AIAA Infotech@Aerospace Conference
and
AIAA Unmanned...Unlimited Conference
6 - 9 April 2009, Seattle, Washington

AIAA 2009-1962

Biologically Inspired Monocular Vision Based Navigation and Mapping in GPS-Denied Environments

Koray Celik*, Soon-Jo Chung†, Matthew Clausman‡, and Arun K. Somani§
Iowa State University, Ames, Iowa, 50011, USA

This paper presents an in-depth theoretical study of bio-vision inspired feature extraction and depth perception method integrated with vision-based simultaneous localization and mapping (SLAM). We incorporate the key functions of developed visual cortex in several advanced species, including humans, for depth perception and pattern recognition. Our navigation strategy assumes GPS-denied manmade environment consisting of orthogonal walls, corridors and doors. By exploiting the architectural features of the indoors, we introduce a method for gathering useful landmarks from a monocular camera for SLAM use, with absolute range information without using active ranging sensors. Experimental results show that the system is only limited by the capabilities of the camera and the availability of good corners. The proposed methods are experimentally validated by our self-contained MAV inside a conventional building.

I. Introduction

Undoubtedly the most influential perceptual sensory mechanism in biology is vision. Contrary to popular belief, even echolocating bats rely on vision for ranging beyond the attenuation of their sonar, with the visual acuity to discriminate a coin from several meters.[1] Perhaps the most interesting aspect of vision is the ability to estimate the range to an object without emitting any wave signal that can be detected by the prey; the ultimate eavesdropping tool. These properties render vision a particularly useful method for situation awareness in predators, and, an intuitive for SLAM in probabilistic robotics. A vision guided platform also has a wide potential for military applications held at GPS denied environments.

Since the photoreceptor cells in retina capture the surrounding geometry through photometric effects, the work of merit in visual ranging belongs to the hyper-complex neurons in visual cortex. The intricate details as to how these neurons function is still a mystery. However, studies such as Hubel et al.[2] shed light on how visual cortex might operate, in which the extracellularly responses from the dorsal aspect of cat brain was studied using the actions of the animal as a probe to estimate the functions of these neurons in response to visual stimuli. It was discovered that the visual cortex prefers tracking small contours of the environment, and the optokinetic nystagmus of the animal suggested that moving contours were of particular interest.

Our approach takes these contours into account, inspired by animals with two dimensional retinae that perceive depth via such monocular visual cues such as line perspectives, relative height, texture gradient, and motion parallax. We would like to stress the term *monocular* here; studies on cats have shown that 80% of all cells in visual cortex were influenced independently by the two eyes, suggesting that when it comes to long-range navigation, monocular vision is more influential than stereo-vision in biology. Eagles for instance, utilize the two eyes independently to track multiple landmarks (and targets) simultaneously, and estimate depths in a monocular manner. Using both eyes in unison is only useful for objects within immediate vicinity. All other times they stare in parallel to obtain a single wide-angle image. Similarly in probabilistic robotics, stereo vision does not have the potential for real-time online SLAM applications with

*Doctoral Research Assistant, Department of Electrical and Computer Engineering, koray@iastate.edu.
†Assistant Professor of Aerospace Engineering and Electrical & Computer Engineering, sjchung@alum.mit.edu.
‡Department of Electrical and Computer Engineering, mclausma@iastate.edu
§Anson Marston Distinguished Professor, Jerry R. Junkins Endowed Chair, and Department Chair, Department of Electrical and Computer Engineering, arun@iastate.edu.

reasonable computation as well as robustness and a wide range of depths. We use a monocular camera as the only range measurement sensor for SLAM, which delivers the best information-to-weight ratio, and we solve the depth problem by exploiting moving contours.

A. Other Methods of Remote Range Measurement in Literature

Binocular cameras receive particular attention as they can shift two simultaneous images over each other to find the parts with the best match and the disparity at which objects in the image best match is used to calculate their distance. Nevertheless, stereo vision has a significant limitation in its useful range, particularly when a large region of the image contains homogeneous textures. Moreover, most mammals feature adaptive binocular vision; viewing angle of the eyes with respect to each other change according to the distance to the observed object in order to detect different ranges. Such a system would introduce a significant mechanical complexity that the current stereo-vision cameras do not permit. Parabolic and panoramic cameras are often used in robotics for their extremely wide field of view[13] however they are better suited for mobile ground robots with wheel odometry due to size and complexity.[24]

The literature has also resorted to distance measurement via attaching photo lenses to optical flow sensors.[16,17] Similar to insect vision, this light-weight sensor contains a low resolution (usually 18×18 pixels) CMOS chip which outputs two values representing the total optical flow. If the properties of the lens are known, it is possible to exploit the parallax effect for distance measurement. However, the device requires incessant motion and hence becomes useless in a hovering MAV. Assumptions made about the surface shape and the orientation pose a significant limitation, since the sensor cannot determine correct orientation of the surface independently. Also, an 18×18 image patch is too ambiguous for landmark association procedure, an essential step for a vision based SLAM.

Vision research has particularly concentrated on reconstruction problems from small image sets, known as Structure from Motion. It is based on analysis of a complete sequence of images to produce a reconstruction of the camera trajectory and scene structure. This approach may be suitable for solving the offline-SLAM problem; automatic analysis of recorded footage from a completed mission cannot scale to consistent localization over arbitrarily long sequences in real-time.

Using moving lenses for depth extraction has also been studied.[9] Objects inside the field-of-view of a camera lens, but not beyond the infinity point, can be made to appear out of focus as the lens focal length is varied. The resulting Gaussian blur selectively destroys the discrete tonal features and their spatial relationships, and the remaining area where the camera has the sharpest focus can be considered for distance measurement if the lens properties are known. Nonetheless, the focus of interest may not necessarily be an useful feature to begin with and the method cannot focus at multiple depths at once. Large orthogonal objects return a 2D multi-modal probability distribution for the location of the measured depth, therefore the method alone is not reliable enough for SLAM. Moreover, unless the lenses can be moved at a very high frequency, the approach will signi?cantly reduce the sensor bandwidth.

The most popular sensors in the SLAM research community have been laser range finders and sonar. However, the simplicity of direct depth measurements comes at the expense of the sensor weight and measurement ambiguity. More importantly, the measurements are 2D by nature; a complicated mechanical gimbal is required to perform a 3D scan,[8] which can only be supported by a sufficiently large robotic platform, naturally clumsy in an indoor environment. Three dimensional SLAM methods use a swivel mechanism that nods a 2D laser range finder 90 degrees on a gimbal along its horizontal. This fairly common technique was also used in the DARPA Grand Challenge, the laser range finder of preference being the SICK LMS-200, which weighs nearly 10 pounds. In theory, it is desirable to nod rapidly for an appropriate 3 dimensional ranging. However, the power consumption in nodding such a mass at high frequency is overwhelming for most robots. When the nodding is performed slowly, the gimbal mechanism becomes the bottleneck for observation bandwidth. The landmarks sets are analyzed for clusters that are distinguishable to serve as high level features for navigation using principal component analysis and 2D least squares fitting of a plane inside a point cloud. Landmark extraction in part is based on corners. A corner is the intersection of three orthogonal planes. With the way the laser range finder is installed and rotated on the robot, it will be difficult to detect corners as most will fall into the blind spots of the device.

B. Related Work

Efforts to retrieve depth information from a still image by using machine learning such as the Markov Random Field (MRF) learning algorithm[10,18] have no notable limitation that a-prior information about the environment must be obtained from a training set of images. These requirements disqualify it as a candidate for an online SLAM algorithm in an unknown environment. Structure From Motion approaches based on automatic analysis of recorded footage from a completed mission are only suitable for solving only the offline-SLAM problem. Extended Kalman Filter (EKF) based approaches to probabilistic vision based SLAM such as MonoSLAM[6] are excellent for applications requiring precise and repeatable localization within immediate vicinity of the starting point, but not well-suited for vision guided navigation of an MAV that covers a large unknown area. A full-covariance EKF SLAM is a quadratic time algorithm with respect to number of landmarks, which severely limits the size of a manageable map in realtime, restricting the method to room sized domains. Depth information is estimated from sideways movement of the camera, however an MAV helicopter through a corridor is restricted by walls, it has to make best use of motion along the optic axis of the camera. Finally, the need to know the approximate starting location of the camera is a significant limitation for an MAV mission that can start at any arbitrary location in an unknown environment. Global localization techniques like CondensationSLAM require the full map to be provided a-priori. Azimuth based techniques such as CognitiveSLAM are highly parametric, and locations are centered on the robot which will not work with ambiguous landmarks. Image registration based methods, such as,[20] propose a different formulation of the vision-based SLAM problem based on motion, structure, and illumination parameters without first having to find feature correspondences. For real time implementation, a local optimization procedure is required, and there is a high chance of getting trapped in a local minimum. Further, without merging regions with a similar structure, the method becomes computationally unmanageable. The Structure extraction method[7] has its own limitations since an incorrect incorporation of points into higher level features will have an adverse effect on consistency. Higher level structures are purely constructed from the information contained in the map while there is an opportunity to combine the map with the camera readings. Further, these systems depends on a successful selection of thresholds which have a considerable impact on the system performance, thus limited to small scale maps.

