Attitude Heading Reference System with Rotation-Aiding Visual Landmarks

Chris Beall, Duy-Nguyen Ta, Kyel Ok, and Frank Dellaert
Center for Robotics and Intelligent Machines, Georgia Institute of Technology

***Abstract*—In this paper we present a novel vision-aided attitude heading reference system for micro aerial vehicles (MAVs) and other mobile platforms, which does not rely on known landmark locations or full 3D map estimation as is common in the literature. Inertial sensors which are commonly found on MAVs suffer from additive biases and noise, and yaw error will grow without bounds. The bearing-only measurements, which we call vistas, aid the vehicle's heading estimate and allow for long-term operation while correcting for sensor drift. Our method is experimentally validated on a commercially available low-cost quadrotor MAV.**

I. Introduction

Over the past few years micro aerial vehicles (MAVs) have become increasingly popular robotic platforms due to advances in built-in sensor capabilities and greatly reduced cost. These devices typically come with a variety of built-in sensors such as cameras, accelerometers, gyroscopes, and sophisticated state estimation algorithms that run on an onboard computer. MAVs are very well suited for applications such as exploration in cluttered indoor environments, where wheeled platforms would have to negotiate around obstacles. A drawback is the small payload the vehicles can carry, which limits the number and type of sensors that can be used.

Quadrotors are inherently unstable, and therefore generally rely on an inertial measurement unit (IMU) comprised of a three-axis gyroscope and three-axis accelerometer for stability and flight control. Since the accelerometer measurement contains gravity, the pitch and roll can be easily computed at any time. However, the IMU does not provide information about absolute heading (yaw), and given the integration of gyro measurements over time, the heading error estimate can grow without bounds. This can be overcome by adding a magnetometer, or by relying on GPS to compute absolute heading over time when the vehicle is moving. When a magnetometer is not available, or in GPS-denied environments such as indoors, a different aiding approach becomes necessary.

In this paper we present a novel *attitude heading reference system* (AHRS) which uses bearing-only visual measurements, to aid the IMU estimation, without the need for estimating the actual 3D position of the visual landmarks themselves. We call our aiding measurements vistas, as shown in Fig. 1.Vision-aided inertial navigation systems that have been demonstrated in the literature rely on known landmark locations for visual aiding, or build up a full 3D map of the environment. Our technique does not rely on known landmarks or build up a map, and instead takes advantage of the fact that landmarks which are sufficiently distant from the camera are ideal for rotation estimation.

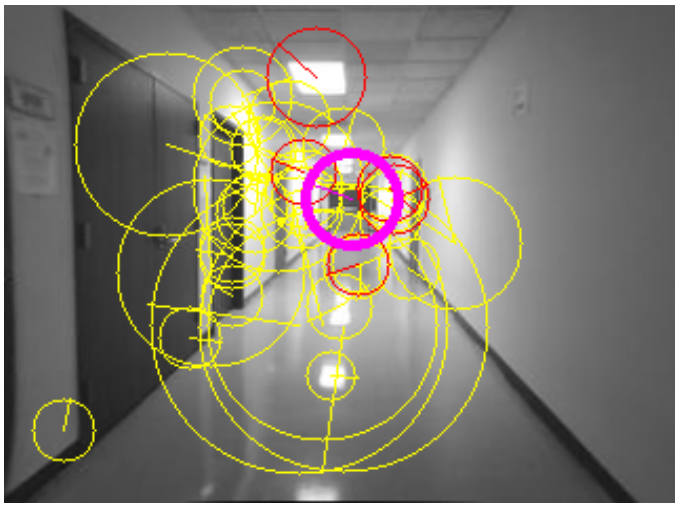


Figure 1: Detected *rotation-aiding landmarks/vistas* (in red) and features that do not satisfy the *selection* criteria (in yellow) are shown. One of the *vistas* is selected (bold circle) as the aiding landmark. Figure is best viewed in color.

In what follows, we first discuss related work, after which e introduce our rotation-aiding visual landmarks in Sec. II. Sec. III presents the AHRS with rotation-aiding landmarks for drift correction, followed by experimental evaluation in sec. IV.

A. Related Work

Vision is frequently used for robot pose estimation and navigation, especially in indoor environments which are typically GPS-denied. Many of the techniques that have been proposed for MAVs in the literature actually rely on vision alone, and there are many similarities to structure from motion (SFM) and simultaneous localization and mapping algorithms.

For example, [8] builds an incremental map during flight by using a downward looking monocular camera. This method assumes that the area below the MAV is rich in texture, and therefore may fail in many environments. Another approach is to exploit the known structure of the environment to be mapped as in [7], where the authors detect the vanishing point at the end of the hallway by finding intersection of long lines along the corridors. [26] also exploits hallway specific properties such as high entropy, symmetry, self-similarity, etc. to infer the direction of a corridor being traveled by a robot.

