y \omega_x + \beta v_x \omega_y)}{\alpha \beta v_z^2} \tag{15}$$

$$\dot{v}_j = \frac{-t_y - \alpha\beta v_z \omega_x + \beta v_x \omega_z}{\beta v_z} - \frac{v_y(-t_z - \beta v_y \omega_x + \beta v_x \omega_y)}{\alpha \beta v_z^2} , \tag{16}$$

which can be rewritten as

$$\dot{\mathbf{s}}_j = \mathbf{J}_j \, \mathbf{v}_c , \tag{17}$$

with $\dot{\mathbf{s}}_j = [\dot{u}_j, \dot{v}_j]^T$, the image velocities of control point j, and $\mathbf{v}_c = [\mathbf{t}^T, \mathbf{\Omega}^T]^T$. $\mathbf{J}_j$ is our seeked calibration-free image Jacobian for the j-th control point, and takes the form

$$\mathbf{J}_j = \begin{bmatrix} \frac{-1}{\beta v_z} & 0 & \frac{v_x}{\alpha \beta v_z^2} & \frac{v_x v_y}{\alpha v_z^2} & \frac{-v_x^2 - \alpha^2 v_z^2}{\alpha v_z^2} & \frac{v_y}{v_z} \\ 0 & \frac{-1}{\beta v_z} & \frac{v_y}{\alpha \beta v_z^2} & \frac{v_y^2 + \alpha^2 v_z^2}{\alpha v_z^2} & \frac{-v_x v_y}{\alpha v_z^2} & \frac{-v_x}{v_z} \end{bmatrix} . \tag{18}$$

Stacking these together, we get the image Jacobian for all control points

$$\mathbf{J}_{vs} = \begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_4 \end{bmatrix} . \tag{19}$$

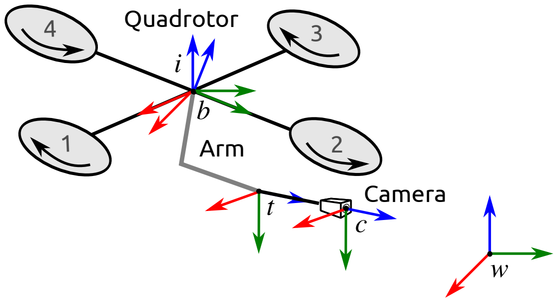

Figure 2: Distribution of coordinate frames within the robot-arm system.

Control Law

The aim of our image-based control scheme is to minimize the error

$$\mathbf{e}(t) = \mathbf{s}(t) - \mathbf{s}^* , \tag{20}$$

where $\mathbf{s}(t)$ are the current image coordinates of the set of target features, and $\mathbf{s}^*$ are their final desired positions in the image plane, computed with our initial value for α. If we select $\mathbf{s}$ to be the projection of the control points $\mathbf{c}$, and disregarding the time variation of α, and consequently of $\mathbf{s}^*$, the derivative of Eq. 20 becomes

$$\dot{\mathbf{e}} = \dot{\mathbf{s}} = \mathbf{J}_{vs}\, \mathbf{v}_c \, . \tag{21}$$

The desired camera velocities $\mathbf{v}_c$ that would drive the robot with an exponential decoupled decrease of the error, i.e., $\dot{\mathbf{e}} = -\lambda \mathbf{e}$, become

$$\mathbf{v}_c = -\lambda\, \mathbf{J}_{vs}^{+}\, \mathbf{e}, \tag{22}$$

where $\mathbf{J}_{vs}^{+}$ is the left Moore-Penrose pseudoinverse of $\mathbf{J}_{vs}$, that is $\mathbf{J}_{vs}^{+} = (\mathbf{J}_{vs}^{T}\, \mathbf{J}_{vs})^{-1}\, \mathbf{J}_{vs}^{T}$.

For a more exhaustive explanation of this method we refer the reader to [41].

5.2 Task priority control

The above control law was designed to drive the robot so as to minimize the error in image coordinates, i.e. to servo the robot to a desired location with respect to a target. We have been working on an extension to this technique that exploits the extra degrees of freedom provided by an arm attached to the base of the robot, to achieve secondary tasks that enhance overall platform performance [42].

Coordinate Frames

Consider the quadrotor-arm system equipped with a camera mounted on the arm's end-effector's as shown in Fig. 2. The goal is to servo the camera to a desired target, say for instance, a fiducial mark on an object to be manipulated. We assume the visual servo approach from section 5.1 to provide camera velocities to reach this task.

Without loss of generality, we consider the world frame w to be located at the target. With this, the position of the target with respect to the camera in

c can be computed integrating the camera velocities obtained from the visual servo, and expressed as a homogeneous transform $\mathbf{T}_c^w$.

The quadrotor high-level controller commands velocities in the so-called inertial frame i, as shown in Fig. 2. This frame indicates the location of the vehicle w.r.t. w but rotated about the yaw axis. Both frames i and w have their x and y axes in parallel planes. The quadrotor is an underactuated vehicle [15] with only 4 DOF, namely the linear velocities plus the yaw angular velocity $(v_{qx}, v_{qy}, v_{qz}, \omega_{qz})$ acting on this inertial frame. The low-level attitude controller moves the quadrotor body frame b to reach the desired velocities in i. Both frames i and b have the respective origins in the same point but a rotation about the roll and pitch angles exists between them.