Our method addresses these shortcomings using a 2-megapixel monocular camera that occupies 1×2 inches on the MAV and weighs less than 2 ounces. The rest of this paper is organized as follows. Section II explains the procedures for perception of world geometry as pre-requisites for SLAM, such as range measurement methods explained in sections C and D, a visual turn-sensing algorithm described in F, and performance evaluations of proposed methods in section E. SLAM formulations are provided in section III, including experimental validation in section B. Fig.1 can be used as a guide to sections as well as to the process flow of our method.

II. Ranging and SLAM Formulation

We propose a novel method to estimate the absolute depth of features using a monocular camera, which is the unaccompanied sensor in our experiments. The only a-priori information required is the MAV altitude above ground, and the only assumption made is that the landmarks are stationary.

A. Landmark Extraction

No SLAM approach is a dependable solution without reliable landmarks. A landmark in SLAM context is a conspicuous, distinguishing landscape feature marking a location. This definition is sufficient for SLAM, but not necessary. A minimal landmark can consist of two measurements with respect to robot position, range and bearing. To automate landmark extraction we begin extracting prominent parts of the image that are more attractive than other parts in terms of energy. We refer to these as features. For instance, a corner makes a nice feature. So does a fire alarm on a white wall. But the wall itself is uniform and thus unlikely to attract a feature scanner. It must be noted that landmarks exist in real 3D world and they are distinctive whereas features exist on the 2D image plane and they are ambiguous. We select and convert qualifying features into landmarks as appropriate.

In our preliminary results,[4,5] we have tried various methods, including an extension of the Harris corner detection algorithm starting with the idea that corners at the intersection of three orthogonal walls can lead to most consistent landmarks. However, due to its Markovian nature the algorithm was not well suited for

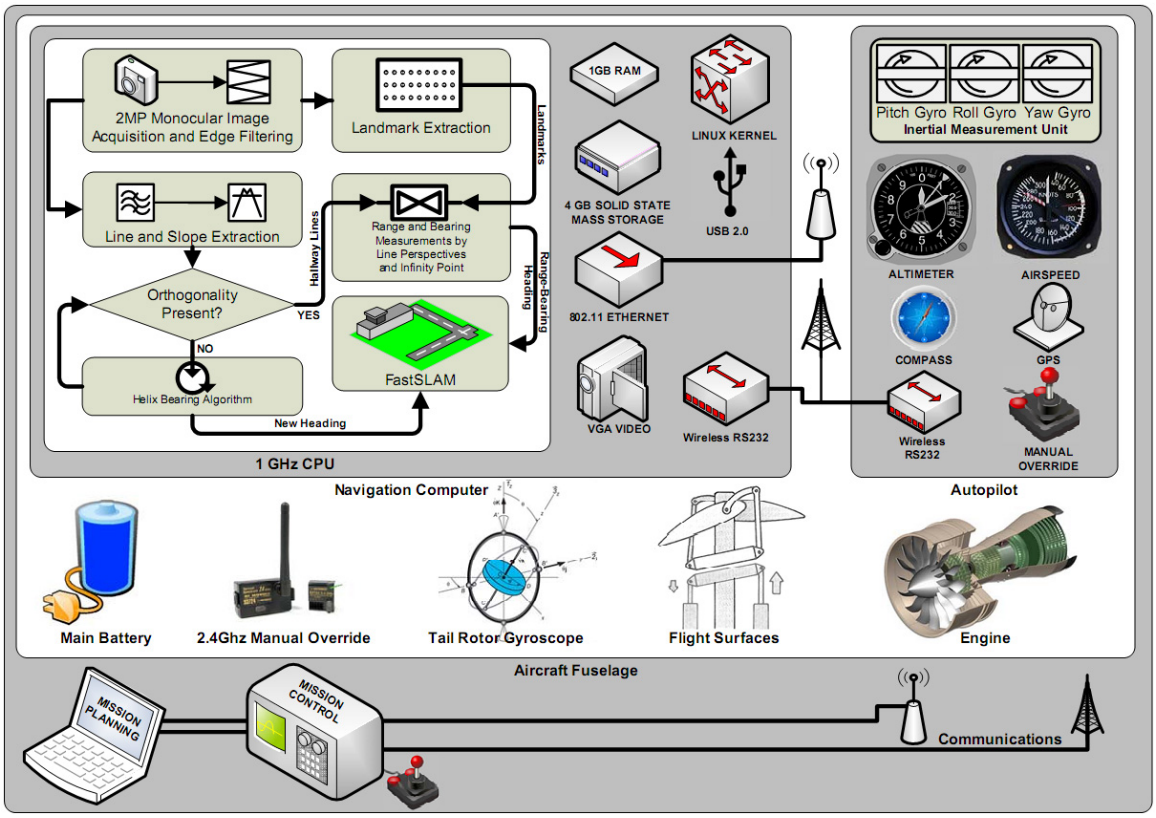

Figure 1. Block diagram illustrating the MAV systems and operational steps of the monocular vision navigation and ranging at high level.

tracking agile motion. Harris method is a feature detector, not effectively a feature tracker since every frame is considered independently and no history of features is kept. In slow image sequences this may provide a sparse and consistent set of corners due to its immunity to rotation, scale, illumination variation, and image noise.

We have obtained the best feature detection, and tracking performance from the continuous algorithm proposed by Shi and Tomasi which works by minimizing the dissimilarity between past images and the present image in a sequence. Features are chosen based on their monocular properties such as texture, dissimilarity, and convergence; sections of an image with large eigenvalues are considered "good" features; conceptually similar to the surface integration of the human vision system. The authors in[23] present a latest measure of feature "goodness", based on the Lucas-Kanade tracker performance. The method selects a large number of features based on the criteria set forth by Shi-Tomasi and then removes features with small convergence region. Although this improves the consistency of the earlier method, it is still probabilistic and therefore, it cannot make an any more educated distinction than Shi-Tomasi between an useless feature and a potential landmark. That distinction is later performed by our method, extracting a sparse set of reliable landmarks from a populated set of questionable features.

B. Line and Slope Extraction

For our range measurement algorithms to work as described in sections C and D, the architectural ground lines should be extracted. We use Hough Transform on edge filtered frames to detect lines with a finite slope $\phi \neq 0$ and curvature $\kappa = 0$. Detections are then sorted with assumption of orthogonality of the environment, and lines referring to the ground edges are extracted. Although these are virtually parallel in the real world, on the image plane they intersect and the horizontal coordinate of this intersection point is later used as a heading guide. And features that happen to coincide with these lines become landmark candidates.

