

An educational simulation platform for Unmanned Aerial Vehicles aimed to detect and track moving objects

Giuseppe Silano and Luigi Iannelli

1. Motivation

During the last ten years, much effort has been put into the research field of (semi) autonomous unmanned aerial vehicles (UAVs). Thus, by considering the strong increase of the use of UAVs for

- inspection and surveillance purposes,
- detecting and tracking arbitrary moving objects,

it follows the **need for tools** that allow to understand what it happens when some new applications are going to be developed.

The Goal: a complete software platform that gives the possibility to test different algorithms for UAV moving in a simulated 3D environment is more and more important for the whole design process, as well as for the educational purposes.

2. System Description

Several tools (either open source or proprietary) are available: Gazebo, V-REP, Webots, We looked at the “simplest” (familiar with) solution for control engineering students: Matlab/Simulink and the **MathWorks Virtual Reality Toolbox**.

The toolbox simulates a 3D world to observe the interaction between complex dynamic systems and the surrounding environment. In particular the tool has been employed for simulating a drone following a car. Given the particular scenario, we started from one of the example available on the MathWorks platform describing a quite detailed model of the car dynamics where the car moves along a non trivial path.

The key idea: the camera position and orientation is replaced by the drone position and orientation so as determined by the dynamical equations [1].

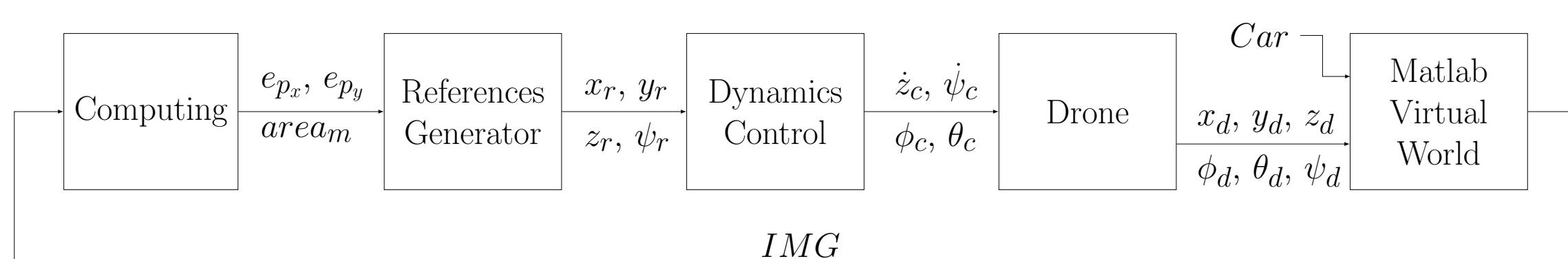


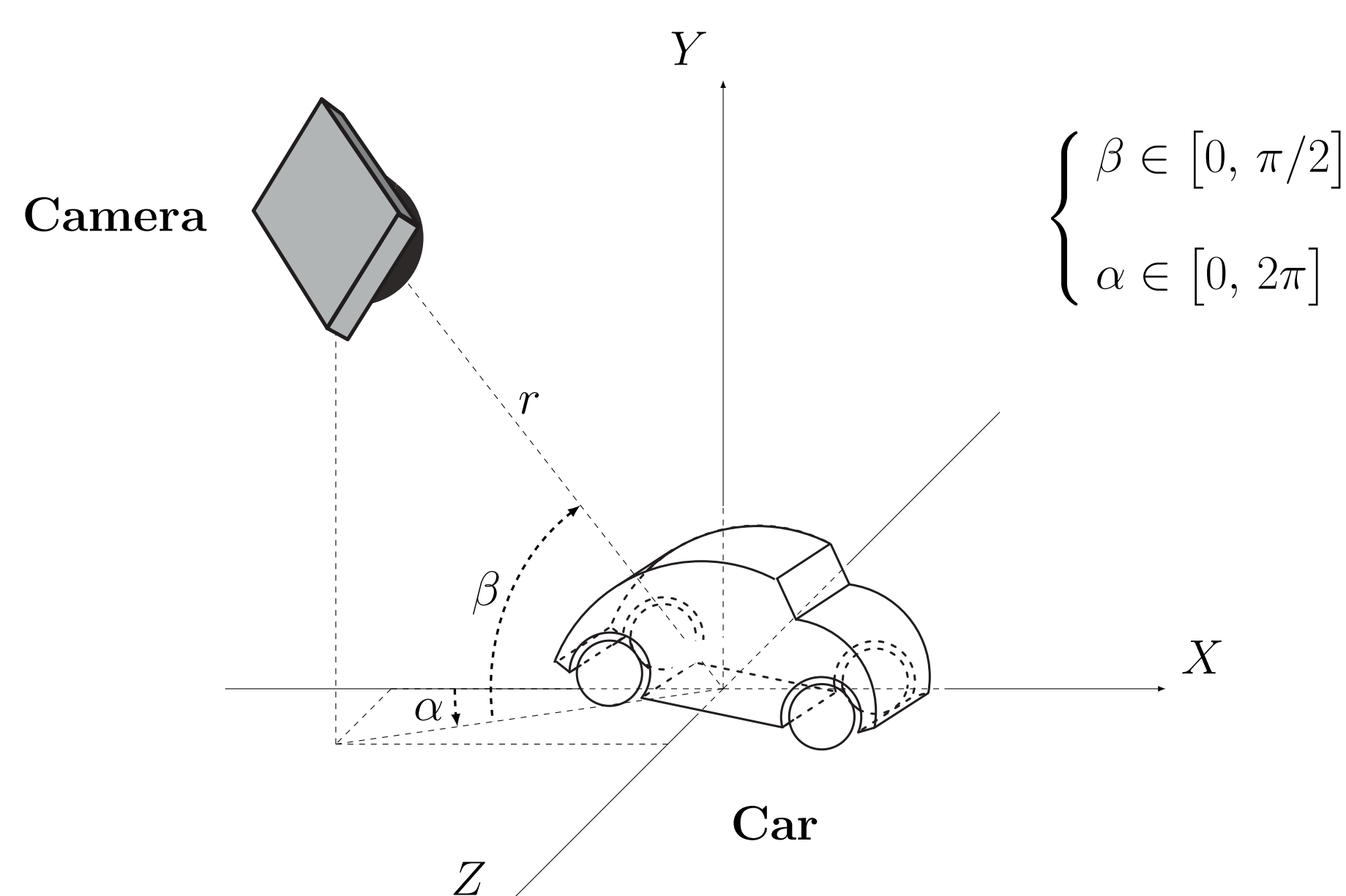
Figure 1: The control scheme. Subscript c indicates the commands, r indicate the references and d indicate the drone.