Let $\mathbf{q}_a = \left[q_{a1}, \ldots, q_{an}\right]^T$ be the angles of the n joints of the robotic arm attached to the vehicle. With the arm base frame coincident with the quadrotor body frame b, the relation between the quadrotor inertial frame and the camera frame, $\mathbf{T}_c^i$, is given by the concatenation of the fixed quadrotor inertial-body and tool-camera transforms with that of the arm kinematics $\mathbf{T}_t^b(\mathbf{q}_a)$

$$\mathbf{T}_c^i = \mathbf{T}_b^i \, \mathbf{T}_t^b \, \mathbf{T}_c^t \, . \tag{23}$$

Hence, the pose of the quadrotor $(x, y, z, \phi, \theta, \psi)$ with respect to the target is determined by the transform

$$\mathbf{T}_w^i = \mathbf{T}_c^i \, (\mathbf{T}_c^w)^{-1} \, . \tag{24}$$

Robot Kinematics

We are in the position now to define a joint quadrotor-arm Jacobian that relates the local translational and angular velocities of the platform acting on the inertial frame and those of the n arm joints, $\mathbf{v}_{qa} = (v_{qx}, v_{qy}, v_{qz}, \omega_{qx}, \omega_{qy}, \omega_{qz}, \dot{q}_{a1}, \ldots, \dot{q}_{an})^T$, to the desired camera velocities as computed from the visual servo

$$\mathbf{v}_c = \mathbf{J}_{qa} \, \mathbf{v}_{qa} \, , \tag{25}$$

with $\mathbf{J}_{qa}$ the Jacobian matrix of the whole robot.

This velocity vector in the camera frame, can be expressed as a sum of the velocities added by the quadrotor movement and the arm kinematics (superscripts indicate the reference frame to make it clear to the reader)

$$\mathbf{v}_c^c = \mathbf{v}_q^c + \mathbf{v}_a^c \, , \tag{26}$$

where $\mathbf{v}_a^c$ is obtained with the arm Jacobian $\mathbf{J}_a$.

$$\mathbf{v}_a^c = \begin{bmatrix} \mathbf{R}_b^c & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_b^c \end{bmatrix} \mathbf{J}_a \, \dot{\mathbf{q}}_a = \overline{\mathbf{R}}_b^c \, \mathbf{J}_a \, \dot{\mathbf{q}}_a \, , \tag{27}$$

and where $\mathbf{R}_b^c$ indicates the rotation of b with respect to c, and $\mathbf{v}_q^c$ corresponds to the velocity of the quadrotor expressed in the c frame,

$$\mathbf{v}_q^c = \overline{\mathbf{R}}_b^c \begin{bmatrix} \mathbf{v}_q^b + \boldsymbol{\omega}_q^b \times \mathbf{r}_c^b \\ \boldsymbol{\omega}_q^b \end{bmatrix} = \begin{bmatrix} \mathbf{R}_b^c & \mathbf{R}_b^c \left[\mathbf{r}_c^b\right]_\times^T \\ \mathbf{0} & \mathbf{R}_b^c \end{bmatrix} \mathbf{v}_q^b \, . \tag{28}$$

The term $\mathbf{r}_c^b(\mathbf{q}_a)$ indicates the vector between the b and c frames, i.e. the direct arm kinematics.

Finally, the velocity vector of the quadrotor in the body frame, $\mathbf{v}_q^b$, can be obtained using the quadrotor Jacobian $\mathbf{J}_q$ formed by the rotation $\mathbf{R}(\phi, \theta)$ and the transfer matrix $\mathbf{T}(\phi, \theta)$ between the quadrotor inertial and body frames

$$\mathbf{v}_q^b = \mathbf{J}_q \, \mathbf{v}_q^i = \begin{bmatrix} \mathbf{R} & 0_{3\times3} \\ 0_{3\times3} & \mathbf{T} \end{bmatrix} \mathbf{v}_q^i \,, \tag{29}$$

where

$$\begin{aligned} \mathbf{R}(\phi, \theta) &= \begin{bmatrix} c_\theta & s_\theta \, s_\phi & s_\theta \, c_\phi \\ 0 & c_\phi & -s_\phi \\ -s_\theta & c_\theta \, s_\phi & c_\theta c_\phi \end{bmatrix} \,, \\ \mathbf{T}(\phi, \theta) &= \begin{bmatrix} 1 & s_\phi \, t_\theta & c_\phi \, t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi / c_\theta & c_\phi / c_\theta \end{bmatrix} \,, \end{aligned} \tag{30}$$

and the notation $s_x = \sin(x)$, $c_x = \cos(x)$, $t_x = \tan(x)$.

The camera velocities are obtained using the above mentioned visual servo front end [41]. Combining Eqs. 22 and 25, we get

$$\mathbf{J}_{qa} \mathbf{v}_{qa} = -\lambda \mathbf{J}_{vs}^{+} \mathbf{e} \,. \tag{31}$$

Unfortunately, the quadrotor is an underactuated vehicle [15] with only 4 DOF. Its pitch and roll are internally controlled by the attitude subsystem and we cannot directly actuate them. So, to remove these variables from the control command, their contribution to the visual servo error can be isolated from that of the other control variables by extracting the columns of $\mathbf{J}_{qa}$ and the rows of $\mathbf{v}_{qa}$ corresponding to ω_{qx} and ω_{qy}, reading out these values from the platform gyroscopes, and subtracting them from the camera velocity [24].