The concept of *ground lines* in a hallway is a logical entity which is fuzzy in reality. Doors, reflections,

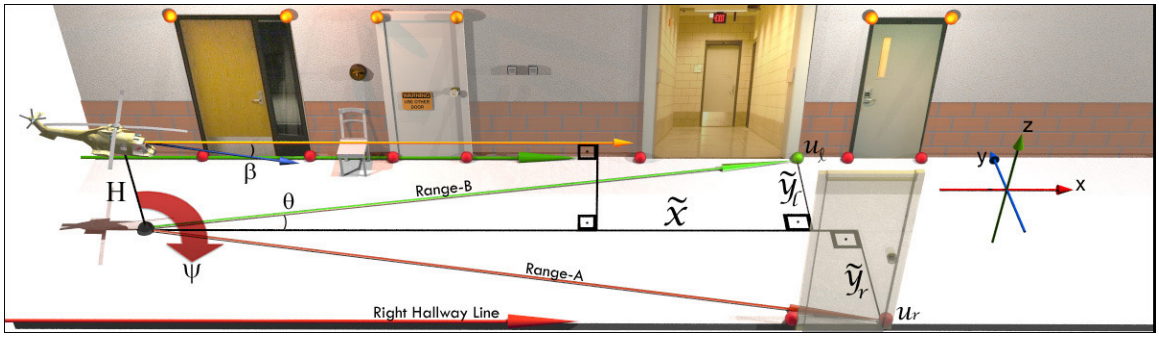

Figure 2. ***Left:*** **A three dimensional representation of the corridor showing line perspectives and corner-like features.**

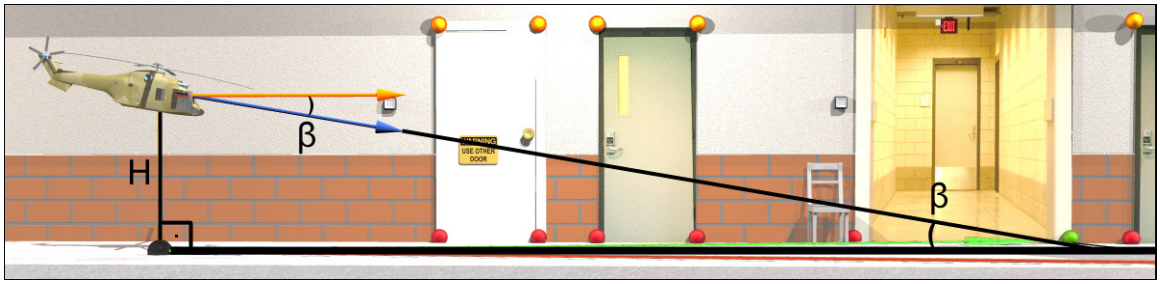

Figure 3. The image shows a conceptual cutaway of the corridor from the right. The angle β represents the angle at which the camera is pointing down.

and random introduction of stationary or moving objects continuously segment, sometimes even mimic these lines. Further, the far end of a hallway appears too small on the image plane and therefore is aliased creating what appears as noise, causing the corresponding ends of the hallway lines to translate randomly. The stochastic presence and absence of these perturbations result in lines that are inconsistent about their position, even when the MAV is at hover, causing noisy slope measurement and in turn noisy landmarks. Since our range measurement methods depend on these lines, their overall accuracy becomes a function of the robustness in detecting the hallway lines. The high measurement noise in slopes has adverse effects on SLAM and should be minimized to prevent inflating the uncertainty in $L_1 = \tan\phi_1$ and $L_2 = \tan\phi_2$ in (3) or the infinity point (P_x, P_y). To reduce this noise, lines obtained in the earlier step are cross-validated with collinear line segments obtained via pixel neighborhood based line extraction in which the results obtained rely only on a local analysis. Their coherence is further improved using a postprocessing step via exploiting the texture gradient.

C. Range Measurements by Infinity Point Method

Inspired by the recent Nature paper,[21] our monocular ranging algorithm mimics the human perception system, and accurately judges the absolute distance by integrating local patches of the ground information into a global surface reference frame. This new method, efficiently combined with the feature extraction method and SLAM algorithms, significantly differs from optical flows in that the depth measurement does not require a successive history of images.

Once features and both of the ground lines are detected, our range and bearing measurement strategy assumes that the height of the camera from the ground, H, is known a priori. This can be the altimeter reading of the MAV. The camera is pointed at the far end of the corridor, tilted down with an angle β. The incorporation of the downward tilt angle of the camera was inspired by the human perception system that perceives distances by a directional process of integrating ground information up to 20 meters,[21] indeed, humans cannot judge the absolute distance beyond 2 to 3 meters without these visual cues on ground. Fig. 2 describes the environment at this stage, note the two ground lines that define the ground plane of the

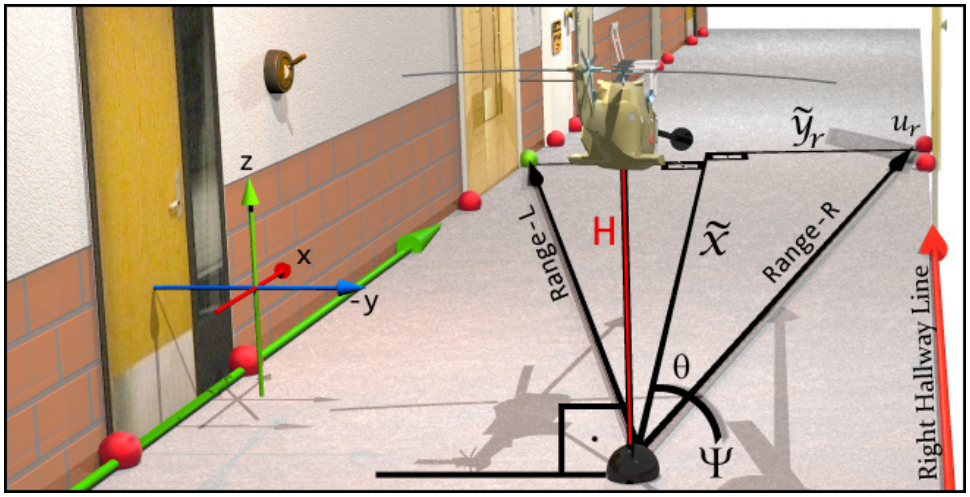

Figure 4. The corridor as seen by the MAV.

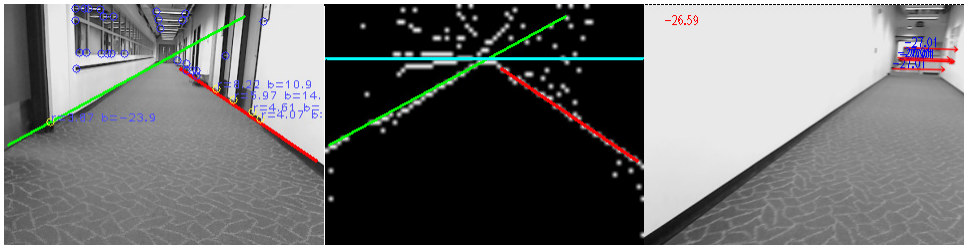

Figure 5. Screenshots of the line and turn detection algorithms.

corridor.

The concept of infinity point, (P_x, P_y) was added to obtain vehicle yaw angle and camera pitch angle. Infinity point is an imaginary concept where the projections of the two hallway lines happen to intersect on the image plane. Since this imaginary intersection point is infinitely far from the camera, it presents no parallax from translation of the camera. It does, however, effectively represent the yaw and pitch of the camera. Assume the end points of the hallway ground lines are $E_{H1} = [l, d, -H]$ and $E_{H2} = [l, d - w, -H]$ where l is length and w is the width of the hallway, d is the horizontal displacement of the camera from the left wall, and H is the MAV altitude. The Euler rotation matrix to convert from the camera frame to the hallway frame is given in (1),

$$A = \begin{vmatrix} c\psi c\beta & c\beta s\psi & -s\beta \\ c\psi s\phi s\beta - c\phi s\psi & c\phi c\psi + s\phi s\psi s\beta & c\beta s\phi \\ s\phi s\psi + c\phi c\psi s\beta & c\phi s\psi s\beta - c\psi s\phi & c\phi c\beta \end{vmatrix} \tag{1}$$

where c and s are abbreviations for cos and sin functions respectively. The vehicle yaw angle is denoted by ψ, camera pitch is denoted by β and vehicle roll is denoted by ϕ which is is controlled by the autopilot system to remain at zero.