The type of approach we are interested in relies on inertial sensors to estimate the vehicle pose by numerical integration, while aiding with vision to correct for drift. Wu et al. [28]

proposed a system that tracks the location and size of a visual marker of known location and size with respect to the aerial vehicle to estimate its own pose . In [16] IMU drift was corrected by tracking monocular image features during flight. A similar approach is taken in [2], where a full feature-based map was constructed while incorporating IMU data, and estimating sensor biases. Several authors have proposed systems that fuse inertial measurements with pose information obtained through feature tracking, and showed results for planetary lander guidance, as well as ground vehicle navigation [27], [24], [25].

These and many other techniques rely on robust data association for long-term feature tracking, or exploit environmental properties which limit where the algorithm can be expected to work. Conte et al. [11] proposed the use of geo-referenced aerial or satellite imagery to aid the inertial navigation system of an unmanned aerial vehicle, applicable in the event of GPS failure. Bearing-only feature measurements have been used for MAV navigation in [21], [10], however no IMU aiding was done.

Many of these techniques rely on robust data association for long-term feature tracking, or exploit environmental properties which limit where the algorithm can be expected to work.

Inertial and vision is also used to estimate egomotion. [5] fuses inertial and vision data to estimate egomotion by using multirate EKF and multirate UKF to deal with the problem that these two types of data have different update rates. [15] extends [5] by also estimating the scene structure.

An important question to address when fusing inertial and vision sensor data is which states are observable, and which parameters are identifiable using just inertial and visual measurements. Recently, [19] has presented a theoretical answer to this question in an inertial-vision fusion SLAM system. [23] similarly determines the observability of each states in different cases, and derives a closed-form solution to estimate these states from visual and inertial measurements. There have also been specialized workshops held about fusing vision and inertial sensors, for example [13].

II. Vistas: Rotation-Aiding Landmarks

A. Minimum Depth Condition for Vistas

In this section, we derive conditions for landmarks such that they can be used to aid rotation estimates. To be useful for rotation estimation the projection of these features in the camera should depend only on rotation, and be independent of translation. For example, it is well-known that if a point is at infinity, its projection in the camera image can be purely determined by the camera rotation, whereas the camera translation has no effect on the landmark position in the image. Hence those special points, such as vanishing points, are normally used to determine the camera rotation in practice [4].

We call these rotation-aiding landmarks **vistas**. Our main observation is that, due to the limited resolution of the camera, these landmarks do not need to be at infinity to be useful for aiding rotation: they only need to be far enough such that effects of the camera translation on their projections are indiscernible. The coordinate systems used in the explanations that follow are shown in Fig. 2.

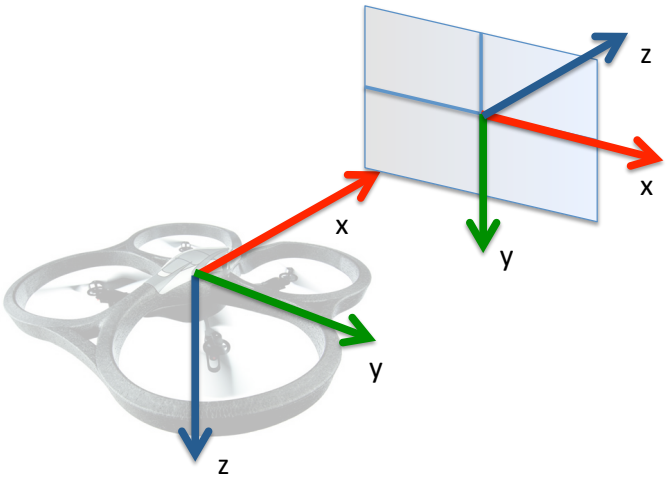


Figure 2: Body and camera coordinate frames

Let us consider a landmark $\mathbf{P}$ in space and assume that the camera is moving under a motion $\mathbf{X}_2^1 = \{\mathbf{R}, \mathbf{t}\} \in \mathbb{SE}3$ from frame 1 to frame 2. Furthermore, let $\mathbf{p_1} \in \mathbb{R}^3$ be the 2D homogeneous coordinate of the projection of $\mathbf{P}$ in camera 1, $\mathbf{p}_2^r$ be its projection in camera 2 under the pure camera rotation $\mathbf{R}$, and $\mathbf{p}_2^t$ be its projection in camera 2 under the full camera motion $\{\mathbf{R}, \mathbf{t}\}$. In order for $\mathbf{t}$ to have *indiscernible* effect on $\mathbf{P}$'s projection, the distance between $\mathbf{p}_2^t$ and $\mathbf{p}_2^r$ should be small enough to be undetectable. We derive the condition for $\mathbf{P}$ based on this requirement.

