Horizon Constraint for Unambiguous UAV Navigation in Planar Scenes

Omar Oreifej
Computer Vision Lab
University of Central Florida
Orlando, FL
oreifej@eecs.ucf.edu

Niels Lobo
Computer Vision Lab
University of Central Florida
Orlando, FL
niels@cs.ucf.edu

Mubarak Shah
Computer Vision Lab
University of Central Florida
Orlando, FL
shah@eecs.ucf.edu

Abstract— **When the UAV goes to high altitudes such that the observed surface of the earth becomes planar, the structure and motion recovery of the earth's moving plane becomes ambiguous. This planar degeneracy has been pointed out very often in the literature; therefore, current navigation methods either completely fail or give many confusing solutions in such scenario. Interestingly, the horizon line in planar scenes is straight and distinctive; hence, easily detected. Therefore, we show in this paper that the horizon line provides two degrees of freedom that control the relative orientation between the camera coordinate system and the local surface of earth. The recovered degrees of freedom help linearize and disambiguate the planar flow, and therefore we obtain a unique solution for the UAV motion estimation. Unlike previous work which used the horizon to provide the roll angle and the pitch percentage and only employed them for flight stability, we extract the exact angles and directly use them to estimate the ego motion. Additionally, we propose a novel horizon detector based on the maximum a posteriori estimation of both motion and appearance features which outperforms the other detectors in planar scenarios. We thoroughly experimented on the proposed method against information from GPS and gyroscopes, and obtained promising results.**

I. INTRODUCTION

Vision based navigation is still a challenging problem with many aspects to be explored. One of the main challenges takes place when the scene points are not situated in general positions, but rather on certain degenerate shapes. From such circumstances, the planar degeneracy [1], [2], [3], [4] is crucial for UAVs since it is rather typical for a UAV to fly at a high altitude such that the effects of perturbations of the surface of the earth become negligible; in addition to cases where the UAV is flying over a naturally planar field. Ego motion estimation for such scenario is inherently ambiguous [3] with many solutions arising from the same point correspondence or optical flow.

When discussing motion and structure estimation, it is quite standard to refer to two categories of methods, namely, point-based methods and instantaneous methods. In point-based structure from motion, the epipolar geometry of the scene is computed, from which the relative orientation of the cameras is deduced, and consequently the scene structure is triangulated. The planar degeneracy in the point based methods basically rise from the fact that an infinite number of fundamental matrices satisfy the epipolar geometry constraint if all points lie on a plane [4], this essentially means that the solution has one DOF of ambiguity. However, if intrinsics of the camera are known (calibrated SFM), the

degrees of freedom of the ambiguity can be reduced to two-fold [5]. The chirality constraints can then be applied to resolve this ambiguity unless all points are closer to one viewpoint than the other, in which the motion would be unresolvable [1]. Several methods such as [2], [6], [7] were proposed in order to overcome the planar degeneracy in the epipolar geometry-based SFM by incorporating the knowledge of the coplanarity in the algorithm design. Such works provide sufficient solution to the problem since scene regularities can generally be detected [8]. However, point-based SFM in principle does not fit real-time applications such as UAV navigation because it requires a large baseline such that the epipolar geometry would be correctly defined, therefore it cannot be used to estimate consecutive high frame rate motion [9].

On the other hand, the instantaneous SFM methods rely on differential motion estimation to recover the velocity of the camera based on the optical flow. This continuous motion model is the most appropriate to represent the real time scenario such as the UAV. However, the planar case is also degenerate for this model, and is rather poorly addressed in the literature. State of the art instantaneous methods such as [10], [9], [11] become very unreliable in such scenarios. Additionally, the planar instantaneous methods such as the Homography decomposition [12], [13] give rise to four motion hypotheses which cannot always be disambiguated through the additional processing of applying the chirality constraints; therefore the problem is still unresolved.

Many decent frameworks have been recently reported as successful SLAM techniques for UAVs. In [14], ego motion is recovered in high altitude flights by embedding a homography decomposition in the prediction stage of EKF. The planar degeneracy is tackled by considering a sequence of images rather than two frames. In principal, considering $n > 2$ frames can solve the ambiguity; however, that puts in additional complexities, errors, and limitations; therefore, n-view methods are only considered for off-line pose tracking, where both the pose and the scene structure can be refined through non-linear techniques such as bundle adjustment. Hence, it is a general consensus in the literature that n-view methods are separated from two-view methods.