The points E_{H1} and E_{H2} are transformed into the camera frame via multiplication with the transpose of the matrix in (1), $E_{C1} = A^T.[l, d, -H]$ and $E_{C2} = A^T.[l, d - w, -H]$. This 3D system is then transformed into the 2D image plane via $u = yf/x$ and $v = zf/x$, where u is the pixel horizontal position from center (right is positive), v is the pixel vertical position from center (up is positive), and f is the focal length. The end points of the hallway lines have now transformed from $E1_{Hall}$ and $E2_{Hall}$ to $[Px_1, Py_1]$ and $[Px_2, Py_2]$, respectively. An infinitely long hallway can be represented by $\lim_{l\to\infty} Px_1 = \lim_{l\to\infty} Px_2 = f \tan\psi$ and $\lim_{l\to\infty} Py_1 = \lim_{l\to\infty} Py_2 = -f \tan\beta/\cos\psi$, which is conceptually same as extending the hallway lines to

infinity. The fact that $Px_1 = Px_2$ and $Py_1 = Py_2$ literally means the intersection of the lines in the image plane is the end of such an infinitely long hallway. Solving the resulting equations for ψ and β yields the camera yaw and pitch respectively,

$$\psi = \tan^{-1}(P_x/f), \quad \beta = -\tan^{-1}(P_y \cos\psi/f)$$

A generic form of the transformation from pixel position, $[u, v]$ to $[x, y, z]$, can be derived in a similar fashion. The equations for u and v also provide general coordinates in the camera frame as $[z_c f/v, uz_c/v, z_c]$ where z_c is the z position of the object in the camera frame. Multiplying with (1) transforms the hallway frame coordinates $[x, y, z]$ into functions of u, v, and z_c. Solving the new z equation for z_c and substituting into the equations for x and y yields,

$$\begin{aligned} \tilde{x} &= ((a_{12}u + a_{13}v + a_{11}f)/(a_{32}u + a_{33}v + a_{31}f))z \\ \tilde{y} &= ((a_{22}u + a_{23}v + a_{21}f)/(a_{32}u + a_{33}v + a_{31}f))z \end{aligned} \tag{2}$$

where a_{ij} refers to the elements of the matrix in (1). Refer to Fig.2 for descriptions of $\tilde{x}$ and $\tilde{y}$.

For objects likely to be on the floor, the height of the camera above the ground is the z position of the object. Also, if the platform roll can be measured, or assumed negligible, then the combination of the infinity point with the height can be used to give the range to any object on the floor of the hallway. This same concept applies to objects which are likely to be on the same wall, or the ceiling. By exploiting the geometry of the corners present in the corridor, our method computes the absolute range and bearing of the features, effectively turning them into landmarks needed for the SLAM formulation.

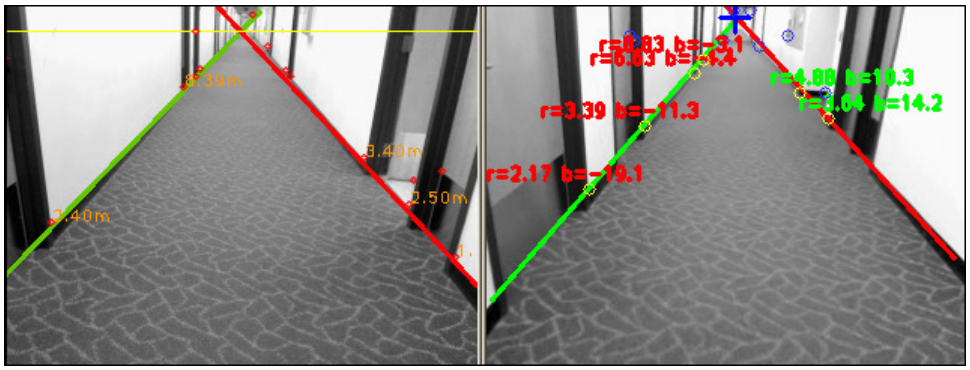

Figure 6. Screenshots of the range measurement algorithms in action. *Left:* Line-Perspectives method. *Right:* Infinity-Point method. Note the crosshair indicating the infinity point.

D. Range Measurements by Line Perspectives Method

Prior to the algorithm described in (C) our preliminary tests[4] employed an older method of range measurement and Infinity-Point method is an improvement over that. However, in the rare event when only one hallway line is detectable and the infinity point is lost, the system switches from Infinity-Point method to Line-Perspectives method until both lines are detected again. Line-Perspectives method applies successive rotational and translational transformations among the camera image frame, the camera frame, and the target corner frame to compute the slope angles for ground lines.

$$L_1 = \tan\phi_1 = H/(\tilde{y}_l \cos\beta), \quad L_2 = \tan\phi_2 = H/(\tilde{y}_r \cos\beta) \tag{3}$$

From (3), we can determine the left and right slopes, L_1 and L_2. If the left and right corners coincidentally have the same relative distance $\tilde{x}$ and the orientation of the vehicle is aligned with the corridor, $\tilde{y}_l + \tilde{y}_r$ gives the width of the corridor. Finally, we solve for the longitudinal distance $\tilde{x}$ and the transverse distance $\tilde{y}_l$, by combining the preceding equations:

$$\tilde{y}_l = u_e H/\alpha\sqrt{(1 - v_e/u_e/L_1)^2 + \alpha^2/{u_e}^2/L_1^h 2} \tag{4}$$

$$\cos\beta = H/\tilde{y}_l/L_1, \quad \tilde{x} = \left(\alpha\tilde{y}_l/u_e - \sin\beta H\right)/\cos\beta \tag{5}$$

where $\alpha = f/d$, $u_e = u_L - u_0$, $v_e = v_L - v_0$, $H = \alpha\tilde{y}_l/u_e \sin\beta + v_e/u_e\tilde{y}_l \cos\beta$. The process is recursive for all features visible and close to hallway lines.

E. Empirical Comparisons of Proposed Ranging Algorithms

The graph in Fig.7 illustrates the disagreement in between Line-Perspectives and Infinity-Point method (Sec.C). Both algorithms executed simultaneously on the same video feed. Line-Perspectives (Sec.D) has a calculated 89% confidence on the distance measurements whereas Infinity-Point method has a calculated 93% confidence. This suggests that disagreements not exceeding half a meter are in the favor of the new method. Transient measurement errors such as occasional introduction of deceptive objects which falsely mimic the shape of the environment, positions of walls, etc. result in small disagreements from the ground truth, otherwise in perfect hallways the algorithm would make perfect measurements. Divergence between the two ranges that is visible in between samples 20 and 40 in Fig.7 is caused by a hallway line anomaly from the line extraction process, independent of ranging. In that particular case both of the hallway lines shifted causing the infinity point to move left, visible by the bearing shift of both algorithms illustrated in Fig. (8). The bearing resolution of the camera used in our experiments is 0.2 degrees, and in this data set the bearing of both algorithms shifts by about a degree. The Fig.8 illustrates in the same manner, the disagreements for bearing to a feature. On average the two methods disagree less than 1 degree for bearing measurement. Note that a horizontal translation of the infinity point has minimal effect on measurement performance of the infinity-point method.

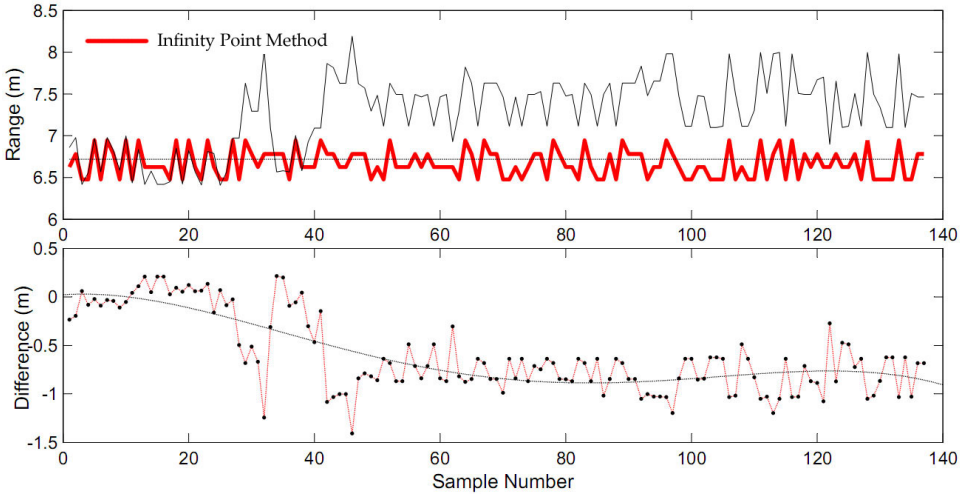

Figure 7. ***Top:*** **Illustrates the accuracy of the two range measurement methods with respect to ground truth (flat line).** ***Bottom:*** **Residuals for above graph.**

F. Challenges Associated with Turns

Humans, cats, and many other mammals can rotate their eyes against the head, and rotate their head against the body, feature inertial measurement units inside the ear canals to aid in measuring the rate the body is rotating. When the head is restrained to shoulders, inner ears are put to sleep with thiopental, and eye motion paralyzed with succinylcholine, the measurement becomes an estimation problem for the visual cortex. Since the interocular separation of eyes is rather small, stereo vision cannot be used effectively, the estimation must be performed in a monocular fashion. Srinivatsan[3] investigated how insects exploit the rich information resulting from the optic flow as they fly through a stationary environment, and use it to distinguish in between objects at different distances. Insects cannot rotate the head or eyes. Further, owing to the small interocular separation, most insects cannot rely on stereo-vision for this purpose, but rather perceive depths in terms of translational and rotational velocities of objects on the retina. By casual observation it is possible to see that when a bee flies through a window, it tends pass through the center.