Since $\mathbf{p}_2^t$ and $\mathbf{p}_2^r$ are in homogeneous form, one way to enforce such a constraint in homogeneous space is to have the angle between these two vectors to be less than a threshold, or equivalently, $\cos(\mathbf{p}_2^r, \mathbf{p}_2^t) \geq \epsilon$, where ϵ is the cosine of the minimum angle between any two neighboring pixels in the image. Unfortunately, although this scheme makes it easy to derive the requirements for $\mathbf{P}$, it has numerical instability issue, because the minimum angle between two neighboring pixels in the image is so small that ϵ is very close to 1.

We choose to enforce the non-homogeneous distance between $\mathbf{p}_2^r$ and $\mathbf{p}_2^t$ to be less than 1 pixel so that the $\mathbf{p}_2^r$ and $\mathbf{p}_2^t$ are indistinguishable given the camera image resolution:

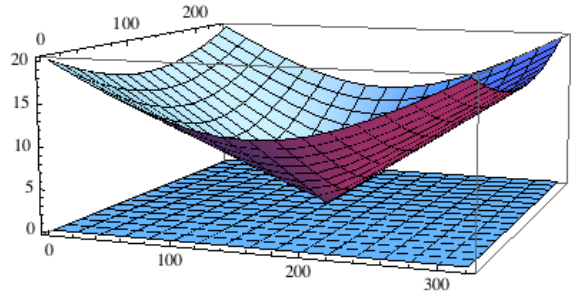
$$||\frac{1}{z_2^r}\mathbf{p}_2^r - \frac{1}{z_2^t}\mathbf{p}_2^t||^2 \leq 1, \tag{1}$$

where z_2^r and z_2^t are the third components of $\mathbf{p}_2^r$ and $\mathbf{p}_2^t$, respectively. Let the camera calibration matrix be defined as

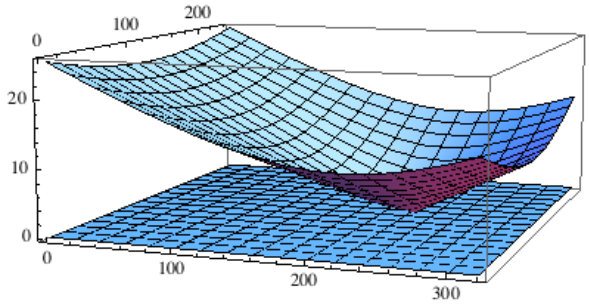
$$\mathbf{K} = \begin{bmatrix} f_x & 0 & o_x \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

When $\mathbf{P}$ is at infinity or if the camera motion is under a pure rotation ($\mathbf{t} = 0$), the projections $\mathbf{p}_1$ and $\mathbf{p}_2^r$ are related by the infinite homography $\mathbf{H} = \mathbf{K}\mathbf{R}_1^2\mathbf{K}^1$ between the two images [17]:

$$\mathbf{p}_2^r \sim \mathbf{K}\mathbf{R}_1^2\mathbf{K}^{-1}\mathbf{p}_1, \tag{3}$$



(a) $t_x = t_y = 0,\ t_z = 0.1$



(b) $t_x = 0.03,\ t_y = 0,\ t_z = 0.1$

Figure 3: Minimum Z^r_{1min} distances for vistas. The horizontal xy-plane is the image pixel coordinate, and the vertical z-axis is the minimum Z_1 required at each pixel. Plot with camera calibration: $o_x = 160,\ o_y = 120,\ f_x = f_y = 210$.

where $\mathbf{R}_1^2 = \mathbf{R}^\top$, and $\sim$ denotes equivalent up to a constant scale factor.

Otherwise, if the camera motion also involves translation, i.e. $\mathbf{t} \neq 0$, and the feature is not at infinity, the relationship between $\mathbf{p}_1$ and $\mathbf{p}_2$ is:

$$\begin{aligned} \mathbf{p}_2^t &\sim \mathbf{K}(\mathbf{R}_1^2 Z_1 \mathbf{K}^{-1}\mathbf{p}_1 + \mathbf{t}_1^2) \\ &\sim \mathbf{p}_2^r + \frac{1}{Z_1}\mathbf{K}\mathbf{t}_1^2, \end{aligned} \tag{4}$$

where $\mathbf{t}_1^2 = -\mathbf{R}^\top\mathbf{t}$, and Z_1 is the feature depth (Z-coordinate) in the camera's coordinate frame.

Using (3) and (4) to solve for the constraint in (1), we have the following result:

$$Z_1 \geq t_z + \sqrt{[f_x t_x + t_z(o_x - x)]^2 + [f_y t_y + t_z(o_y - y)]^2} \tag{5}$$

where $\mathbf{t} = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^\top$, and (x, y) is the non-homogeneous coordinate of $\mathbf{p}_1$.

This formula shows that the minimum depth $Z_{1min} \leq Z_1$ of a vista in the first camera view depends on the feature projection (x, y) in the first image, and also the camera translation $\mathbf{t}$. Figure 3 shows the Z_{1min} required for each pixel landmark location in the image where the camera moves forward in (1) z direction, and (2) both z and x directions.