Rearranging terms

$$\mathbf{J}_{qa1} \dot{\mathbf{q}} = \underbrace{-\lambda \mathbf{J}_{vs}^{+} \mathbf{e} - \mathbf{J}_{qa2} \begin{bmatrix} \omega_{qx} \\ \omega_{qy} \end{bmatrix}}_{\dot{\mathbf{q}}_1} \,, \tag{32}$$

where $\mathbf{J}_{qa2}$ is the Jacobian formed by the columns of $\mathbf{J}_{qa}$ corresponding to ω_{qx} and ω_{qy}, and $\mathbf{J}_{qa1}$ is the Jacobian formed by all other columns of $\mathbf{J}_{qa}$, corresponding to the actuated variables $\dot{\mathbf{q}} = [v_{qx}, v_{qy}, v_{qz}, v_{qz}, \dot{q}_{a1}, \ldots, \dot{q}_{an}]^T$.

With this, $\dot{\mathbf{q}}_1$ becomes our primary task velocity corresponding to the visual servo.

$$\dot{\mathbf{q}} = \mathbf{J}_{qa1}^{\#} \, \dot{\mathbf{q}}_1 \,, \tag{33}$$

where $\mathbf{J}_{qa1}^{\#} = \mathbf{W}^{-1} \, \mathbf{J}_{qa1}^T \, (\mathbf{J}_{qa1} \, \mathbf{W}^{-1} \, \mathbf{J}_{qa1}^T)^{-1}$ is the weighted generalized inverse of the matrix $\mathbf{J}_{qa1}$ where the weight matrix $\mathbf{W}$ affects the motion distribution over the controlled variables considering the different moving capabilities of the robotic arm and the quadrotor. More specifically, large movements of the flying platform should be achieved by the quadrotor leaving more precise motion to the robotic arm due to its dexterity, i.e. regulating their actuation as a function of the distance d of the platform to the target.

To achieve this behavior, we define a time-varying diagonal weight-matrix as proposed in [21]

$$\mathbf{W}(d) = \mathrm{diag}((1-\alpha)\,\mathbf{I}_4, \alpha\,\mathbf{I}_n), \tag{34}$$

with n the arm's DOF and

$$\alpha(d) = \frac{1+\underline{\alpha}}{2} + \frac{1-\underline{\alpha}}{2}\tanh\left(2\,\pi\,\frac{d-\delta_W}{\Delta_W-\delta_W} - \pi\right), \tag{35}$$

where $\alpha \in [\underline{\alpha}, 1]$, and δ_W and Δ_W ($\Delta_W > \delta_W$) are the distance thresholds corresponding to $\alpha \cong 1$ and $\alpha \cong \underline{\alpha}$, respectively. The blocks of $\mathbf{W}$ weight differently the velocity components of the arm and the quadrotor by increasing the velocity of the quadrotor when the distance to the target $d > \Delta_W$, while for distances $d < \delta_W$ the quadrotor is slowed down and the arm is commanded to accommodate the precise arm movements.

Note that since the quadrotor is underactuated, we have to choose $\underline{\alpha}$ in such a way that the arm can still compensate the roll and pitch movements produced by the quadrotor during flight not to loose the target from the image plane.

Task Priority Control

The redundancy obtained with the arm's extra degrees of freedom can be exploited to achieve additional tasks acting on the null space of the quadrotor-arm Jacobian [30], while preserving the primary task in Eq. 33:

$$\dot{\mathbf{q}} = \mathbf{J}_{qa1}^{\#}\,\dot{\mathbf{q}}_1 + \mathbf{N}_{qa1}\,\dot{\mathbf{q}}_0\,, \tag{36}$$

where $\mathbf{N}_{qa1} = (\mathbf{I} - \mathbf{J}_{qa1}^{+}\,\mathbf{J}_{qa1})$ is the null space projector for the main task. With this, the secondary task velocity $\dot{\mathbf{q}}_0$ will be used to reconfigure the robot structure without changing both the position and orientation of the end-effector (usually referred to as *internal motion*).

One possible way to specify the secondary task is to choose the velocity vector $\dot{\mathbf{q}}_0$ as the gradient of a scalar objective function to achieve some kind of optimization [10, 29]. With a more general approach, let $\boldsymbol{\sigma} = \mathbf{f}(\mathbf{q}) \in \mathbb{R}^m$ be the variables of a secondary task to be controlled, the following differential relationship holds:

$$\dot{\boldsymbol{\sigma}} = \frac{\partial \mathbf{f}(\mathbf{q})}{\partial \mathbf{q}}\dot{\mathbf{q}} = \mathbf{J}_{\sigma}(\mathbf{q})\dot{\mathbf{q}}\,, \tag{37}$$

where $\mathbf{J}_{\sigma}(\mathbf{q}) \in \mathbb{R}^{m\times(4+n)}$ is the configuration-dependent task Jacobian. Hence, by inverting Eq. 37 and by considering a regulation problem of $\boldsymbol{\sigma}$ to the desired value $\boldsymbol{\sigma}^*$, the following general solution can be employed

$$\dot{\mathbf{q}} = \mathbf{J}_{qa1}^{\#}\,\dot{\mathbf{q}}_1 + \mathbf{N}_{qa1}\,\mathbf{J}_{\sigma}^{+}\boldsymbol{\Lambda}_{\sigma}\tilde{\boldsymbol{\sigma}}\,, \tag{38}$$

where $\boldsymbol{\Lambda}_{\sigma} \in \mathbb{R}^{m\times m}$ is a positive-definite matrix of gains, and $\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}^* - \boldsymbol{\sigma}$ is the task error.