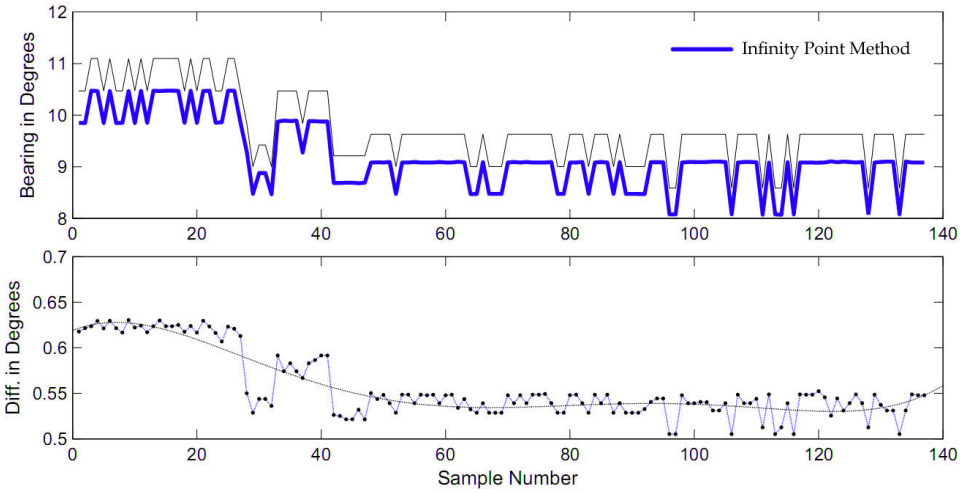

Figure 8. ***Top:*** **Illustrates the amount the two methods agree with each other with respect to bearing to a feature. The bias between the two measurements is due to camera calibration.** ***Bottom:*** **Residuals for above graph.**

Quite possibly the bee balances speeds of image motion on both eyes, assuming the window was stationary. These findings suggest that the visual cortex is capable of computing angular speed of a grating, independent of the spatial frequency.

When the MAV approaches a turn, an exit, a T-section, or an otherwise dead-end, a similar condition occurs. Both ground lines tend to disappear simultaneously. Consequently, both range measurement methods cease to function. Although a set of features might still be detected and the MAV can make a confident estimate of their spatial locality, their possible depths comprise an undetermined degree of freedom. For each feature, a set of discrete depth hypotheses is uniformly distributed, which can be thought of as a one-dimensional probability density over depth represented by a two-dimensional particle distribution. A bio-inspired turn-sensing algorithm to estimate ψ in the absence of orthogonality cues is automatically triggered, in which a yaw rotation of the body frame will be initiated until another passage is found. Estimating ψ with an accuracy that SLAM map will update correctly is a procedure that combines machine vision with the data matching and estimation problem. If the MAV approaches a left-turn after exploring one leg of an "L" shaped hallway for instance, and then turns left 90 degrees and continues through the next leg, the map is expected to display two hallways joined at a 90 degree angle. Similarly, a 180 degree turn before finding another hallway would indicate a dead-end. This way, the MAV can also determine where turns are located the next time they are visited. The SLAM procedure on turns would not be able to determine how far the MAV had to turn before finding a new hallway by itself, and the resulting map would not coincide with real world map.

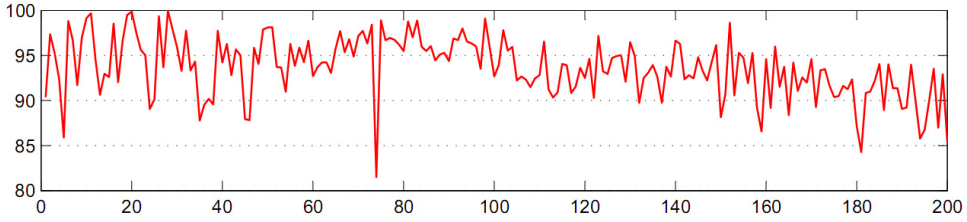

Figure 9. This graph illustrates the accuracy of the bearing estimation algorithm measuring 200 samples of laser-protractor calibrated 90 degree turns at varying locations. Angular rates were chosen randomly but not exceeding 1 radian-per-second to stay within the flight characteristics of the MAV and capabilities of the camera. The tests were performed in the absence of known objects.

Solving the estimation problem at turns begins with computing the instantaneous velocity, (u, v) of every helix (a feature with optic flow) that the MAV is able to detect as Fig.12 illustrates. Helix velocity is

recovered as $V(x,y,t) = (u(x,y,t),(v(x,y,t)) = (dx/dt, dy/dt)$ using a variation of the pyramidal Lucas-Kanade method. The result is a two dimensional vector field obtained via perspective projection of the 3D velocity field of a moving scene onto the image plane. At discrete time steps, a video frame is defined as a function of the previous video frame as $I_{t+1}(x,y,z,t) = I_t(x+dx, y+dy, z+dz, t+dt)$. By applying the Taylor series expansion,

$$I(x,y,z,t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial z}\delta z + \frac{\partial I}{\partial t}\delta t \tag{6}$$

then by differentiating with respect to time yields, the helix velocity is obtained in terms of pixel distance per time step k which advances at 30Hz in our current configuration. At this point, each helix is assumed to be identically distributed and independently positioned on the image plane, associated with a velocity vector $V_i = (v,\alpha)^T$ where α is the angular displacement of velocity direction from North of the image plane where $\pi/2$ is East, π is South and $3\pi/2$ is West. Although the associated depths of the helix set appearing at stochastic points on the image plane are unknown, assuming a constant $(\psi)\Delta t$, there is a relationship in between distance of a helix from the camera and its instantaneous velocity on the image plane. This suggests that a helix cluster with respect to closeness of individual instantaneous velocities is likely to belong on one planar object, such as a door frame. Let a helix with a directional velocity be the triple $h_i = \langle V_i, u_i, v_i\rangle$ where (u_i, v_i) represents the position of this particle on the image plane. At any given time (k), let Ψ be a set containing all these features on the image plane such that $\Psi(k) = \{h_1, h_2, \cdots, h_n\}$. The z component of velocity as obtained in (6) is the determining factor for α. Since we are most interested in the set of helix in which this component is minimized, $\Psi(k)$ is re-sampled such that,

$$\Psi'(k) = \{\forall h_i, \{\alpha \approx \pi/2\} \cup \{\alpha \approx 3\pi/2\}\} \tag{7}$$

sorted in increasing velocity order. $\Psi'(k)$ is then processed through histogram sorting to reveal the modal helix set such that,

$$\Psi''(k) = max\left|\{\sum_{i=0}^{n} i \text{ if } (h_i = h_{i+1}),\ 0 \text{ else}\}\right| \tag{8}$$

$\Psi''(k)$ is likely to contain clusters that tend to have a distribution which can be explained by spatial locality with respect to objects in the scene, whereas the rest of the initial helix set from $\Psi(k)$ may not fit this model. The RANSAC algorithm[22] is a useful method to estimate parameters of such models, however in the interest of efficiency, the MAV uses an agglomerative hierarchial tree, T, to identify the clusters. To construct the tree, $\Psi''(k)$ is heat mapped, represented as a symmetric matrix M, with respect to Manhattan distance in between each individual helix,

$$M = \begin{vmatrix} h_0 - h_0 & \cdots & h_0 - h_n \\ \vdots & \ddots & \vdots \\ h_n - h_0 & \cdots & h_n - h_n \end{vmatrix}$$

Tree construction and from M is as follows,

	Algorithm: Disjoint cluster identification from heat map M
1	Start from level $L(0) = 0$ and sequence $m = 0$
2	**Find** $d = min(h_a - h_b)$ in M where $h_a \neq h_b$
3	$m = m + 1$, $\Psi'''(k) = merge([h_a, h_b])$, $L(m) = d$
4	**Delete from** M**:** rows and columns corresponding to $\Psi'''(k)$
5	**Add to** M**:** a row and a column representing $\Psi'''(k)$
6	$if(\forall h_i \in \Psi'''(k))$, **stop**
7	*else*, go to 2

It is desirable to stop the algorithm before it completes since this would eventually result in $\Psi'''(k) = \Psi''(k)$. In other words, the tree should be cut at the sequence m such that $m+1$ does not provide significant benefit in terms of modeling the clusters. After this step, the set of velocities in $\Psi'''(k)$ represent the largest planar object in field-of-view with the most consistent rate of pixel displacement in time. At the lack of

absolute depth information, if no identifiable objects exist in the field-of-view, the system is updated such that $\Psi(k+1) = \Psi(k) + \mu(\Psi'''(k))$ as the best effort estimate as shown in Fig.9. However, if the MAV is able to identify a world object of known dimensions, $dim = (x, y)^T$ from its internal object database, such as a door, and the cluster $\Psi'''(k)$ sufficiently coincides with this object, Helix bearing algorithm can estimate depth to this cluster using $dim(f/dim')$ where dim is the actual object dimensions, f is the focal length and dim' represents object dimensions on image plane. Note that presence of known objects is not a requirement for the method to work, however they would increase its accuracy.