Note that at the Focus Of Expansion (FOE) point, where the camera translation vector intersects with the camera image plane, we have $\mathbf{p}_2^t \sim \mathbf{p}_2^r \sim \mathbf{K}\mathbf{t}_1^2$, and the minimum Z_{1min} determined from (5) is very close to the camera. In this case, any features along the camera translation vector can be used to aid the rotation estimate reliably. As a trivial example, when the camera moves forward without rotation, $R = I_{3\times3}$, the minimum distance for rotation-aiding condition is $Z_{1min} = t_z$; hence, the image of any point along the camera optical axis is not affected by the camera translation, regardless of how far the camera moves, and how close the landmark is to the camera center as long as it is in front of the second camera view.

B. Computing Landmark Depth

We can easily use condition (5) to determine if a landmark is useful for rotation aiding, assuming we can estimate its depth Z_1 in the camera frame where the landmark is first observed. However, in the monocular case, this value is difficult to estimate and can only be estimated up to a scale factor without other sources of information about the camera motion.

We present in this section a simple method to quickly determine the landmark depth, based on the relationship between sizes of the landmark in camera images and the vehicle translation as estimated by using either the accelerometers or the downward-facing camera on the MAV. Our key observation is that the change in size of the landmark's image in the camera frames is related to its distance from the camera and the speed of the robot. For example, if a landmark is far away from the robot, the size of its projection will not change much while the robot is flying toward it. On the contrary, if the landmark is close to the robot, its size in the camera will change significantly.

We use a well-known property in perceptual psychology under the τ-theory [20], stating that the ratio τ of the object's image size to the rate of its size change determines the time-to-collision (TTC) to the object. In computer vision, this property has also been utilized in many work to compute TTC using optical flow [3], [12] or direct methods [18], [26]. We, on the other hand, apply this property directly on scale-space features such as SIFT [22] and SURF [6], of which the sizes can be optimally determined by 3D-quadratic fitting of the scale-space feature responses around the maximum response at the feature location.

Assuming the robot is moving forward, let s_1, s_2 be the sizes of the features associated with a landmark, and Z_1, Z_2 be the distance to the landmark from the camera at frame 1 and 2 respectively. We further assume that the landmark is far enough so that its surface could be approximated as a plane orthogonal to the camera view axis. Then we have $s_i = f\frac{S}{Z_i}$, where S is the real size of the landmark in 3D and f is the camera focal length. Consequently,

$$\frac{s_1}{s_2} = \frac{Z_2}{Z_1}. \tag{6}$$

Let $\Delta s = s_2 - s_1$ be the change in size of the features and $\Delta Z = Z_2 - Z_1 = -t_z$ be the amount of forward movement of the robot between two consecutive frames. It can be easily derived from (6) that

$$Z_1 = -\frac{s_2}{\Delta s}\Delta Z = \frac{s_2}{\Delta s} t_z. \tag{7}$$

Using (7), the landmark depth in the first camera frame is now determined from its feature sizes s_1, s_2 in two images and the camera forward translation t_z. Hence, knowing the camera velocity or the amount of translation between two consecutive frames, we then track features frame-to-frame, compute the depth of their associated landmarks using (7), and use the condition (5) to reject or accept vistas. We then choose the vista closest to the mean of all the accepted ones to aid the MAV rotation estimates as described in section III-C. As shown in Figure 1, this method is simple yet efficient to detect vistas in the environment.

III. Attitude Heading Reference System

The AHRS estimates the current attitude and heading by continuously integrating gyroscope measurements, and corrects for bias errors over time by making use of aiding measurements with the help of an error-state Kalman Filter. The system state is given by $\vec{\mathbf{x}} = \begin{bmatrix} \mathbf{R}_n^b & b_g & b_a \end{bmatrix}^\top$, where $\mathbf{R}_n^b$ is the rotation of the navigation frame with respect to the body frame, and b_g and b_a denote the time varying measurement biases of the gyroscope and accelerometer respectively. The system equations given here largely follow those provided in [14].

A. Initialization

The AHRS is initialized under the assumption that the vehicle starts out stationary, and the accelerometer can therefore be used to estimate the initial attitude. Let the mean accelerometer measurement in the *body* frame during the stationary initialization period be denoted by $\bar{\mathbf{a}}^\mathbf{b} = [\bar{a}_1, \bar{a}_2, \bar{a}_3]^\top$, and the initial roll ϕ, pitch θ, and yaw ψ in radians are given by

$$h(\mathbf{a}) = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} atan2(\bar{a}_2, \bar{a}_3) \\ atan2\left(-\bar{a}_2, \sqrt{\bar{a}_2^2, +\bar{a}_3^2}\right) \\ 0 \end{bmatrix} \tag{8}$$

The accelerometer gives us no information about the initial yaw ψ, and we therefore assume $\psi = 0$. Our state actually maintains $\mathbf{R}_n^b$, so we compute a rotation matrix given the result of (8), which gives us $\mathbf{R}_b^n$,and then store its inverse.