Conceivably, the closest work to ours is Shakernia et al. [15], where an algorithm for landing a UAV on a planar landing pad was proposed. To recover the motion parameters, they first estimate the differential epipolar geometry con-

straint and then decompose that into four motion hypotheses. In their method, some of the possible solutions could be eliminated; however, in most cases the method will not give a unique answer. For this reason, they assume given desired landing configurations to disambiguate the motion. A similar work was later presented in [16], where Sharp et al. proposed a linear solution to the UAV landing problem followed by a non-linear optimization. The planar ambiguity in the latter method is overcome by the assumption that the world coordinates of the points at the landing pad are previously measured. Similarly, in [17], a given 3D geometry of the scene is employed in the estimation of the camera pose in a planar field. The assumptions from such methods are workarounds which are only feasible when the UAV is landing over a specific area which was previously examined, but not over an arbitrary terrain.

The horizon line was previously employed for stability and control of the UAV in [18], [19], [20], [21], [22], [23]. There are three major differences between all this work and ours: First, our method is a complete ego-motion estimation/SFM algorithm where we compute the actual UAV motion, while the previous work recovered partial motion parameters (eg. the body rates in [23]), or only aimed at providing stability measures for the UAV so that it maintains a level flight, detects extreme attitudes, or points to a predefined location (eg. a runway in [18]); second, such methods mostly use the detected horizon to estimate the roll angle and the pitch percentage (the percentage of the image above the horizon line); our method, by contrast, extracts the exact angles and plugs them into the motion estimation framework; finally, while the aforementioned methods heavily rely on color and texture models for the sky/ground in order to detect the horizon line, our detector extends that by employing additional generic motion and gradients' constraints to provide better horizon detection for planar scenes.

The main contribution of this work is solving the ego motion estimation for planar scenes without ambiguities, meaning that the solution is unique without the need to enforce any additional constraints. The other contributions are: First, modelling the equation of the ground's plane in terms of the UAV's rotation angles and height; second, using the horizon line to acquire the angles directly from the images; third, using a simple yet effective switching scheme that evaluates the planar condition in order to harness both planar and non-planar algorithms. Moreover, we propose a robust maximum a posteriori estimator for better detection of the horizon line. The rest of the paper is organized as the following: In the next section, we derive the equation of local surface of the earth and demonstrate how to use the equation in the proposed planar navigation. Consequently, we present the MAP estimation for the horizon detection in section three. The results and experiments are illustrated in section four. Finally, section five concludes the paper.

II. PLANAR UAV NAVIGATION

We propose to bypass the planar ambiguity through pre-calibration for two rotation angles using the horizon line. In the coming subsection, we show how the calculated angles are used in the equation of the ground plane, a step which will be exploited later to linearize the motion equations.

A. Estimating the Equation of the Ground Plane

Any UAV-Ground relative orientation can be decomposed into a roll then a pitch combination applied to a level camera. The roll angle α depicted in figure 1-a can be directly obtained from the image since it corresponds to the angle formed by the image of the horizon line and the horizontal image direction as illustrated in figure 2-a. This is true because the horizon line rotates exactly by the same amount as the roll of the camera since it is orthogonal to the camera's rotation axis. This imaged angle is invariant to all other motion components since camera translation does not induce image motion to the points at infinity, and the pitch and yaw rotations affect the x and y components of the imaged line with the same amount, theretofore the slope of the horizon will remain constant under such motion. On the other hand, the relation between the pitch angle β and the horizon line is a little more complicated. Previous work such as [18], [19], [20], [21], [22] estimate the pitch percentage which is correlated with the actual pitch. However, for the ego motion estimation we need the exact pitch angle of the UAV; hence, we extend the previous work and find the pitch using the intrinsic parameters of the camera. Let us refer to the point of intersection of the horizon line and the principal axis of a level camera as P, and its image as p. If the camera is initially level with the ground (i.e. $\alpha = 0$, $\beta = 0$), then P will always project at the principal point p_0 as shown in figure 1-a regardless of the height of the UAV. This can also be seen from the perspective projection equation: $p = (X_P f)/Z_P$, where if $Z_P = \infty$, then $p = 0$. Roll is attained by rotating the camera around its principle axis, a sort of motion to which p is invariant since it lies on the principal axis. However, when the UAV undergoes pitch, the imaged point p moves vertically in the image plane. Therefore, the vertical distance between the horizon line and the principal point only depends on the pitch angle as illustrated in figure 2-b. Under perspective projection, the pitch angle is given by $tan^{-1}((p - p_0)/f)$.