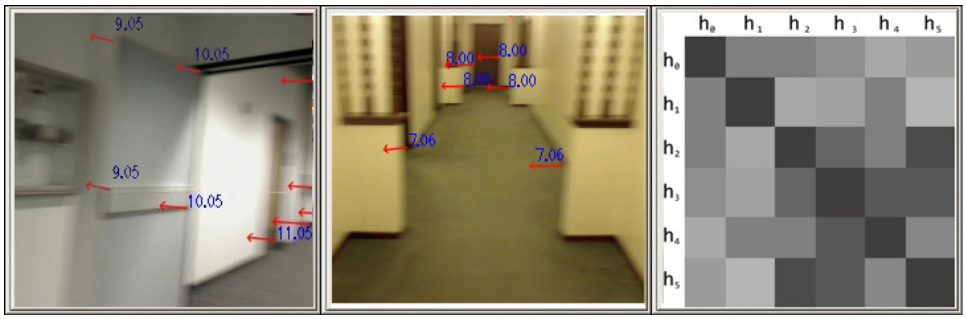

Figure 10. ***Left, Middle:*** **The figure describes the operation of bearing estimation algorithm. Arrows represents the instantaneous velocity vector of detected particles. All units are in pixels. Reduced sets are displayed for visual clarity; typically, dozens are detected at a time.** ***Right:*** **The heat map.**

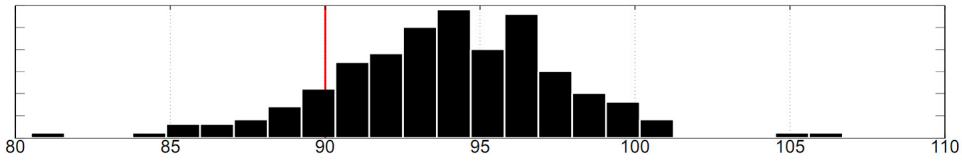

Figure 11. Histogram illustrating the distribution of the data plotted in Fig. 9. Note that the accuracy of the bearing estimation is proportional to the availability of identifiable helix clusters.

III. SLAM formulation with FastSLAM

Our previous experiments[4,5] showed that due to the highly nonlinear nature of the observation equations, traditional nonlinear observers such as EKF do not scale to SLAM in larger environments containing vast numbers of potential landmarks. Measurement updates in EKF require quadratic time complexity due to the covariance matrix it maintains, rendering the data association increasingly difficult as the map grows. An MAV with limited computational resources is particularly impacted from this condition. FastSLAM[11] is a dynamic Bayesian approach to SLAM, exploiting the conditional independence of measurements. A random set of particles is generated using the noise model and dynamics of the vehicle in which each particle is considered a potential location for the vehicle. A reduced Kalman filter (typically containing two states) per particle is then associated with each of the current measurements. Considering the limited computational resources of an MAV, maintaining a set of landmarks large enough to allow for accurate motion estimations, yet sparse enough so as not to produce a negative impact on the system performance is imperative. The noise model of the measurements along with the new measurement and old position of the feature are used to generate a statistical weight. This weight in essence is a measure of how well the landmarks in previous position correlated with the measured position, taking noise into account. Since each of the particles has a different estimate of the vehicle position resulting in a different perspective for the measurement, each particle is assigned different weights. Particles are re-sampled every iteration such that the lower weight particles are removed, and higher weight particles are replicated. This results in a cloud of random particles of track towards the best Kalman Filter results, which are the positions which yield the best correlation between the features previous position and the new measurement data. The positions of landmarks are stored by the particles such as $Par_n = [X_L^T, P]$ where $X_L = [x_{ci}, y_{ci}]$ and P is the 2×2 covariance matrix

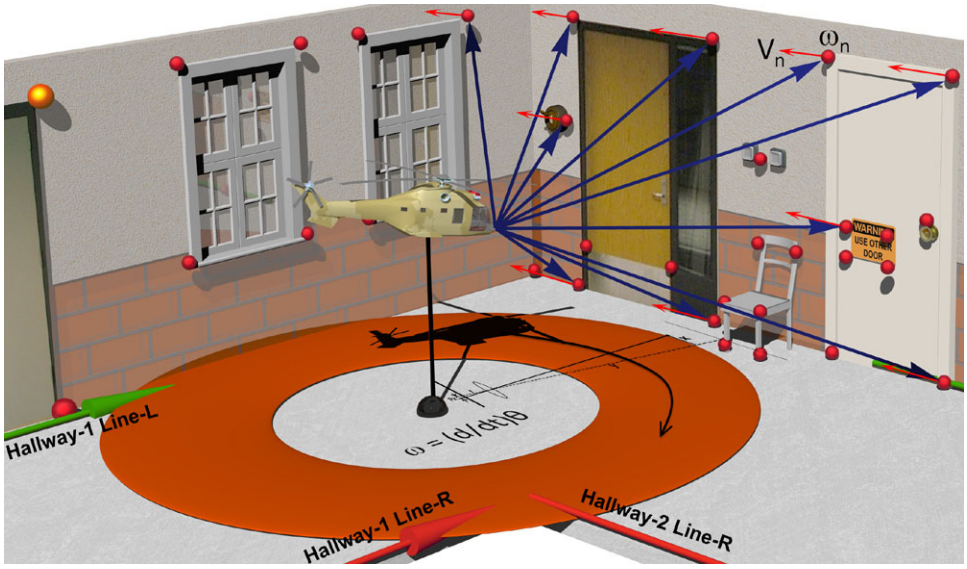

Figure 12. The bearing algorithm exploits the optical flow field resulting from the features not associated with architectural lines. A reduced association set is shown for clarity. Velocities that form statistically identifiable clusters indicate the presence of large objects, such as doors, that can provide estimation for the angular rate of the MAV during the turn.

for the particular Kalman Filter contained by Par_n.

The 6DOF vehicle state, x_v, can be updated in discrete time steps of (k) as shown in (9) where $R = (x_r, y_r, H)^T$ is the position in inertial frame, from which the velocity in inertial frame can be derived as $\dot{R} = v_E$. The vector $v_B = (v_x, v_y, v_z)^T$ represents linear velocity of the body frame, and $\omega = (p, q, r)^T$ represents the body angular rate. $\Gamma = (\phi, \theta, \psi)^T$ is the Euler angle vector, and L_{EB} is the Euler angle transformation matrix for (ϕ, θ, ψ). The 3×3 matrix T converts $(p, q, r)^T$ to $(\dot{\phi}, \dot{\theta}, \dot{\psi})$. At every step, the MAV is assumed to experience unknown linear and angular accelerations, $V_B = a_B \Delta t$ and $\Omega = \alpha_B \Delta t$ respectively.

$$x_v(k+1) = \begin{pmatrix} R(k) + L_{EB}(\phi, \theta, \psi)(v_B + V_B)\Delta t \\ \Gamma(k) + T(\phi, \theta, \psi)(\omega + \Omega)\Delta t \\ v_B(k) + V_B \\ \omega(k) + \Omega \end{pmatrix} \tag{9}$$

The MAV being a helicopter, there is only a limited set of orientations it is capable of being in the air at any given time without partial or complete loss of control. For instance, it cannot generate useful lift when oriented sideways with respect to gravity. Moreover, the on-board autopilot incorporates IMU, gyroscope, and compass measurements in a best-effort scheme to keep the MAV at hover in the absence of external control inputs. Therefore we can simplify the 6DOF system dynamics to a new paradigm of 2D system dynamics with an autopilot. Accordingly, the particle filter then simultaneously locates the landmarks and updates the vehicle states x_r, y_r, θ_r described by,

$$\mathbf{x}_v(k+1) = \begin{pmatrix} \cos\theta_r(k) u_1(k) + x_r(k) \\ \sin\theta_r(k) u_1(k) + y_r(k) \\ u_2(k) + \theta_r(k) \end{pmatrix} + \gamma(k) \tag{10}$$

where $\gamma(k)$ is the linearized input signal noise, $u_1(k)$ is the forward speed, and $u_2(k)$ the angular velocity.