Given $h(\mathbf{a})$ above the Jacobian with respect to accelerometer measurements is

$$\mathbf{H}_a = \frac{\partial \mathbf{h}}{\partial \mathbf{a}} = \begin{bmatrix} 0 & \frac{a_3}{c} & \frac{-a_3}{c} \\ \frac{\sqrt{c}}{a_1^2+c} & -\frac{a_1 a_2}{\sqrt{c}(a_1^2+c)} & -\frac{a_1 a_3}{\sqrt{c}(a_1^2+c)} \\ 0 & 0 & 0 \end{bmatrix} \tag{9}$$

where $c = a_2^2 + a_3^2$ for brevity. The initial covariance matrix for the error-state filter is given by

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{11} & \mathbf{0} & \mathbf{P}_{12} \\ \mathbf{0} & \mathbf{P}_g & \mathbf{0} \\ \mathbf{P}_{12}^\top & \mathbf{0} & \mathbf{P}_{aa} \end{bmatrix} \tag{10}$$

where $\mathbf{P}_a$ and $\mathbf{P}_g$ are the bias covariances obtained from the stationary accelerometer and gyroscope measurements, respectively, and

$$\mathbf{P}_{11} = \mathbf{\Omega}_T \mathbf{H}_a (\mathbf{P}_a + \mathbf{P}_f) \mathbf{H}_g^\top \mathbf{\Omega}_T^\top \tag{11}$$

$$\mathbf{P}_{12} = -\mathbf{\Omega}_T \mathbf{H}_a \mathbf{P}_a \tag{12}$$

$$\mathbf{\Omega}_T = \begin{bmatrix} \cos(\psi)\cos(\theta) & -\sin(\psi) & 0 \\ \sin(\psi)\cos(\theta) & \cos(\psi) & 0 \\ -\sin(\theta) & 0 & 1 \end{bmatrix} \tag{13}$$

B. Mechanization

The gyroscope measures angular velocity ω in rads/sec in the inertial frame. The current bias estimate b_g is subtracted from this measurement and we then compute an incremental rotation using the Rodrigues formula

$$\Delta \mathbf{R}_n^n = rodrigues\left(\mathbf{R}_{n_t}^{b^\top} - (\omega - \mathbf{b}_g)\right) \tag{14}$$

which is then composed with the previous rotation state, all in the navigation frame:

$$\mathbf{R}_{n_{t+1}}^b = \mathbf{R}_{n_t}^b \Delta \mathbf{R}_n^n \tag{15}$$

The state transition matrix is

$$F = \begin{bmatrix} -\mathbf{R}_n^b & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I}/\tau_g & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{I}/\tau_a \end{bmatrix} \tag{16}$$

where τ_g and τ_a are empirically determined gyroscope and accelerometer bias time constants, respectively.

C. Vista Aiding

Aiding with vistas is slightly more involved than the case of aiding with an accelerometer or magnetometer. For example, in the case of the accelerometer the assumption is made that gravity always points straight down in the navigation frame. Given the current estimate of the body rotation, this vector is easily transformed into the body frame, such that

$$\mathbf{a}^b = \mathbf{R}_n^b \mathbf{a}^n, \tag{17}$$

and the measurement residual $\mathbf{r}$ is then obtained by subtracting the current accelerometer measurement so we obtain

$$\mathbf{r} = \mathbf{R}_n^b \mathbf{a}^n - \mathbf{a}^b \tag{18}$$

Vista aiding, however, is not performed with respect to an absolute reference direction that can be treated as fixed for flight duration. As the MAV moves through the world it will observe different vistas, and these will project to different (x, y) coordinates in the image. Furthermore, the camera frame-rate is much lower than the IMU measurement rate. When a rotation-aiding landmark is successfully tracked between two consecutive image frames I_k and I_{k+1} the position of the landmark in the current frame is predicted based on the incremental rotation that has taken place between these two frames as obtained from gyroscope measurement

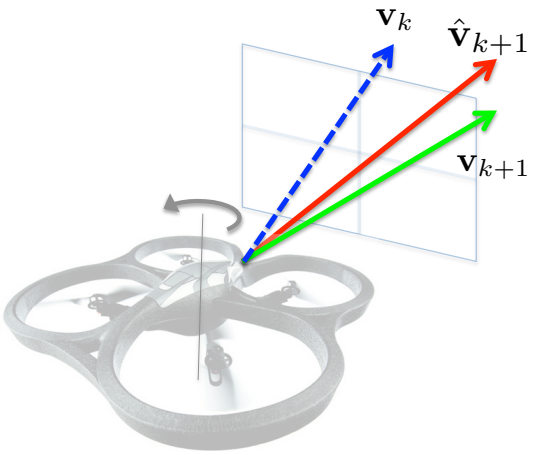


Figure 4: Vista unit vectors as observed by the MAV. $\mathbf{v}_k$ represents the observation in the current image frame, $\hat{\mathbf{v}}_{k+1}$ is the predicted landmark location in the next frame, which will have moved to the right in the image assuming a counterclockwise MAV rotation, and $\mathbf{v}_{k+1}$ is the actual observation in the next frame.