The equation of the ground plane in the camera coordinate system can be estimated by the hessian normal form $\hat{\mathbf{n}}.(\mathbf{X} - \mathbf{X}_0) = 0$, where $\hat{\mathbf{n}}$ is the unit normal of the plane, $\mathbf{X} = (X, Y, Z)^T \in \mathbb{R}^3$ corresponds to the world coordinates of a point on the ground plane, and $\mathbf{X_0} \in \mathbb{R}^3$ is an arbitrary point on the plane. As illustrated in figure 1-b, if $\alpha = \beta = 0$, the normal can be taken as $\mathbf{n_0} = (0, 1, 0)^T$ and $\mathbf{X_0} = h\mathbf{n_0}$, where h is the height of the UAV. When roll and pitch are applied, both $\mathbf{n_0}$ and $\mathbf{X_0}$ will rotate by the same amounts, therefore we can always set $\mathbf{X_0} = h\hat{\mathbf{n}}$. Accordingly, the equation of the plane can be rewritten as:

$$\hat{\mathbf{n}}.(\mathbf{X} - h\hat{\mathbf{n}}) = 0 \rightarrow \hat{\mathbf{n}}.\mathbf{X} = h. \tag{1}$$

The rotated unit normal $\hat{\mathbf{n}}$ can be obtained by a two-step rotation of $\mathbf{n_0}$, namely $\hat{\mathbf{n}} = R_\beta R_\alpha \mathbf{n_0}$, where R_β and $R_\alpha \in SO(3)$ are the rotational matrices defined as:

$$R_\alpha = \begin{bmatrix} cos\alpha & -sin\alpha & 0 \\ sin\alpha & cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}, R_\beta = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\beta & -sin\beta \\ 0 & sin\beta & cos\beta \end{bmatrix} \quad (2)$$

Substituting the rotated normal $\hat{\mathbf{n}}$ in equation 1 and rearranging in terms of the depth Z result in the following equation of the ground plane:

$$Z = bX + cY + a, \quad (3)$$

where $a = -h/(sin\beta cos\alpha)$, $b = tan\alpha/sin\beta$, and $c = -1/tan\beta$. Since α and β can be obtained from the image of the horizon line, the values of the b and c parameters in equation 3 can be directly calculated. In the next subsection, we show how equation 3 contributes to determining a unique and linear solution for the ego motion without the knowledge of the height of the UAV.
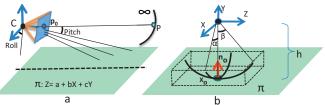

Fig. 1. (a) The point at infinity in the direction of the principal axis (P) always projects on the principal point p_0 if the pitch angle is 0. (b) Rotation of the unit normal around the coordinate system. The ground equation at any instance can be described by rotating the canonical unit normal $\mathbf{n_0} = (0,1,0)^T$ around the camera coordinate system with angles α and β.
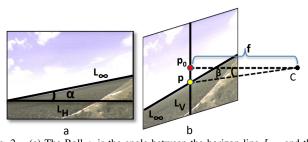

Fig. 2. (a) The Roll α is the angle between the horizon line L_∞ and the horizontal direction L_H. (b) The Pitch β is the angle between the camera principal axis and the line joining the camera center C with point p, where the location of p is determined by the intersection of horizon line L_∞ with the image vertical line L_V passing through the principal point p_0.

B. Planar Ego-Motion Estimation

Under the instantaneous motion model, the velocity of a world's point is given by:

$$\mathbf{X}' = \mathbf{\Omega} \times \mathbf{X} + \mathbf{V}, \quad (4)$$

where $\mathbf{X}' = (X', Y', Z')$ is the velociy, $\mathbf{X} = (X, Y, Z)$ is the coordinates of the point, $\mathbf{\Omega} = (\Omega_1, \Omega_2, \Omega_3)$ is the rotational velocity, and $\mathbf{V} = (V_1, V_2, V_3)$ is the translational velocity. If the world coordinate is attached to the camera, the image coordinates of the points under perspective projection will be $x = (Xf)/Z$, and $y = (Yf)/Z$; therefore, the induced image motion is:

$$u = x' = (fX')/Z - (xZ')/Z, \quad (5)$$
$$v = y' = (fY')/Z - (yZ')/Z.$$

Replacing $\mathbf{\Omega}$ from 4 with its skew symmetric matrix will transform the cross product into matrix multiplication. Consequently, substituting the new equation in 5, we obtain

$$
\begin{aligned}
u &= f(V_1/Z + \Omega_2) - (xV_3)/Z - \Omega_3 y \quad (6)\\
&\quad - (\Omega_1 xy)/f + (\Omega_2 x^2)/f, \\
v &= f(V_2/Z - \Omega_1) - (yV_3)/Z + \Omega_3 x \\
&\quad - (\Omega_2 xy)/f - (\Omega_1 y^2)/f.
\end{aligned}
$$