Considering the high redundancy of the quadrotor-arm system, multiple secondary tasks can be arranged in hierarchy. As proposed in [21], the secondary objective function can be defined as a weighted sum of different objective sub-functions, with the advantage that the weights can be modeled time-varying, i.e. the effect of the secondary task can be changed depending on the phase of the flight. However, the use of some of the sub-functions at the same time can produce undesired behaviours on the arm due to opposite effects of the sub-tasks. To deal with that and to avoid conservative stability conditions [3], the *augmented inverse-based projections method* is here considered [5]. In detail, the generic task is not projected onto the null space of the high hierarchy task, but onto the null space of the task achieved by considering the augmented Jacobian of all the higher hierarchy tasks.

In this work we consider two sub-tasks: 1) center of gravity control, 2) joint-limits avoidance control. By denoting with $\mathbf{J}_G$ and $\mathbf{J}_L$ the Jacobian matrices for the center of gravity and for the joint-limits avoidance control, respectively, where the priority of the task follows the previous enumerating order, the desired system velocity can be rewritten as follows,

$$\dot{\mathbf{q}} = \mathbf{J}_{qa1}^{\#}\,\dot{\mathbf{q}}_1 + \mathbf{N}_{qa1}\,\mathbf{J}_G^{+}\,\tilde{\sigma}_G + \mathbf{N}_{qa1|G}\,\mathbf{J}_L^{+}\,\tilde{\sigma}_L\,, \tag{39}$$

with $\mathbf{N}_{qa1|G}$, the joint projector of the primary task and of the center of gravity secondary task, which is defined as

$$\mathbf{N}_{qa1|G} = (\mathbf{I} - \mathbf{J}_{qa1|G}^{+}\,\mathbf{J}_{qa1|G})\,, \tag{40}$$

and $\mathbf{J}_{qa1|G}$ represents the augmented Jacobian

$$\mathbf{J}_{qa1|G} = \begin{bmatrix}\mathbf{J}_{qa1}\\ \mathbf{J}_G\end{bmatrix}\,. \tag{41}$$

We now define the scalar objective functions for each of the secondary tasks in the hierarchy.

Center of Gravity

If the arm and quadrotor center of gravity (CoG) are not vertically aligned, the motion of the arm produces an undesired torque on the quadrotor base, that perturbs the system attitude and position. This effect can be mitigated by minimizing the distance between the arm CoG and the vertical line of the quadrotor gravity vector.

The task function we introduce is the square distance of the arm CoG with respect to the z axis of the i frame, which can be written as

$$\sigma_G = \lambda_G\,(\mathbf{p}^i_{Gxy})^T\,\mathbf{p}^i_{Gxy}, \tag{42}$$

where λ_G is a suitable positive gain and with

$$\mathbf{p}^i_{Gxy} = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\mathbf{R}^i_b\,\mathbf{p}^b_G, \tag{43}$$

where $\mathbf{R}_b^i$ is the rotation matrix of the body frame b with respect to the inertial frame i, and the desired task variable is $\sigma_G^* = 0$, i.e. $\tilde{\sigma}_G = -\sigma_G$. The position of the arm CoG $\mathbf{p}_G^b$ is a function of the arm joint configuration and is defined by

$$\mathbf{p}_G^b = \frac{\sum_{i=1}^n m_i \, \mathbf{p}_{Gi}^b}{\sum_{i=1}^n}, \tag{44}$$

where m_i and $\mathbf{p}_{Gi}^b$ represent the i-th link mass and the position of its CoG.

As proposed in [6], we can define the CoG of a partial chain of links, with respect to the body frame, from the link j to the end-effector as

$$\mathbf{p}_{Gj}^{*b} = \mathbf{R}_j^b \frac{\sum_{i=j}^n m_i \, \mathbf{p}_{Gi}^b}{\sum_{i=j}^n}, \tag{45}$$

where $\mathbf{R}_j^b$ is the existing rotation between the link j and the quadrotor body frame. Notice that all these quantities are functions of the current joint configuration $\mathbf{q}_a$.

The differential relationship between the CoG, $\mathbf{p}_G$, and the arm joint values is

$$\dot{\mathbf{p}}_G^b = \mathbf{J}_G^b \, \dot{\mathbf{q}}_a, \tag{46}$$

where $\mathbf{J}_G^b \in \mathbb{R}^{3\times n}$ is the CoG Jacobian, expressed in the quadrotor body frame, defined as follows

$$\mathbf{J}_G^b = \frac{\partial \mathbf{p}_G^b}{\partial \mathbf{q}_a} = \left(\mathbf{J}_{G1}^b ... \mathbf{J}_{Gn}^b\right), \tag{47}$$

with $\mathbf{J}_{Gi}^b$ the individual joint i Jacobian formulated from the partial CoG

$$\mathbf{J}_{Gj}^b = \frac{\sum_{i=j}^n m_i}{\sum_{i=0}^n} \left(\mathbf{z}_j \times \mathbf{p}_{Gj}^{*b}\right). \tag{48}$$

Notice how the resultant linear velocity is scaled by the mass of the partial CoG in Eq. 48 because the CoG is the average of the multi-mass system and high velocities on smaller masses play a lesser role on the total velocity of the CoG.

Finally, the corresponding task Jacobian from the derivative of Eq. 42 becomes

$$\mathbf{J}_G = \begin{bmatrix} \mathbf{0}_{1\times 4} & 2\,\lambda_G \,(\mathbf{p}_{Gxy}^i)^T \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \mathbf{R}_b^i \, \mathbf{J}_G^b \end{bmatrix}. \tag{49}$$

With this choice, the CoG of the arm is controlled to be aligned with the CoG of the vehicle along the direction of the gravitational force.