Figure 5: Vista tracking from frame to frame. The tracked vista location from the previous frame is shown in blue, the current location in green, and the red cross denotes the predicted vista location based on the incremental body rotation since the last image frame.

mechanization. The incremental rotation that has been incurred between two image frames is computed using

$$\Delta\mathbf{R} = \mathbf{R}^b_{n_{k+1}} \mathbf{R}^{b^\top}_{n_k} \tag{19}$$

Pixel coordinates (x, y) from landmark image observations are converted to unit vectors for ease of rotation and residual computation. Since a unit vector is sought, an arbitrary positive scene depth Z is chosen for the back-projection, and world 3D coordinates X and Y are computed using the pinhole camera model:

$$X = (x - o_x)\, Z/f_x \tag{20}$$

$$Y = (y - o_y)\, Z/f_y \tag{21}$$

The 3D coordinates are then collected into a vector representing the location (bearing) of the vista in the camera coordinate frame

$$\mathbf{P} = [\ X \quad Y \quad Z\ ]^\top \tag{22}$$

and the unit vector representing the direction of the vista from the camera center in the body frame is then given by

$$\mathbf{v} = \mathbf{R}^\mathbf{b}_\mathbf{c} \frac{\mathbf{P}}{\|\mathbf{P}\|} \tag{23}$$

where $\mathbf{R}^b_c$ accounts for the rotation from the camera frame to the body frame. Given a vista observation vector in frame k (shown in blue in Figs. 4,5), the predicted bearing vector in frame $k+1$ (shown in red in Figs. 4,5) is then obtained as

$$\hat{\mathbf{v}}_{k+1} = \mathbf{\Delta R} \mathbf{v}_k \tag{24}$$

The aiding residual is then obtained by subtracting the actual measurement in frame $k+1$ (shown in green in Figs. 4,5)

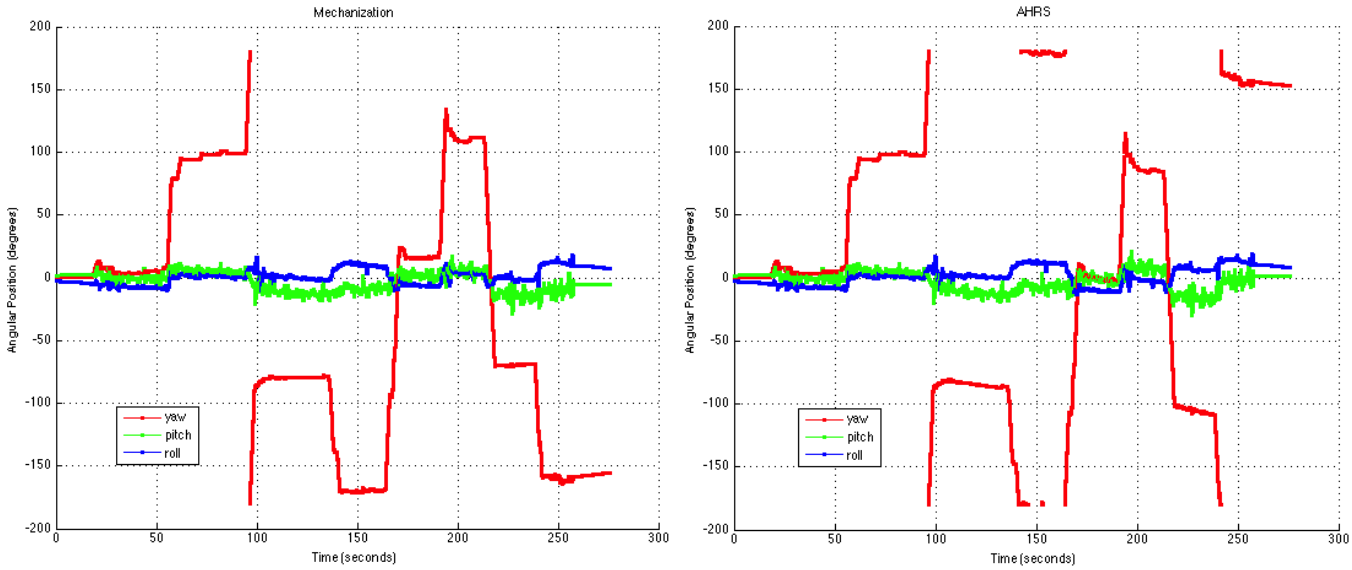
$$\mathbf{r} = \hat{\mathbf{v}}_{k+1} - \mathbf{v}_{k+1} \tag{25}$$

Figure 4 illustrates the vista prediction and residual calculation described in equations (19)-(25). The equations above assume that the center of rotation of the MAV coincides with the camera's center of projection. We have the requirement that rotation-aiding landmarks will always be far from the MAV, and thus this offset can be safely neglected. Figure 5 shows an example of vistas being tracked between different frames.

