I. UAV CONTROL

The visual tracking system in the previous section provides the control algorithm with a vector of normalized image coordinates for every track in the camera field of view. The control algorithm activates follow mode when there exists a target with the ID that a human operator has assigned for following. When the given ID is not found in the vector that the tracker provides, the control algorithm commands the UAV to hold its position until another target ID that exists among the tracks is assigned to follow. In this section, the control algorithm is described in more detail.

A. Coordinate Frame Convention

Before giving a detailed explanation of the control algorithm, it is worth clarifying our assumptions and the coordinate frames used.

First, an East-North-Up (ENU) coordinate frame is used as opposed to the common North-East-Down (NED) coordinates for UAV [**?**] in order to match the frame convention used in the `mavros` package in ROS [**?**]. Let $\mathcal{F}^i$ be the inertial frame, which in this case coincides with the ENU frame, and let $\mathcal{F}^v$ be the vehicle frame that is translated to the UAV center of mass, with the same orientation as $\mathcal{F}^i$. Vehicle-1 frame, $\mathcal{F}^{v1}$ indicates the frame that is only rotated about the z-axis of $\mathcal{F}^v$ by ψ, the heading angle of the multirotor. The rotation matrix from $\mathcal{F}^v$ to $\mathcal{F}^{v1}$ can be expressed as R_v^{v1}. Other involved frames are optical, camera, and body frames expressed as $\mathcal{F}^o$, $\mathcal{F}^c$, $\mathcal{F}^b$, respectively.

Second, a flat-earth model is used to properly scale the target position relative to the camera in $\mathcal{F}^{v1}$ and we have access to the correct altitude information.

Third, the displacement between the center of mass of the UAV and the focal point of camera is ignored since it is negligible compared to the distance between the camera and the target.

Fourth, we rely on GPS position controller on autopilot. The visual-servoing controller in this paper computes the desired multirotor position in order to follow the target and send the position command to the autopilot.

B. Forward and heading motion control

The first step of the motion control is to transform the line of sight (LOS) vector to the target in $\mathcal{F}^o$ into $\mathcal{F}^{v1}$. Let

$$\ell^o = [\epsilon_x^o, \epsilon_y^o, 1]^\top \tag{1}$$

where ℓ^o is the normalized line of sight vector in $\mathcal{F}^o$ and its third element 1 indicates the focal length of the camera in the normalized image. In our case, this is the information provided by the R-RANSAC tracker. Let also the unit vector along the optical axis in $\mathcal{F}^o$ be defined as

$$\mathbf{m}^o = [0, 0, 1]^\top. \tag{2}$$

By applying sequential transformations to ℓ^o and $\mathbf{m}^o$, we get

$$\ell^{v1} = R_b^{v1}(\phi,\theta)R_c^b(\alpha)R_o^c\ell^o = [\ell_x^{v1}, \ell_y^{v1}, \ell_z^{v1}]^\top \tag{3}$$

$$\mathbf{m}^{v1} = R_b^{v1}(\phi,\theta)R_c^b(\alpha)R_o^c\mathbf{m}^o = [m_x^{v1}, m_y^{v1}, m_z^{v1}]^\top \tag{4}$$

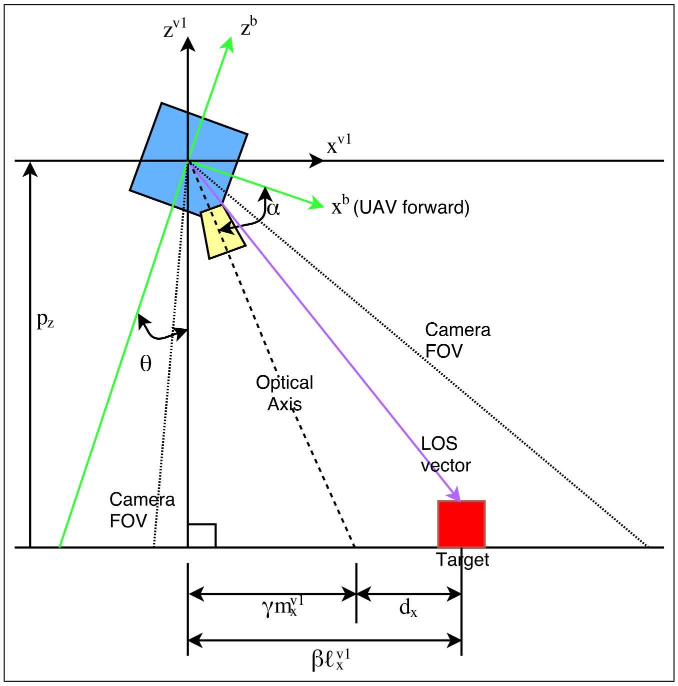

Fig. 1: Side view of the multirotor.

where R_c^b is a matrix with fixed values depending on how the camera is mounted with respect to $\mathcal{F}^b$, R_b^{v1} is a matrix requiring the roll and pitch angles of the multirotor. The ℓ^{v1} is the displacement of the target relative to the multirotor and the $\mathbf{m}^{v1}$ is the displacement of the projection of the optical axis to the ground relative to the multirotor. The LOS vector ℓ^{v1} and $\mathbf{m}^{v1}$ do not have proper scalings due to the unknown depth information to the target in $\mathcal{F}^o$, but can be recovered using the altitude of the camera. Let

$$\beta = \frac{P_z}{\ell_z^{v1}} \tag{5}$$

$$\gamma = \frac{P_z}{m_z^{v1}} \tag{6}$$

where P_z is the altitude of the multirotor. Then, the desired forward position from the current multirotor position can be computed as

$$d_x = \beta\ell_x^{v1} - \gamma m_x^{v1}. \tag{7}$$

This d_x may be further broken down into east and north components

$$d_n = d_x \sin(\psi) \tag{8}$$

$$d_e = d_x \cos(\psi) \tag{9}$$

where ψ is the heading of the multirotor. These north and east components are added to the current multirotor east and north positions, and the sum is sent to the autopilot position controller.

It is more suitable to compensate for a target moving horizontally in the image plane by adjusting the multirotor's heading than through lateral motion. Thus, a yaw rate command ω_z can be computed as

$$\omega_z = \eta\ell_y^{v1}, \tag{10}$$

where $\eta > 0$ is a control gain.