All the points in the scene conform to the equation of the ground plane. Therefore, we substitute equation 3 in 6 to get

$$
\begin{aligned}
u &= q_1 + q_2 x + q_3 y + q_4 x^2 + q_5 xy, \quad (7)\\
v &= q_6 + q_7 x + q_8 y + q_4 xy + q_5 y^2,
\end{aligned}
$$

where

$$
\begin{array}{ll}
q_1 = fV_1/a + f\Omega_2, & q_2 = -V_1 b/a - V3/a, \\
q_3 = -V_1 c/a - \Omega_3, & q_4 = q_4 = V_3 b/af + \Omega_2/f, \\
q_5 = V_3 c/af - \Omega_1/f, & q_6 = fV_2/a - f\Omega_1, \\
q_7 = -V_2 b/a + \Omega_3, & q_8 = -V_1 c/a + V_3/a.
\end{array}
$$

Equations 7 are the instantaneous pseudo-perspective projection equations which represent the planar flow [24]. The optical flow at a certain image point gives rise to two sets of equations of such form. Therefore, 4 points should be sufficient to solve for the coefficients $\mathbf{q} = (q_1...q_8)$. However, since ground points typically do not lie exactly on a plane, using more points provides a better estimation which settles on the least squared error solution. Therefore, we stack the optical flow values from all feature points and solve the following set of linear equations for the values of $\mathbf{q}$:

$$
\underbrace{\begin{bmatrix} 1 & x & y & x^2 & xy & 0 & 0 & 0 \\ \multicolumn{8}{c}{\dots} \\ \multicolumn{8}{c}{\dots} \\ \multicolumn{8}{c}{\dots} \\ \hline 0 & 0 & 0 & xy & y^2 & 1 & x & y \\ \multicolumn{8}{c}{\dots} \\ \multicolumn{8}{c}{\dots} \\ \multicolumn{8}{c}{\dots} \end{bmatrix}}_{M_{2n \times 8}} \underbrace{\begin{bmatrix} q_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ q_8 \end{bmatrix}}_{Q_{8 \times 1}} = \underbrace{\begin{bmatrix} U \\ ---- \\ V \end{bmatrix}}_{U_{2n \times 1}}
$$

The parameters of the pseudo-perspective transformation can also be found through featureless techniques which employ the image spatiotemporal derivatives [25]. The coefficients $\mathbf{q}$ provide 8 non-linear equations in terms of the translational and rotational velocities of the UAV, and the equation of the ground plane. The solution for the equations is ambiguous since they embed nine degrees of freedom in only eight equations. We propose to resolve the ambiguity by first acquiring the values of b and c parameters based on the horizon line as described in section 2.1. Consequently, we use the following new set of equations obtained by subtraction and taking ratios of the components of $\mathbf{q}$:

$$
\begin{aligned}
(q_3 + q_7)/(f^2 q_5 - q_6) &= (-V_1 c - V_2 b)/(fV_3 c - fV_2) \quad (8)\\
(q_3 + q_7)/q_8 &= (-V_1 c - v_2 b)/(-V_2 c - V_3) \\
(f^2 q_5 - q_6)/q_8 &= (fV_3 c - fV_2)/(-V_2 c - V_3).
\end{aligned}
$$

This step eliminates the parameter a which contains the height of the UAV, and transforms the equations into three linearly independent equations of only the translational velocity. The LHSs of the above equations are constants; hence, they are easily solved by rearranging to the form $Ax = 0$, and then finding the null space of A. Since the equations were obtained by taking ratios, the resultant translation vector is up to a signed scale, therefore we define the correct translational velocity in terms of the recovered one as

$$\mathbf{V} = k\mathbf{V}'. \tag{9}$$

Returning to the set of equations for $\mathbf{q}$, we add q_3 and q_7

$$q_3 + q_7 = -V_1 c/a - V_2 b/a. \tag{10}$$

Substituting 9 in 10 and replacing the plane parameter a with $-h/(sin\beta cos\alpha)$ then rearranging the result, we obtain:

$$k/h = -(q_3 + q_7)/((-V_1' c - V_2' b)(sin\beta cos\alpha)). \tag{11}$$

Solving for the ratio k/h resolves the sign of the translational velocity because the height (h) of the UAV is always negative; therefore, if the recovered ratio is positive, the correct translation direction will be the inverse of the recovered value. In order to solve for the rotational velocity component $\mathbf{\Omega}$, we rewrite equations q_1, q_6, and q_7 in terms of equation 9; accordingly, we end up with:

$$
\begin{aligned}
\Omega_1 &= -(sin\beta cos\alpha)v_2'(k/h) - q_6/f, \\
\Omega_2 &= q_1/f - v_1'(k/h)(sin\beta cos\alpha), \\
\Omega_3 &= q_7 - v_2'b(k/h)(sin\beta cos\alpha).
\end{aligned}
\tag{12}
$$

On the other hand, if the height of the UAV was available (for instance from an altimeter), recovering b and c from the horizon line directly transforms equations $(q_1...q_8)$ into linear with only seven unknowns, in this case a metric solution can be directly obtained without any further steps.

At this stage, we demonstrated how the motion can be uniquely and linearly recovered under the proposed framework. Various iterative operations could be applied as refinement; for example, RANSAC-based model fitting could be employed to reject the outliers from the optical flow. Moreover, global non-linear optimization and filtering can be used to refine the recovered motion. Further elaboration on such methods is out of the scope of this paper; hence, we focus only on the linear instantaneous estimation of motion from a pair of frames.

C. Switching Scheme

Throughout a typical UAV flight, the scene geometry changes between planar and non-planar, based on the land structure and the altitude of UAV. Hence, the navigation system should be aware of such conditions in order to switch to the algorithm that fits the current state. Therefore, before processing frames, we evaluate the fitness of the estimated planar model for the motion of points on a plane by computing the ℓ_2 norm error of the model (pe):

$$pe = (\sum_i^n \|U_i - U_i'\|)/n, \tag{13}$$

where n is the number of feature points, U is the optical flow vector $[u\ v]^T$, and U' is the estimated image velocity vector $[u'\ v']^T$ computed as in equation 7 using the points' coordinates and the estimated model parameters. We additionally experimented on other planer image motion models such as the planar differential essential matrix [15] and the instantaneous Homography [12]. Such models are also based on the least squared error estimation and therefore they provide error responses that are identical to ours. We found in our experiments that assigning an empirical threshold to the computed error efficiently determines if the scene planarity is adequate for the model. Interestingly, regular non-planar SFM techniques which are unreliable in planar scenarios regain reliability at the same threshold. Therefore we use the computed error to switch between our proposed planar ego and non-planar methods, from which we select the renormalization [10] in view of the fact that we found it robust in our experiments.

III. MAP FOR HORIZON DETECTION

Given the type of scenes we are dealing with which are significantly lacking in objects and features, horizon detection methods which depend on the geometry of the objects in the scene such as [26], [27] do not perform desirably. In addition, methods which employ dynamic texture [28] or homogenous texture [29] do not fit our framework since the ground texture can be entirely random. Interestingly, in our case, the horizon line is physically apparent and considerably distinctive in the image; furthermore, it is always approximately straight in the planar scenes since there are no major perturbations such as mountains. Therefore, sky-ground segmentation methods such as [20], [21], [19] perform very well in our scenario. However we noticed that such methods heavily rely on appearance (color and texture) to model the ground/sky, therefore they occasionally fail due to lighting and weather changes. For this reason, we additionally employ the less scene dependent features of gradients and motion to enhance the performance in a maximum a posteriori estimator (MAP) for the location of the correct horizon line. Using bayes rule, the posterior distribution of the unknown horizon line L given an image I is defined as

$$P(L|I) \quad \propto \quad P(I|L)P(L). \tag{14}$$

Applying the negative logarithm on equation 14, we get

$$-\log P(L|I) \quad \propto \quad -\log P(I|L) - \log P(L). \tag{15}$$

We employ three energy functions λ_1, λ_2, and λ_3 to represent the measurement likelihood $P(I|L)$. Our experiments provided evidence that the energy term proposed by Ettinger et al. [20] performs desirably in most scenarios. Therefore, we use the same term and set $\lambda_1(L) = 1/(|\sum_{Ab}| + |\sum_{Un}| + Tr(\Lambda_{Ab})^2 + Tr(\Lambda_{Un})^2)$, where $\sum_{Ab}, \sum_{Un}$ are the covariance matrices of colors above and under L respectively, and $\Lambda_{Ab}, \Lambda_{Un}$ are the diagonal eigen values' matrices for the covariance matrices. Since the horizon line in the image