IV. Experiments

To evaluate our aided AHRS system we collected data using an AR.Drone quadrotor which is commercially available and low-cost [1]. The quadrotor used in our experiments is shown in Figure 2. The AR.Drone has downward- and front-facing cameras, as well as gyroscopes and accelerometers. A detailed explanation of the quadrotor can be found in [9].The on-board computer uses a filter to estimate the current pose using the downward facing camera, but heading drift can be observed over time nonetheless. We flew the quadrotor around an L-shaped hallway that is about 20×30 meters, and aim to arrive at the starting location with the same heading as we started. In Fig. 6a the raw mechanization can be seen, which exhibits considerable accumulated drift around 170-190 seconds. This is corrected due to vista aiding in the AHRS result shown in Fig. 6. The gyroscope biases that are estimated and corrected due to vista-aiding are shown in Fig. 7.

We conducted another experiment where we disabled the quadrotor's on-board filter and the AR.Drone was kept stationary. We observed much larger drift in the gyroscope readings than in the previous experiment. To highlight the impact of vista-aiding, we only show yaw in the following figures. The mechanization without aiding is shown in Fig. 8a, and the AHRS result with aiding is shown in Fig. 8b. It



(a) Open loop mechanization without aiding.

(b) With aiding the loop around the square hallway is completed at 0 degrees, as can be seen around 170-190 seconds.

Figure 6: AHRS result before and after vista aiding for a square flight sequence. The MAV flies around in an L-shape in a square hallway, making sharp 90 degree turns at each corner.

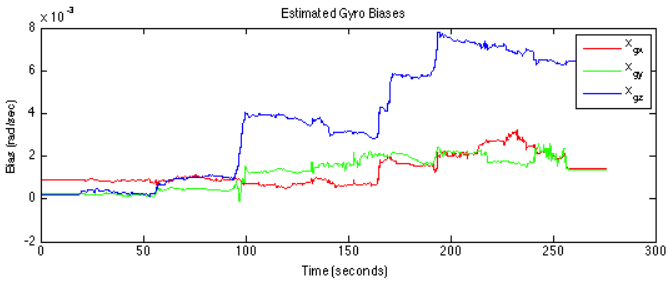


Figure 7: Estimated gyroscope biases. Significant corrections are made during turns when errors are highly observable.

is important to remember that the image frame-rate is much lower than the IMU measurement rate (about 10Hz vs. 190Hz in this experiment), and vista aiding therefore occurs quite infrequently, comparatively speaking.

V. Conclusion

In this paper we have presented a novel method for vision-aided inertial navigation in the context of an attitude heading reference system. We have introduced lightweight aiding landmarks which we call vistas, and have demonstrated their ability to correct for IMU drift on a flying quadrotor. Vistas are ideal for aiding in a variety of IMU equipped platforms, as they can be used without having to estimate the actual 3D location or even having to build up an entire map of the environment in which to localize.

References

[1] http://ardrone.parrot.com/, March 2012.

[2] S. Ahrens, D. Levine, G. Andrews, and J.P. How. Vision-based guidance and control of a hovering vehicle in unknown, gps-denied environments. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 2643–2648. IEEE, 2009.

[3] N. Ancona and T. Poggio. Optical flow from 1-d correlation: Application to a simple time-to-crash detector. *International Journal of Computer Vision*, 14(2):131–146, 1995.

[4] M. Antone and S. Teller. Automatic recovery of relative camera rotations in urban scenes. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 282–289, Hilton Head, June 2000.

[5] L. Armesto, J. Tornero, and M. Vincze. Fast ego-motion estimation with multi-rate fusion of inertial and vision. *The International Journal of Robotics Research*, 26(6):577–589, 2007.

[6] H. Bay, T. Tuytelaars, and L. V. Gool. Surf: speeded up robust features. In *Eur. Conf. on Computer Vision (ECCV)*, 2006.

[7] C. Bills, J. Chen, and A. Saxena. Autonomous MAV flight in indoor environments using single image perspective cues. In *Robotics and Automation, 2011. ICRA 2011. Proceedings of the 2011 IEEE International Conference on*, 2011.

[8] M. Blosch, S. Weiss, D. Scaramuzza, and R. Siegwart. Vision based mav navigation in unknown and unstructured environments. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 21–28. IEEE, 2010.

[9] P.J. Bristeau, F. Callou, D. Vissière, and N. Petit. The navigation and control technology inside the ar. drone micro uav. In *World Congress*, volume 18, pages 1477–1484, 2011.