Joint Limits

To avoid arm joint limits we can drive the arm joints toward a desired value $\mathbf{q}_a^*$ that can be chosen far from an undeliverable configuration and/or close one characterized by a high manipulability index or suitable with respect to the assigned task. Hence our cost function for this task is a weighted squared sum of the joint angle differences from the desired values over the joint limit ranges

$$\sigma_L = \sum_{i=1}^{n} \lambda_L \left(\frac{q_{ai} - q_{ai}^*}{\overline{q}_{ai} - \underline{q}_{ai}} \right)^2 , \qquad (50)$$

where $\overline{q}_{ai}$ and $\underline{q}_{ai}$ are the high and low joint limit values, respectively, for the i-th link . Rearranging Eq. 50, the proposed task function becomes

$$\sigma_L = (\mathbf{q}_a - \mathbf{q}_a^*)^T \, \mathbf{\Lambda}_L \, (\mathbf{q}_a - \mathbf{q}_a^*), \qquad (51)$$

where $\mathbf{\Lambda}_L$ is a diagonal matrix whose diagonal elements are equal to the inverse of the squared joint limit ranges

$$\mathbf{\Lambda}_L = \lambda_L \, \mathrm{diag}\{(\overline{q}_{a1} - \underline{q}_{a1})^{-2}, \ldots, (\overline{q}_{an} - \underline{q}_{an})^{-2}\}, \qquad (52)$$

and λ_L is a suitable positive gain. The desired task variable is $\sigma_L^* = 0$ (i.e. $\tilde{\sigma}_L = -\sigma_L$), and the corresponding task Jacobian is

$$\mathbf{J}_L = \begin{bmatrix} \mathbf{0}_{1\times 4} & -2 \, \lambda_L \, (\mathbf{\Lambda}_L \, (\mathbf{q}_a - \mathbf{q}_a^*))^T \end{bmatrix} . \qquad (53)$$

A reasonable choice of $\mathbf{q}_a^*$ is to consider the configuration of maximum manipulability [46], which could be evaluated with the Jacobian from Eq. 25

$$w = \sqrt{\left| \mathbf{J}_{qa} \, \mathbf{J}_{qa}^T \right|}. \qquad (54)$$

We have searched for such configuration discretizing all possible arm joint positions and quadrotor inclinations (with ϕ_q and θ_q between the ranges of $[-pi/2, pi/2]$). Unfortunately, in our particular application, the configuration of largest manipulability leads to a configuration with structural self occlusion of the robot body onto the camera frame. During our experiments we have chosen instead desired configurations where the camera maximizes its field of view, i.e. below the quadrotor in either the front or rear parts of the robot.

As stated before, these hierarchical control law and visual servo approaches are contributions already done in the context of the thesis objectives. The remaining tasks involving global robot localization are considered as future work. With this aim, in the next section a brief description of the work plan and its tasks are presented.

6 Work plan

The proposed research is framed within the EU Project ARCAS FP7-287617 and is partially funded by the FPI-UPC grant 87-915, associated with this project. The work is being developed at the Institut de Robtica i Informtica Industrial (IRI), UPC-CSIC, in Barcelona, within the Mobile Robotics research group.

The work plan includes past, current and the expected future work, with the aim of fulfilling the initial objectives, and is divided into six main tasks, two of them are subdivided into two subtasks, as described below. A brief description of each task is provided, but it is only intended to be an orientation of the work required and the objectives of each task. The tasks directly related with contributions entail the formulation of the approaches, their validation and comparisons with other methods, both in simulations and real robot experiments, along the creation and publication of the corresponding Matlab, C++ and ROS (Robot Operating System) code.

The schedule of this plan spans over four years and is presented in Fig. 3 as a Gantt chart. In this chart, **Q1**, ..., **Q4** stand for the four quarters of a year. The work already completed is shown in green.

Task 1: Literature review

The literature review is recursive throughout the thesis work, although during certain periods of time, this task is carried out more intensively. Moreover, literature review of existing numerical methods underlying the approaches of the previous topics will be needed.

Task 2: Visual servoing

This task seeks to obtain a new and robust formulation for an image-based visual servoing method. In this approach we want to formulate a robust method without depending on the camera intrinsic parameters in order to work in situations with noisy calibration parameters and for unexpected changes in the camera zoom.

Task 3: MAV control

Driving a quadrotor with an arm attached below implies to deal with complex dynamic effects of the coupled body. With this, we plan to develop new control law techniques specifically designed for micro aerial manipulation vehicles. This task is divided in two subtasks. Firstly, a non-linear model predictive control (NMPC) that considers the quadrotor-arm dynamic model. Secondly, a hierarchical control law considering the overactuation of the robot.

Task 3.1: Non-linear Model Predictive Control (NMPC)

The NMPC will be based on iterative, finite horizon optimization of the quadrotor-arm model. Specifically, an online calculation will be used to explore control commands that emanate from the current state and find a cost-minimizing control strategy until a future time step.

Task 3.2: Task priority control

A visual servo approach, applied to a flying robot such as a quadrotor, carries the problem of underactuation. The quadrotor has only 4DOF while the camera velocities computed with the visual servo approach has 6DOF. In order to solve this issue, in this task we plan to build and attach a serial arm below the flying platform and exploit the redundancy DOFs obtained.

This filed of research has been popular with robotic arms but in our particular case we plan to extend it to aerial manipulators. It entails the formulation of the flying robot Jacobian as well as the hierarchical control law that takes into account task priority.