will separate the sky from the ground, it is expected that it will attain the highest gradient of intensity across the image; therefore, we assign $\lambda_2(L) = G_{av}(L)$, where G_{av} is the average gradient magnitude across the line. On the other hand, it was shown in [29] and [28] that the gradient of the perspective effects of a plane will be orthogonal to its vanishing direction. Hence, the gradient of motion magnitude should be orthogonal to the correct vanishing line. This can be interpreted as $\lambda_3(L) = \sum_{j=1}^{n} (G_{U_j}^T \cdot \hat{l})^2$, where G_{U_j} is the gradient of optical flow magnitude at pixel j, and $\hat{l}$ is the unit normal to the line. The final conditional likelihood function $P(I|L)$ is in exponential form of the combined energy terms

$$logP(I|L) \propto (\xi_1\lambda_1(L) + \xi_2\lambda_2(L) + \lambda_3(L)), \quad (16)$$

where ξ_1 and ξ_2 are weighting parameters that we learn using training sequences. $P(L)$ in equation 15 is the line prior to which we assign the euclidean distance between the line parameters (slope and intercept) and the parameters of the detected horizon line in the previous frame. This prior is crucial since it acts as a regularization for the likelihood term. To find the most likely (MAP) horizon L, we can simply minimize the negative log likelihood

$$L = \arg\min_{L_i}[-\log P(I|L_i) - \log P(L_i)]. \quad (17)$$

Instead of applying an iterative or exhaustive optimization for the inference of the MAP horizon line, we obtain a set of candidates to test against the probability function using the robust line detector in [30], where we form support regions for lines by connecting the components of gradients which lie in the same direction, and then obtain the line parameters using eigen vector decomposition for the covariance matrix of the support region. The eigen vector with the highest eigen value will correspond to the major direction of the line and the other eigen vector can be neglected; therefore, this line detection algorithm does not require the lines to be exactly straight, and that comes in handy when dealing with scenarios where the curvature of the earth is affecting the horizon line. Once the candidate lines are detected, they are evaluated against the MAP function and the line which attains the minimum is selected. Figure 3 shows the steps followed to detect the horizon line on a sample frame. Since motion features are already computed for the ego motion estimation, the proposed horizon detector does not add any considerable computational overhead.

IV. EXPERIMENTS AND DISCUSSION

We conducted extensive experiments on the proposed method using several synthetic and real sequences. In all of the experiments, we integrate the recovered translational velocity along the sequence in order to obtain a translation track that is more convenient to compare with the groundtruth translation. All measurements recovered are units of translation or rotation per frame. For synthetic experiments, we developed a flight simulator application that models a pinhole camera with perspective projection flying over a star field.
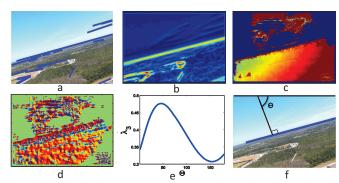

Fig. 3. Horizon line detection (Jet color map is used). (a) Detected candidate lines for a sample frame. (b) Intensity gradient magnitude used in λ_2 function. (c) Optical flow magnitude. (d) Angles of the gradient of optical flow magnitude (e) λ_3 response to all possible line directions. Maximum response is at $\theta \simeq 50$. (f) Resultant horizon line after combining responses from λ_1, λ_2, and λ_3.

Figure 4 shows an example for the recovered rotation and translation compared to the groundtruth after running the simulation for 30 frames of random motion.
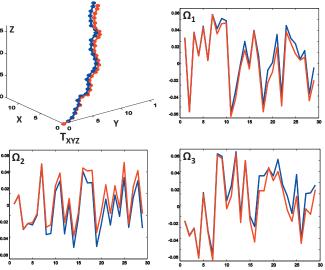

Fig. 4. Recovered translation track Txyz and rotational velocities (Ω_1, Ω_2, Ω_3). The groundtruth (red curves) and our results (blue curves) are almost entirely correlated.