3. Vision Based Target Detection

The camera extends the aircraft sensory capacity and, through that, it is possible to develop an automatic control that commands the UAV using the image-based visual servoing approach. The vision based target detection has been subdivided in three phases:

- **CLASSIFIER LEARNING PHASE**, in which the machine learning technique (Viola & Jones algorithm) is trained in order to detect the target.

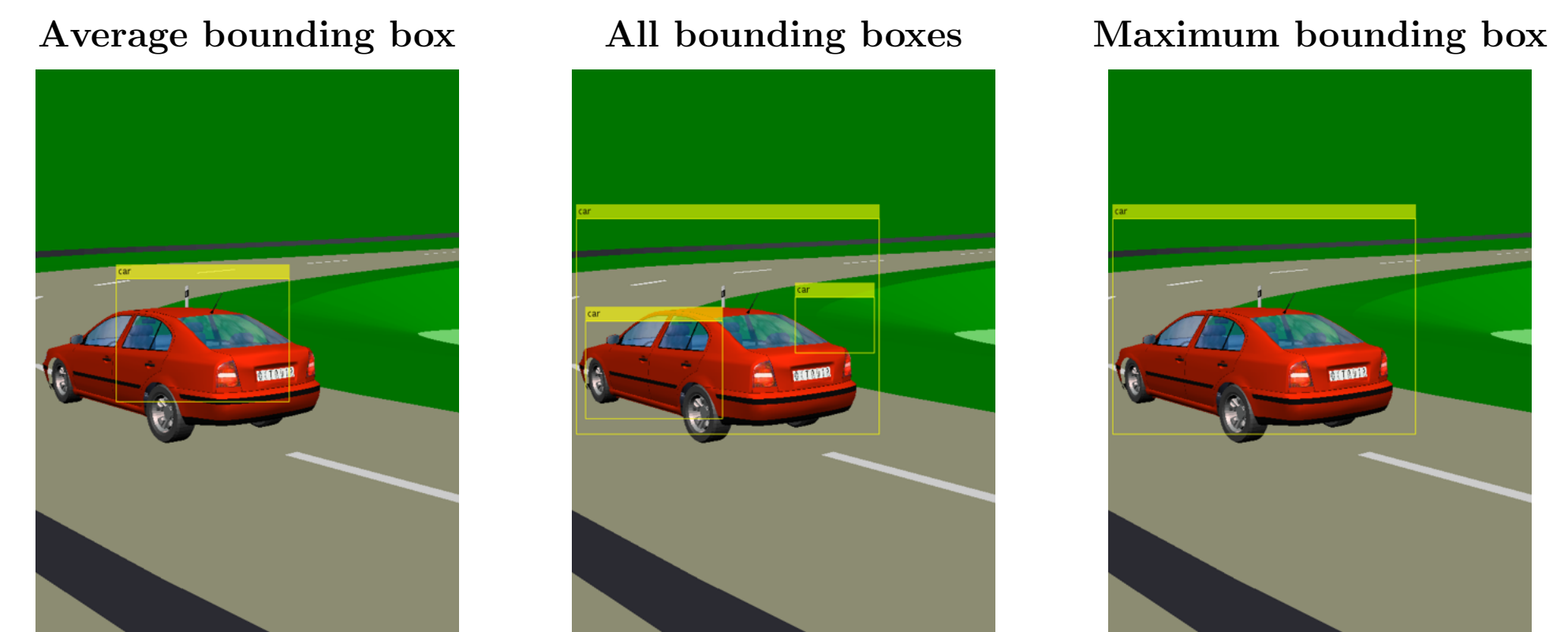
A Matlab script manages the frames acquisition and the computing phase. To this aim it has been simulated the drone moving along a spiral trajectory around the car parked in its initial state. The script takes into account the slight differences between the classic fixed and virtual world reference system.