Let us consider one instantaneous field of view of the camera, in which the center of two ground corners on opposite walls is shifted. From the distance measurements described earlier, we can derive the relative range and bearing of a corner of interest (index i) as follows

$$\mathbf{y}_i = \mathbf{h}(\mathbf{x}) = \left(\sqrt{\tilde{x}_i^2 + \tilde{y}_i^2}, \quad \tan^{-1}\left[\frac{\pm \tilde{y}_i}{\tilde{x}_i} \right], \quad \psi \right) \tag{11}$$

where ψ measurement is provided by the Infinity-Point method. This measurement equation can be related with the states of the vehicle and the i-th landmark at each time stamp (k) as shown in (12) where $\mathbf{x}_v(k) = (x_r(k), y_r(k), \theta_r(k))^T$ is the vehicle state vector of the 2D vehicle kinematic model. The measurement equation $\mathbf{h}_i(\mathbf{x}(k))$ can be related with the states of the vehicle and the i-th corner (landmark) at each time stamp (k) as given in (12),

$$\mathbf{h}_i(\mathbf{x}(k)) = \begin{pmatrix} \sqrt{(x_r(k) - x_{ci}(k))^2 + (y_r(k) - y_{ci}(k))^2} \\ \tan^{-1}(\frac{y_r(k) - y_{ci}(k)}{x_r(k) - x_{ci}(k)}) - \theta_r(k) \\ \theta_r \end{pmatrix} \tag{12}$$

A. Data Association

As a prerequisite for SLAM to function properly, recently detected landmarks need to be associated with the existing landmarks in the map such that each measurement correspond to the correct landmark. In essence, the association metric depends only on the measurement innovation vector, often leading to data ambiguity in a three dimensional environment. We developed a new approach to this issue as a faster and more accurate solution is to exploit the pixel locations on the image plane. The pixel locations of the previous features are kept, and compared with that of the current measurements. Given the expected maximum velocity, the maximum expected change in pixel location can be calculated and used for the association threshold. If a match is found, then the association data from last time is used and the large global map does not need to be searched. This saves computation time and since the pixel location is independent of the noise of the vehicle position, it also improves accuracy. If the last frame contains a feature within a threshold of the feature in question, the association information from the previous feature is used. The literature investigates representing the map as a tree based data structure which, in theory, yields an association time of $log(N)$. However, since our pixel-neighborhood based approach already covers over 80% of the features at any time, a tree based solution is not likely to offer a significant benefit (typically 90% of measurements are associated using our pixel neighborhood approach. The remaining 10% are then associated using the nearest neighbor approach of the entire map). The typical data association method is an inefficient algorithm with an $O(N^2)$ time complexity that compares every measurement with every feature on the map and a measurements becomes associated with a feature if it is sufficiently close to it, a process that would exponentially slow down over time, given limited computational resources. Moreover, since the measurement is relative, the error of the vehicle position is additive with the absolute location of the measurement. Prior to the current use of a modified version of FastSLAM,[11] EKF based SLAM was previously tested with the Line-Perspectives method as described in,[5] which employed such an approach and provided updates at 12Hz. FastSLAM provided a speedup up to 20Hz. Further, it is not plagued with performance degradation over time as the map is populated.

B. Experimental Results

As shown in Fig. 13, our monocular vision SLAM correctly locates the corners, associating them with landmarks. During this process, a top-down map of the environment is built. The red circle with the line represents the MAV and its heading, respectively. The blue circles with yellow or green dots inside represent the landmarks. The green dots are landmarks currently being observed by the MAV. The size of the blue circle represents the uncertainty of the landmark position. The uncertainty is known in both x and y direction in the inertial frame, but for display purposes, the worst of the two is used. The size of uncertainty ellipse is intentionally inflated on the display for better visibility. The MAV assumes that it is at $(0, 0)$ Cartesian coordinates at the start of the mission, with the MAV pointed at positive x axis, therefore, the width of the corridor is represented by the y axis. A large ellipse axis represents an inconsistent feature in that direction which might have been introduced when external disturbances are present, for example, a person walking in front of the MAV. Our system is robust to such transient disturbances since the corner-like features that might have been introduced by the walking person will have very high uncertainty, and will not be considered for the map in long term. The current range of our visual radar is a function of camera resolution. The resolution is a trade-off; lower resolution causes features farther away to become statistically inadequate for a consistent detection, whereas higher resolution cameras demand more computational resources.

Figure 13. After completing a single 104 meter loop the MAV returns to the starting point. The proposed ranging and SLAM algorithm closes the loop with less than 2 meters of error. It should be stressed that building floor plan was superimposed on this image after the mission is complete to provide reference data for the ground truth to demonstrate the performance and accuracy of our method. It is not provided to the MAV a-priori. The error is introduced in the third leg of the hallway which consists of flat white walls.

C. The MAV

Saint Vertigo, the autonomous micro helicopter of Iowa State University serves as the robotic test platform for the development of this study. The aircraft is entirely constructed of carbon fiber with hardened-steel reinforcements, and aircraft grade aluminium. The airfoil used in rotor blades is subsonic $NACA0012$, $450mm$ long with no taper chord and swept tips. The unit measures 20" long, capable of producing 1.20 horsepower, and weighs less than $2lbs$. The Figure 14 shows the sixth revision of the aircraft that is 30% lighter and 110% more powerful than previous versions, which is also a 100% self-contained platform. Our earlier versions of Saint Vertigo lacked the power necessary to have on-board image processing capabilities. A wireless camera was used to broadcast video from the MAV, to be processed at a ground station. However, this type of cameras include a low-cost amplitude modulated radio for wireless transmission of analog video, prone to noise and artifacts. Vast amounts of electromagnetic interference is present on an MAV due to high performance AC electric motors and respective motor controllers. This noise is picked up by radios and gets multiplied with the video signal. The result was a multi-modal distribution of random artifacts, causing non-Gaussian perturbations of our visual landmarks. Earlier versions of our proposed SLAM algorithm was based on EKF and non-Gaussian perturbations will cause it to underperform. Unlike the approach presented in,[?] our case would not benefit from the condensation algorithm since the multi-modality was random but not stochastic. With this setup our average SLAM updates were at 7 Hz. Our results[4,5] with wireless pin-hole cameras also exhibited severe radial distortion of the image plane. In response, the wireless video downlink was eliminated, and all the SLAM computations are now performed on board. For this purpose, a lightweight embedded x86 architecture single board computer with SIMD instructions running a stripped version of Linux is considered. Our current camera is digital, featuring non-interpolated 2MP resolution and motorized rectilinear pincushion lens assembly. With this setup, 12 Hz noiseless updates are made possible.

In contrast with other prior works that predominantly used wireless video feeds and Vicon vision tracking system for vehicle state estimation, our algorithms are validated with one of the world's smallest fully self-contained indoor MAV helicopter with a dedicated 1GHz image processing CPU with 1GB RAM, 4GB on-board storage, 802.11 connectivity, and a sophisticated autopilot system (see Fig. 14). Video clips of successful autonomous flight and preliminary results of vision-based navigation are available at http://www.public.iastate.edu/ sjchung/.

IV. Concluding Remarks

This paper introduced a bio-vision inspired monocular ranging and orientation algorithm, coupled with vision-driven SLAM, and a navigation strategy. In this experimental demonstration an autonomous indoor aerial vehicle flew through hallways of a conventional building by literally seeing its environment; a practical solution for autonomous indoor flight and navigation which is also applicable to ground based robots. The light-weight design properties (to address the requirements of a helicopter initially) brings other advantages such as possible uses in wearable computers and helmet mounted mapping devices without having to depend on GPS coverage. Our design does not need extensive feature initialization procedures and is only limited by the capabilities of the camera such as shutter speed, and the availability of good landmarks which are easily found in nearly any non-homogenous environment. While SLAM methods such as fastSLAM are mainly developed for laser range finders, suggested future work includes the development an efficient vision SLAM and data association algorithms that take advantage of the intermediate image processing data.

References

[1]Eklöf, J., "Vision in echolocating bats", Doctoral thesis, Zoology Department, Gteborg University 2003.