Additionally, we performed a test for measuring the sensitivity to the planar condition. In that, random perturbations in the heights of the stars were added incrementally to the scene in order to resemble the irregularity of the surface of the earth. The perturbations were drawn from a Gaussian distribution with zero mean and a variance relative to the height of the UAV. Figure 5 shows the performance of the proposed planar method and the switching scheme compared to state of the art algorithms: Jepson and Heeger's subspaces [11] in which the translational velocity is recovered by computing its orthogonal constraints that do not include rotational components of flow; Kanatani's renormalization [10] which employs the instantaneous epipolar constraint represented in terms of the twisted flow and the motion parameters; in addition to the homography decomposition (4-point algorithm) as proposed by Faugeras [13] and more
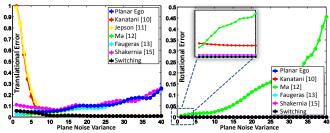
Fig. 5. Normalized errors in translational and rotational velocities versus the plane noise variance. The range of the variance corresponds to $(0-40)\%$ of the height of the UAV; thus, the higher the noise variance, the less planar is the scene. The performance of the proposed Planar Ego, the Switching, Faugeras, and Ma are superior and similar when the scene is planar (though Ma's performance decays ungracefully in recovering rotational velocity). However, our planar ego provides a unique solution while the other planar methods are ambiguous. The switching scheme detects the non-planarity starting from variance 10 and accordingly switches to the renormalization, therefore it provides the best overall performance.

Fig. 6. (a) The UAV and the equipments used in the experiments setup. (b) Four example frames with horizon detection and the computed roll angle α and pitch angle β. (c) Optical flow overlaid on an example frame.

recently by Ma et al. [12]. It is rather essential to note that the more recent state of the art methods for motion and structure such as [5] and [31] belong to the point-based SFM and therefore do not fit this framework since they become singular when the motion is estimated between consecutive frames (i.e. when translation is relatively small). As expected, the subspaces and the renormalization approaches suffer in planar scenarios because the planar scene does not provide enough constraints for such algorithms. On the other hand, results from Faugeras and Ma are similar to ours; however, they give rise to an ambiguous set of four motion hypotheses which cannot always be disambiguated, while our proposed method produces a unique answer. For the purpose of comparison with such methods, we select the motion hypothesis closest to the groundtruth. Moreover, the switching scheme detects non-planarity and switches to the non-planar renormalization, thus providing superior and consistent performance.

In addition to the synthetic data, we evaluated our ego motion algorithm on challenging real data. We collected several sequences by flying a ground-controlled balloon-based UAV that moves in full 6 DOF. The UAV is also equipped with gyroscopes for measuring orientation, and a GPS for measuring location. Additionally, we calibrated the camera to obtain its intrinsic parameters. The testing terrains comprised a variety of approximately planar fields which included land perturbations caused by trees and small buildings such that the perturbations did not generally exceed 10% of the height of the UAV. The first step of the algorithm is to detect the horizon line, for which we applied the MAP detector on the collected sequences. The proposed detector attained 100% detection rate in our sequences closing the gaps experienced when using solely the first term λ_1 from [20] which achieved a detection rate of 93%. On the other hand, using only λ_2 and λ_3 achieved 91%; therefore, it is clear that the proposed combination of functions is rather robust. It is worth mentioning that we preprocess the frames by applying a simple histogram equalization in order to improve the contrast for better horizon detection and motion estimation in hazy scenarios. A more robust haze removal
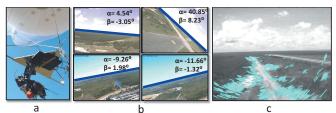
technique such as [32] can be applied when severe haze is present. Figure 6 shows our experiments' setup and example frames with the detected horizon which was used to compute the roll and pitch angles. For the motion estimation, we use sparse optical flow computed over Harris corners that lie under the horizon line (we discard sky features).

Figure 7 shows the obtained rotations compared to the gyroscopes, and the translations compared to the GPS. It is clear that the proposed method provides a robust ego motion estimation with minor errors due to the noisy optical flow. Errors in the UAV translation track are expected for several factors: First, our UAV is balloon-based and translates very slowly, therefore the actual translation is comparable to the noise; second, the errors are cumulative across the track; third, GPS sensitivity does not cover the motion between each two frames; finally, the location provided by our GPS contains only latitude and longitude; therefore, changes in altitude are not present in the groundtruth.