With the formulation of the hierarchical control law and the designed primary task corresponding to the visual servo control commands, a secondary task will be developed to compute the torques exerted on the quadrotor through the actuation of the arm when the overall CoG of the system is modified in order to improve flight behaviour and reduce undesired dynamic effects of the whole body. A third task will be provided to avoid arm singularities and to increase manipulability by setting the hand to a desired configuration.

Task 4: Odometry estimation

This task seeks to obtain the estimation of the vehicle 6DOF pose and velocities using sensor fusion techniques specially oriented to flying vehicles. We aim to propagate the flying robot state using the IMU at high frame rate and correct those measurements with a new optical flow sensor. With this method we will obtain an estimation of the odometry of the vehicle that can be useful in the subsequent tasks such as localization of the robot. The validation of this approach will be experimental and a comparison with an accurate fast-rate localization Vicon system will be provided.

Task 5: MAV localization

To navigate in the scenario we plan to develop new localization techniques. This task is divided in two subtasks, on the one hand, a coarse localization approach will be devised together with a fine localization technique.

Task 5: Coarse localization and mapping

The visual guidance of a quadrotor is based on the observation with the cameras of a specific target, and then to control the MAV with respect to it. Sometimes this target will not be in the camera field of view. To solve this problem we plan to use radio beacons to obtain a coarse localization of the robot and the target. The experimental validation of the approach will be done in both indoor and outdoor scenarios.

Task 5: Fine localization

In the areas where the robot should carry out the main task, localization should be more precise. To that end, we plan to fuse the previous presented visual servo and visual odometry techniques to allow insite operations. A four months international stage is envisaged for the complettion of this task.

Task 6: Thesis write up

The last task of the research is dedicated to the write up of the thesis manuscript, and the preparation of its public defense.

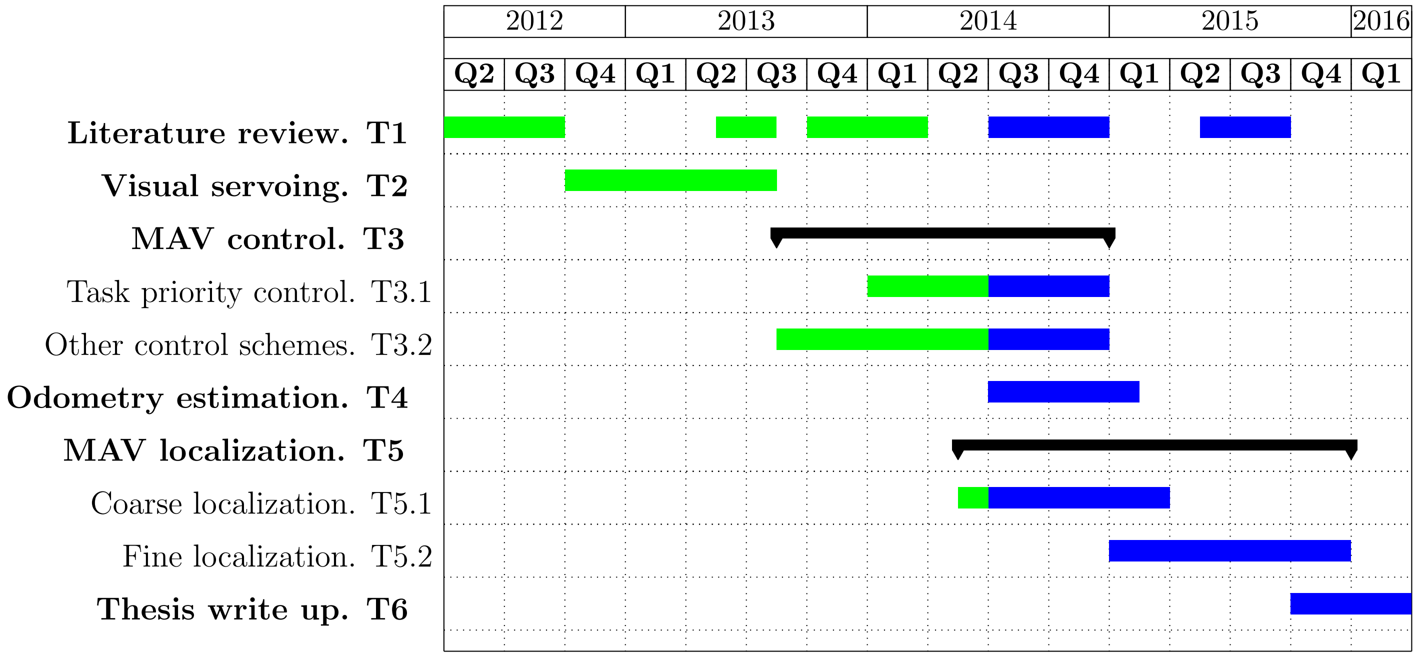

Figure 3: Work plan of the proposed work

7 Publications

Currently accepted or submitted publications resulting from the proposed research.