A high number of images were needed in order to train the classifier. The images have been divided into two groups: positive (that contains the target, 2626) and negative images (5252) achieving a 1 : 2 ratio.

- **BOUNDING BOX SELECTION**, where an algorithm is designed to obtain a unique box surrounding the target.

The car is only partially detected in spite of the high images number used in the learning process, although there are no revelation errors. On the other hand, they introduce enough “useful noise” to help the detection.



The recognition of different bounding boxes requires an algorithm in order to obtain a unique box surrounding the target. A Matlab script computes the “maximum” bounding box achieved from the classifier.

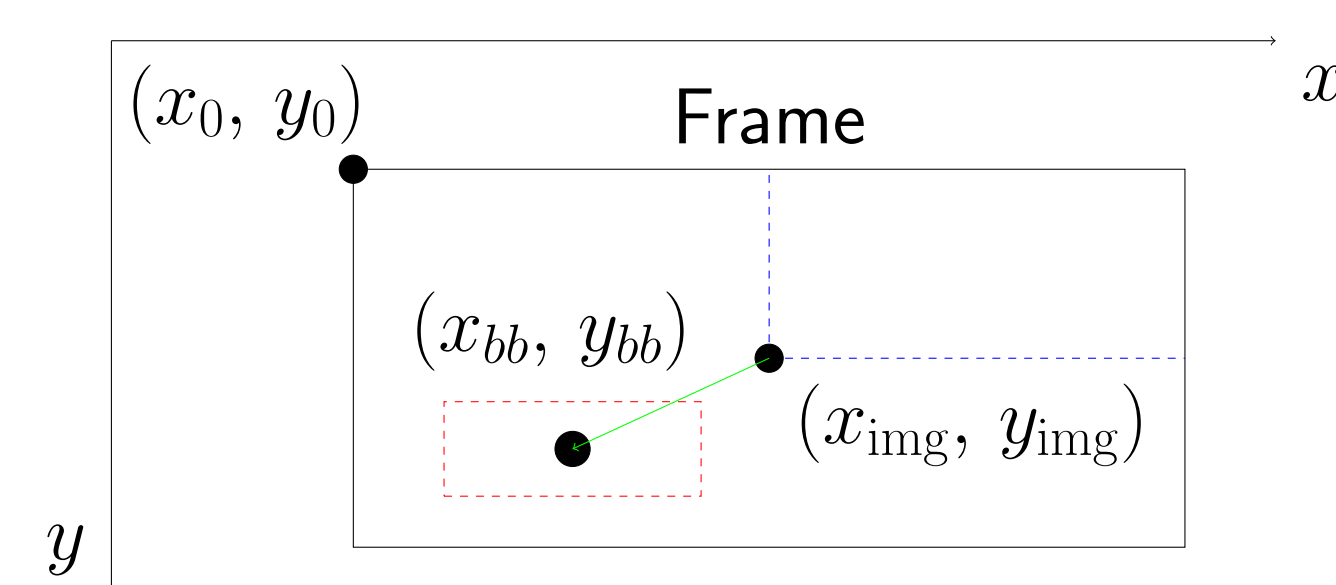
- **TRACKING ALGORITHM**, the classifier is replaced by the tracking algorithm to reduce the computational burden.

The classifier only at the first step or in case of partial occlusion is used to detect the target (the car). Otherwise, a Continuously Adaptive Mean-Shift (CAMShift) algorithm performs tracking by searching for the probability distribution pattern of a target in a local adaptive size window within the frame.