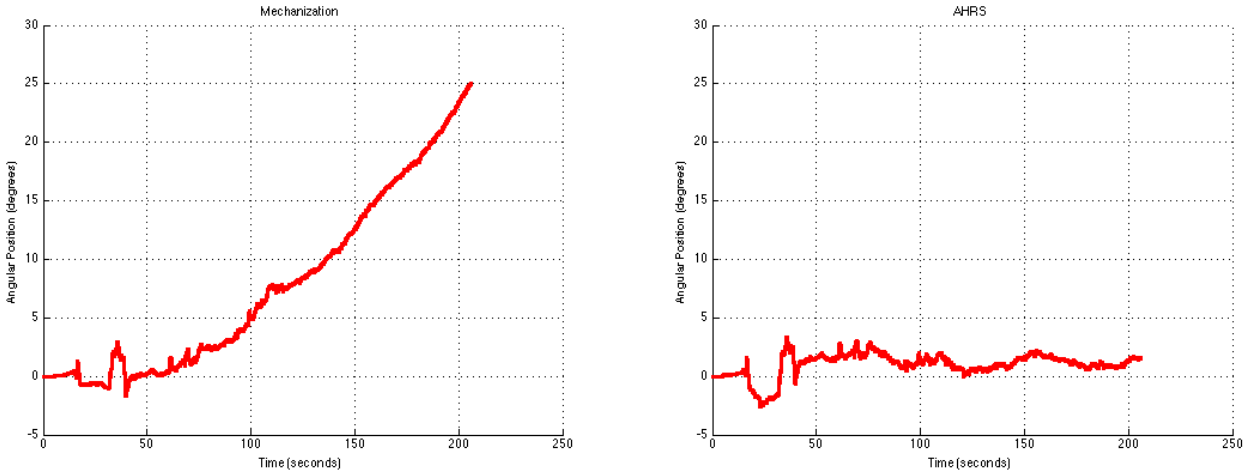
[10] M. Bryson and S. Sukkarieh. Building a robust implementation of bearing-only inertial slam for a uav. *Journal of Field Robotics*, 24(1-2):113–143, 2007.

[11] G. Conte and P. Doherty. An integrated uav navigation system based on aerial image matching. In *Aerospace Conference, 2008 IEEE*, pages 1–10. Ieee, 2008.

[12] D. Coombs, M. Herman, T. Hong, and M. Nashman. Real-time obstacle avoidance using central flow divergence and peripheral flow. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 276–283. IEEE, 1995.

[13] Jorge Dias, Markus Vinzce, Peter Corke, and Jorge Lobo. Editorial: Special issue: 2nd workshop on integration of vision and inertial sensors. *I. J. Robotic Res.*, 26(6):515–517, 2007.

[14] J.A. Farrell. *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill, 2008.



(a) Open loop mechanization (yaw only) without vista aiding.

(b) AHRS result with vista aiding (yaw only).

Figure 8: AHRS result for a stationary AR.Drone flight sequence.

[15] P. Gemeiner, P. Einramhof, and M. Vincze. Simultaneous motion and structure estimation by fusion of inertial and vision data. *The International Journal of Robotics Research*, 26(6):591–605, 2007.

[16] M. George and S. Sukkarieh. Inertial navigation aided by monocular camera observations of unknown features. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 3558–3564. IEEE, 2007.

[17] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[18] B.K.P. Horn, Y. Fang, and I. Masaki. Time to contact relative to a planar surface. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 68–74. IEEE, 2007.

[19] E.S. Jones and S. Soatto. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *Intl. J. of Robotics Research*, 30(4), Apr 2011.

[20] D.N. Lee et al. A theory of visual control of braking based on information about time-to-collision. *Perception*, 5(4):437–459, 1976.

[21] T. Lemaire, S. Lacroix, and J. Sola. A practical 3d bearing-only slam algorithm. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 2449–2454. IEEE, 2005.

[22] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *Intl. J. of Computer Vision*, 60(2):91–110, 2004.

[23] A. Martinelli. Vision and imu data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination. *Robotics, IEEE Transactions on*, (99):1–17, 2012.

[24] A.I. Mourikis and S.I. Roumeliotis. A multi-state constraint Kalman filter for vision-aided inertial navigation. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 3565–3572, April 2007.

[25] A.I. Mourikis and S.I. Roumeliotis. A dual-layer estimator architecture for long-term localization. In *Proc. of the Workshop on Visual Localization for Mobile Platforms at CVPR*, Anchorage, Alaska, June 2008.

[26] V.N. Murali and S.T. Birchfield. Autonomous exploration using rapid perception of low-resolution image information. *Autonomous Robots*, pages 1–14, 2012.

[27] S.I. Roumeliotis, A.E. Johnson, and J.F. Montgomery. Augmenting inertial navigation with image-based motion estimation. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 4, pages 4326–4333. IEEE, 2002.

[28] A.D. Wu, E.N. Johnson, and A.A. Proctor. Vision-aided inertial navigation for flight control. *Journal of Aerospace Computing, Information, and Communication*, 2, 2005.