Workshops

A. Santamaria-Navarro and J. Andrade-Cetto. *Hierarchical Task Control for Aerial Inspection.* euRathlon/ARCAS Workshop and Summer School, 15-18th June 2014, Sevilla. [42]

Conferences

A. Santamaria-Navarro and J. Andrade-Cetto. *Uncalibrated image-based visual servoing.* 2013 IEEE International Conference on Robotics and Automation, 2013, Karlsruhe, pp. 5247-5252. [41]

Journals

A. Santamaria-Navarro, E.H. Teniente, M. Morta and J. Andrade-Cetto. *Terrain classification in complex 3D outdoor environments.* Journal of Field Robotics. To appear. [40]

8 Resources

The above methods and technologies will be integrated in the ARCAS flying robot system that will be validated in the following scenarios:

1. Simulation testbed: A free-flying ROS and Gazebo simulation using multiple robot settings.
2. Indoor testbeds: The indoor testbeds will consists on two arenas. The flying arena of the Institut de Robtica i Informtica Industrial (IRI), CSIC-UPC (Barcelona), shown in Fig. 4(a); and the ARCAS partner CATEC tesbed equipped with a complete VICON system and facilities (Sevilla), shown in Fig. 4(b).
3. Outdoor scenario with helicopters: Scenarios included in the ARCAS project integrations and demonstrations provided by the ARCAS partner DLR (Munich, Germany).

In order to conduct the major part of the experiments and to test the presented approaches, we mainly use an ASCTEC pelican quadrotor called Kinton (http://wiki.iri.upc.edu/index.php/Kinton). Kinton is currently equiped with an onboard PC embedded Intel atom at 1.6GHz, and several sensoring devices such as two cameras, an IMU or GPS. A new low-weight arm prototype of 6 DOFs is currently developed by IRI-Workshop in order to give kinton some

(a) IRI (b) CATEC

Figure 4: Flying arena testbeds

aerial manipulation capabilities. Fig. 5 shows Kinton with the arm attached below which is currently under development.

Figure 5: Kinton with 6DOF arm attached below.

References

[1] Optitrack. [online] `http://www.naturalpoint.com/optitrack`.

[2] Vicon. [online] `http://www.vicon.com`.

[3] G. Antonelli. Stability analysis for prioritized closed-loop inverse kinematic algorithms for redundant robotic systems. *IEEE Trans. Robotics*, 25(5):985–994, 2009.

[4] G. Antonelli, E. Cataldi, P.R. Giordano, S. Chiaverini, and A. Franchi. Experimental validation of a new adaptive control scheme for quadrotors MAVs. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pages 2439–2444, Tokyo, Nov. 2013.

[5] P. Baerlocher and R. Boulic. Task-priority formulations for the kinematic control of highly redundant articulated structures. *J. Intell. Robotic Syst.*, 1:323–329, 1998.

[6] R. Boulic, R. Mas, and D. Thalmann. A robust approach for the control of the center of mass with inverse kinetics. *Comput. Graph.*, 20(5):693–701, 1996.

[7] F. Chaumette. Potential problems of stability and convergence in image-based and position-based visual servoing. In *The Confluence of Vision and Control*, volume 237 of *Lect. Notes Control Info. Sci.*, pages 66–78, 1998.

[8] F. Chaumette and S. Hutchinson. Visual servo control. I. Basic approaches. *Robotics Autom. Mag.*, 13(4):82–90, 2006.

[9] F. Chaumette and S. Hutchinson. Visual servo control. II. Advanced approaches. *Robotics Autom. Mag.*, 14(1):109–118, 2007.

[10] S. Chiaverini. Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. *IEEE Trans. Robotics Autom.*, 13(3):398–410, 1997.

[11] P. I. Corke and S. A. Hutchinson. A new hybrid image-based visual servo control scheme. In *Proc. IEEE Conf. Decision Control*, pages 2521–2527, Sydney, Dec. 2000.

[12] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Trans. Robotics Autom.*, 8(3):313–326, 1992.

[13] A. Farahmand, A. Shademan, and M. Jagersand. Global visual-motor estimation for uncalibrated visual servoing. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pages 1969–1974, San Diego, Nov. 2007.

[14] S. Foix, G. Alenyà, and C. Torras. Lock-in time-of-flight (ToF) cameras: A survey. *IEEE Sensors J.*, 11(9):1917–1926, Sep. 2011.

[15] T. Hamel and R. Mahony. Image based visual servo control for a class of aerial robotic systems. *Automatica*, 43(11):1975–1983, 2007.

[16] D. Honegger, M. Lorenz, P. Tanskanen, and M. Pollefeys. An open source and open hardware embedded metric optical flow cmos camera for indoor and outdoor applications. In *Proc. IEEE Int. Conf. Robotics Autom.*, pages 1736,1741, Karlsruhe, May 2013.

[17] K. Hosoda and M. Asada. Versatile visual servoing without knowledge of true Jacobian. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pages 186–193, Munich, Sep. 1994.

[18] S. Hutchinson, G. D. Hager, and P.I. Corke. A tutorial on visual servo control. *IEEE Trans. Robotics Autom.*, 12(5):651–670, 1996.

[19] C. Korpela, M. Orsag, T. Danko, B. Kobe, C. McNeil, R. Pisch, and P. Oh. Flight stability in aerial redundant manipulators. In *Proc. IEEE Int. Conf. Robotics Autom.*, pages 3529–3530, Saint Paul, May 2012.

[20] C.M. Korpela, T.W. Danko, and P.Y. Oh. Designing a system for mobile manipulation from an unmanned aerial vehicle. In *Proc. IEEE Int. Conf. Tech. Pract. Rob. App.*, pages 109–114, Boston, Apr. 2011.

[21] V. Lippiello, R. Mebarki, and F. Ruggiero. Visual coordinated landing of a UAV on a mobile robot manipulator. In *Proc. 11th IEEE Int. Sym. Saf. Secur. Rescue Robotics*, pages 1–7, Linkoping, Oct. 2013.