4. Flight Control System

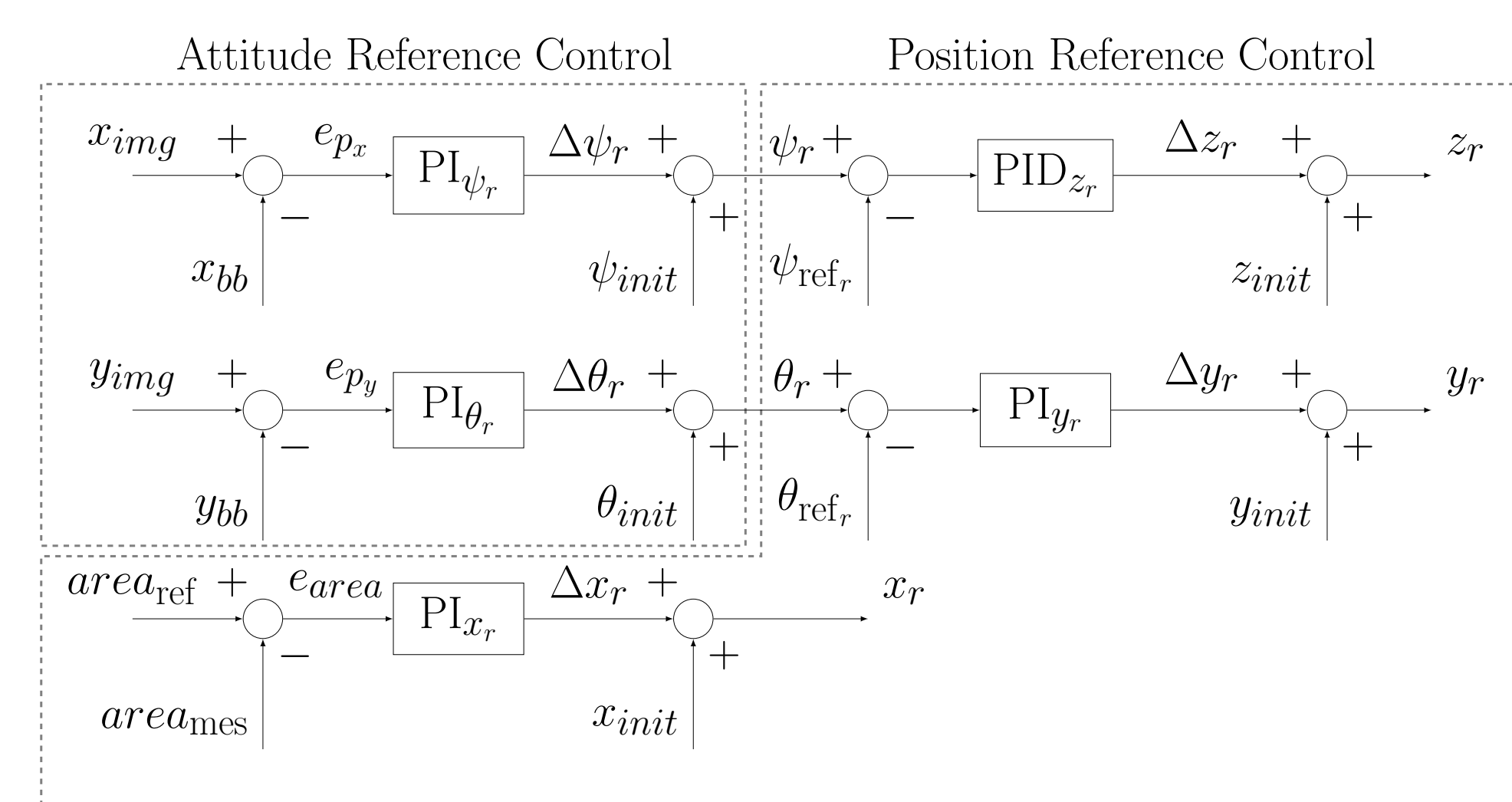
In our application we considered a drone with four rotors, and a pose controller has been designed based on a classical dynamic model, as described in [1]. The flight control system has been split into two parts:

- **REFERENCE GENERATOR**, that uses the information extracted from the images to generate the path to follow.



In the following phase the image (x_{img}, y_{img}) and the bounding box (x_{bb}, y_{bb}) centroids are computed, as well as the distance vector between the centroids and the bounding box area, so as in [2].

The references generator is decomposed into two parts: the attitude and position controller. It tunes the angles $(\psi_d, \phi_d$ and $\theta_d)$ trying to overlap the images and the bounding box centroids. The angles are later used to tune the reference position.



- **INTEGRAL BACKSTEPPING**, used as the controller for the trajectory path tracking.

The trajectory control strategy works for making the aircraft attitude and position to follow the reference generator outputs.

References

- [1] S. Bouabdallah and R. Siegwart, Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. In *Proceedings of the 2005 IEEE International Conference in Robotics and Automation*, pages 2247–2252, April 2005.
- [2] J. Pestana, J.L. Sanchez-Lopez, S. Saripalli, and P. Campoy. Computer vision based general object following for gps-denied multirotor unmanned vehicles. In *American Control Conference*, pages 1886–1891, 2014.