[2]D. H. Hubel, T.N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex", J. Physiol, Vol. 160, pp. 106-154, 1962.

[3]M. V. Srinivatsan, "How bees exploit optic flow: behavioural experiments and neural models", Philosophical Transactions: Biological Sciences, Vol. 337, Issue 1281, pp. 253–259

[4]K. Çelik, S.J. Chung, A.K. Somani, "Mono-vision corner SLAM for indoor navigation". *Proc. IEEE Int'l Conf. on Electro-Information Technology*, Ames, Iowa, May 2008, pp. 343–348.

[5]K. Çelik, S.J. Chung, A.K. Somani, "MVCSLAM: mono-vision corner SLAM for autonomous micro-helicopters in GPS denied environments". *AIAA Guidance and Navigation Conf.*, Honolulu, Hawaii. Aug. 2008.

[6]Davison, A., Nicholas, M., and Olivier, S., "MonoSLAM: real-time single camera SLAM," *Pattern Analysis and Machine Intelligence (PAMI)*(29), no. 6, pp. 1052-1067, 2007.

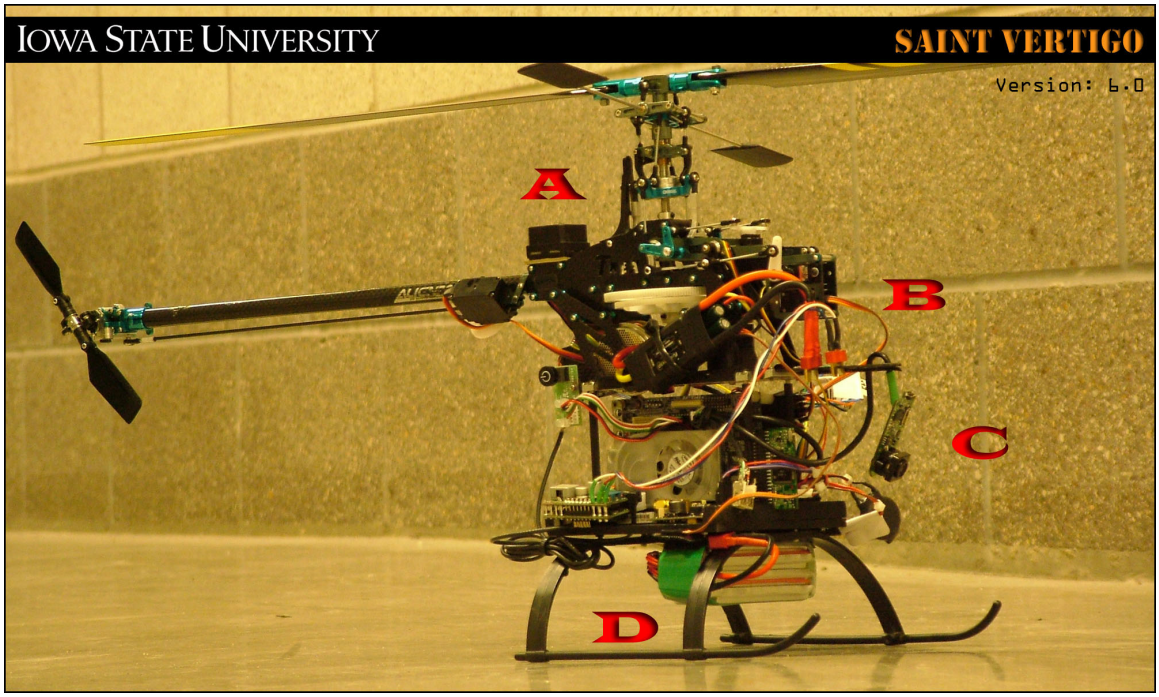

Figure 14. The Saint Vertigo, Iowa State's MAV helicopter, consists of four steel reinforced carbon fiber decks. The A-Deck contains collective pitch rotor head mechanics with mechanical mixing. B-Deck comprises the fuselage which houses the power-plant and controller, transmission, actuators for flight surfaces, rate gyroscope, compass, and the tail rotor. C-Deck contains the brains of Saint Vertigo; an autopilot computer, a navigation computer, an inertial measurement unit, a 2.4 Ghz manual override with double redundancy for fault tolerance, a 900 MHz wireless modem, a 802.11 wireless network device, a 2 megapixel digital video camera, and a collection of sensors including an ultrasonic altimeter, a barometric altimeter, a pitot tube, and a GPS module. D-Deck is the undercarriage designed to act as a shock absorber. This deck also provides a wired ethernet port, an USB port and a VGA port for programming purposes. D-Deck also features hardpoints for attaching the fuel cells, with options available to carry one at the expense of flight time, or two at the expense of maneuverability, depending on the mission requirements.

[7]A. P. Gee, D. Chekhlov, A. Calway, and W. Mayol-Cuevas, "Discovering higher level structure in visual SLAM," *IEEE Trans. Robot.*, vol. 24, no. 5, Oct. 2008.

[8]Harati, A., and Siegwart, R., "Orthogonal 3D-SLAM for indoor environments using right angle corners," *Proc. of the IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS)*, Beijing, China, 2006.

[9]Isoda, N., Terada, K., Oe, S., and IKaida, K., "Improvement of accuracy for distance measurement method by using movable CCD," pp. 29-31, *Sensing Instrument Control Engineering (SICE)*, Tokushima, July 29-31, 1997.

[10]J. Michels, A. Saxena, and A. Y. Ng., "High speed obstacle avoidance using monocular vision and reinforcement learning, " *ICML*, 2005.

[11]M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: a factored solution to the simultaneous localization and mapping problem," *Proc. AAAI Natl Conf. Artificial Intelligence*, 2002.

[12]Ooi, T. L., Wu, B. and He, Z. J., "Distance determined by the angular declination below the horizon," *Nature*, 414, 197200, 2001.

[13]L. M. Paz, P. Pinies, J.D. Tardos and J. Neira "Large-scale 6-DOF SLAM with stereo-in-hand" *IEEE Trans. Robot.*, vol. 24, no. 5, Oct. 2008.

[14]Philbeck, J. W. and Loomis, J. M., "Comparison of two indicators of perceived egocentric distance under full-cue and reduced-cue conditions," *J. Exp. Psychol. Hum. Percept. Perform.* 23, 7285, 1997.

[15]M. Pollefeys, R. Koch, and L.V. Gool, "Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters," Proc. Sixth Intl Conf. Computer Vision, pp. 90-96, 1998.

[16]Ruffier, F. and Franceschini, N., "Visually guided micro-aerial vehicle: automatic take off, terrain following, landing and wind reaction," *Proc. IEEE Int. Conf. on Robot. and Auto.*, 2339-2346, New Orleans, Apr.-May 2004.

[17]F. Ruffier, S. Viollet, S. Amic, and N. Franceschini, "Bio-inspired optical flow circuits for the visual guidance of micro-air vehicles". *Proc. IEEE Int'l Symposium on Circuits And Systems (ISCAS 2003)*, Bangkok, Thailand, vol. 3, pp. 846-849, May 25-28, 2003.

[18]A. Saxena, J. Schulte, A. Y. Ng. , "Depth estimation using monocular and stereo cues," *IJCAI*, 2007.

[19]Sinai, M. J., Ooi, T. L., and He, Z. J., "Terrain influences the accurate judgement of distance," *Nature*, 395, 497500, 1998.

[20]G. Silveira, E. Malis, and P. Rives, "An efficient direct approach to visual SLAM" *IEEE Trans. Robot.*, vol. 24, no. 5, Oct. 2008.

[21]Wu, B., Ooi, T. L., and He, Z. J., "Perceiving distance accurately by a directional process of integrating ground information," *Nature*, vol. 428, pp. 73-77, Mar. 2004.

[22]D. C. K. Yuen and B. A. MacDonald, "Vision-*Based Localization Algorithm Based on Landmark Matching, Triangulation, Reconstruction, and Comparison*," *IEEE Trans. Robot.*, vol. 21, no. 2, Apr. 2005.

[23]Zivkovic, Z., and Heijden, F., "Better features to track by estimating the tracking convergence region," *IEEE Intl. Conf. on Pattern Recognition*, vol. 2, pp. 20635, 2002.

[24]H. Andreasson, T. Duckett, A.J. Lilienthal, "A minimalistic approach to appearance-based visual SLAM", *IEEE Trans. Robot.* vol. 24 pp. 991, October 2008