[22] V. Lippiello and F. Ruggiero. Cartesian impedance control of a uav with a robotic arm. In *Proc. IFAC World Congr. Auton. Control*, volume 10, pages 704–709, Dubrovnik, Sep. 2012.

[23] V. Lippiello and F. Ruggiero. Exploiting redundancy in Cartesian impedance control of UAVs equipped with a robotic arm. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pages 3768–3773, Vilamoura, Oct. 2012.

[24] R. Mahony, V. Kumar, and P. Corke. Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *Robotics Autom. Mag.*, 19(3):20–32, 2012.

[25] E. Malis, F. Chaumette, and S. Boudet. 2-1/2-D visual servoing. *IEEE Trans. Robotics Autom.*, 15(2):238–250, 1999.

[26] L. Marconi, F. Basile, G. Caprari, R. Carloni, P. Chiacchio, C. Hurzeler, V. Lippiello, R. Naldi, J. Nikolic, B. Siciliano, S. Stramigioli, and E. Zwicker. Aerial service robotics: The airobots perspective. In *Proc. 2nd Int. Conf. Appl. Robotics Power Ind.*, pages 64–69, Zurich, Sep. 2012.

[27] D. Mellinger, Q. Lindsey, M. Shomin, and V. Kumar. Design, modeling, estimation and control for aerial grasping and manipulation. In *Proc.*

IEEE/RSJ Int. Conf. Intell. Robots Syst., pages 2668–2673, San Francisco, Sep. 2011.

[28] F. Moreno-Noguer, V. Lepetit, and P. Fua. EPnP: An accurate O(n) solution to the PnP problem. *Int. J. Comput. Vision*, 81(2):155–166, 2009.

[29] Y. Nakamura. *Advanced Robotics: Redundancy and Optimization*. Addison-Wesley, 1990.

[30] Y. Nakamura, H.Hanafusa, and T. Yoshikawa. Task-priority based redundancy control of robot manipulators. *Int. J. Robotics Res.*, 6(2):3–15, 1987.

[31] A. Ollero and K. Kondak. 10 years in the cooperation of unnamed aerial systems. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pages 5450–5451, Vilamoura, Oct. 2012.

[32] M. Orsag, C. Korpela, and P. Oh. Modeling and control of MM-UAV: Mobile manipulating unmanned aerial vehicle. *J. Intell. Robotic Syst.*, 69(1-4):227–240, 2013.

[33] A. Ortega and J. Andrade-Cetto. Segmentation of dynamic objects from laser data. In *Proc. Eur. Conf. Mobile Robots*, pages 115–121, Orebro, Sep. 2011.

[34] I. Palunko, P. Cruz, and R. Fierro. Agile load transportation: Safe and efficient load manipulation with aerial robots. *Robotics Autom. Mag.*, 19(3):69–79, 2012.

[35] A. Penate-Sanchez, J. Andrade-Cetto, and F. Moreno-Noguer. Exhaustive linearization for robust camera pose and focal length estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(10):2386–2400, 2013.

[36] J.A. Piepmeier, B.A. Gumpert, and H. Lipkin. Uncalibrated eye-in-hand visual servoing. In *Proc. IEEE Int. Conf. Robotics Autom.*, pages 568–573, Washington, May 2002.

[37] J.A. Piepmeier, G.V. McMurray, and H. Lipkin. Uncalibrated dynamic visual servoing. *IEEE Trans. Robotics Autom.*, 20(1):143– 147, 2004.

[38] P.E.I. Pounds, D.R. Bersak, and A.M. Dollar. Grasping from the air: Hovering capture and load stability. In *Proc. IEEE Int. Conf. Robotics Autom.*, pages 2491–2498, Shanghai, May 2011.

[39] A. Sanfeliu, J. Andrade-Cetto, M. Barbosa, R. Bowden, J. Capitán, A. Corominas Murtra, A. Gilbert, J. Illingworth, L. Merino, Josep M. Mirats Tur, P. Moreno, A. Ollero, J. Sequeira, and M.T. Spaan. Decentralized sensor fusion for ubiquitous networking robotics in urban areas. *Sensors*, 10(3):2274–2314, Mar. 2010.

[40] A. Santamaria, E.H. Teniente, M. Morta, and J. Andrade-Cetto. Terrain classification in complex 3D outdoor environments. *J. Field Robotics*, 2014. To appear.

[41] A. Santamaria-Navarro and J. Andrade-Cetto. Uncalibrated image-based visual servoing. In *Proc. IEEE Int. Conf. Robotics Autom.*, pages 5247–5252, Karlsruhe, May 2013.

[42] A. Shademan, A. Farahmand, and M. Jagersand. Robust Jacobian estimation for uncalibrated visual servoing. In *Proc. IEEE Int. Conf. Robotics Autom.*, pages 5564–5569, Anchorage, May 2010.

[43] B. Steder, G. Grisetti, C. Stachniss, and W. Burgard. Visual slam for flying vehicles. *IEEE Trans. Robotics*, 24(5):1088–1093, Oct 2008.

[44] L.E. Weiss, A.C. Sanderson, and C.P. Neuman. Dynamic visual servo control of robots: An adaptive image-based approach. In *Proc. IEEE Int. Conf. Robotics Autom.*, pages 662–668, St. Louis, Mar. 1985.

[45] T. Yoshikawa. Manipulability and redundancy control of robotic mechanisms. In *Proc. IEEE Int. Conf. Robotics Autom.*, volume 2, pages 1004–1009, St. Louis, Mar. 1985.