Moreover, we evaluated the sensitivity of the proposed method to the horizon line estimation. Figure 8 shows the error in the recovered motion versus the error in the angles estimated from the horizon line. To obtain the curves, we vary the horizon line parameters around the groundtruth horizon, estimate the roll and pitch angles, and then compute the motion. We further evaluated the proposed planar ego motion estimation along with the other motion estimation techniques by running them for the entire real sequences (around 5000 frames). Figure 9 depicts the normalized average error of translational and rotational velocities. The comparison demonstrates superior performance for the proposed method with being generally close to the other planar algorithms. However, our method provides a unique answer for the motion while the other planar algorithms give rise to a set of ambiguous solutions, from which we compare with the solution closest to the meta-data.

V. CONCLUSION

We have presented a novel linear, robust, and easy to implement ego motion estimation algorithm designed specifically to aid navigation in planar scenes. The method exploits the detected horizon line to provide the relative roll and pitch angles of the UAV and reduce the complexity of recovering the camera motion into solving a set of linear equations for the translational velocity, followed by solving another set of linear equations for the rotational velocity. We showed by experiments that our method provides competitive
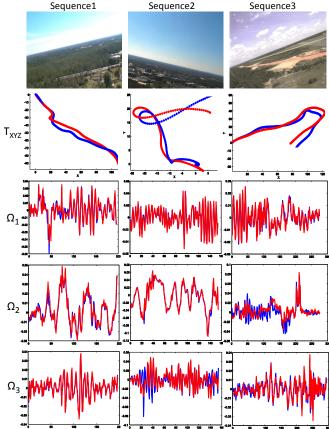
Fig. 7. Translation tracks Txyz and rotational velocities (Ω_1, Ω_2, Ω_3) for three example sequences. Red curves show the groundtruth and blue curves show the results from our method. The recovered translation tracks are very close to the GPS tracks but do not exactly conform with them for several reasons discussed in the text. The recovered rotational velocities are almost exactly the same as the gyroscopes such that the two curves overwrite each other, except for few errors caused by the noisy optical flow.
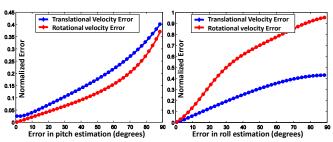

Fig. 8. Error in the recovered motion versus error in the estimated horizon. Left: Motion error versus pitch error. Right: Motion error versus roll error.
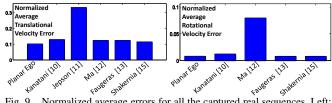

Fig. 9. Normalized average errors for all the captured real sequences. Left: Translational velocity error against the GPS. Right: Rotational velocity error against the gyroscopes.

results with being superior to the other methods in terms of providing a unique unambiguous solution without the need to enforce any additional constraints.

REFERENCES

[1] H. Longuet-Higgins. The reconstruction of a plane surface from two perspective projections, 1986. Proc. R. Soc. Lond. B.

[2] Richard Szeliski and Philip Torr. Geometrically constrained structure from motion: Points on planes. In *SMILE: European Workshop on 3D Structure from Multiple Images of Large-Scale Environments*, 1998.

[3] H. Longuet-Higgins. The visual ambiguity of a moving plane, 1984. Proc. R. Soc. Lond. B.

[4] R. Hartley and A. Zisserman. Multiple view geometry in computer vision. *JCambridge University Press*, 2004.

[5] D. Nister. An efficient solution to the five-point relative pose problem. In *CVPR*, 2003.

[6] Ondrej Chum, Tomas Werner, and Jiri Matas. Two-view geometry estimation unaffected by a dominant plane. In *CVPR*, 2005.

[7] Marc Pollefeys, Frank Verbiest, and Luc Van Gool. Surviving dominant planes in uncalibrated structure and motion recovery. In *ECCV*, 2002.

[8] Kenichi Kanatani. Geometric information criterion for model selection. *IJCV*, 1998.

[9] Y. Ma, J. Koseck, and S. Sastry. Linear differential algorithm for motion recovery: A geometric approach. *IJCV*, 1998.

[10] K. Kanatani. 3-d interpretation of optical flow by renormalization. *IJCV*, 1993.

[11] Allan Jepson and David J. Heeger. Linear subspace methods for recovering translational direction. In *Spatial Vision in Humans and Robots*, 1992.

[12] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry. An invitation to 3d vision. *The MIT Press*, 2003.

[13] O.Faugeras. Three dimensional computer vision. *The MIT Press*, 1993.

[14] F. Caballero, L. Merino, J. Ferruz, and A. Ollero. Vision-based odometry and slam for medium and high altitude flying uavs. *J. Intell. Robotics Syst.*, 2009.

[15] Omid Shakernia, Yi Ma, T. John Koo, T. John, and Shankar Sastry. Landing an unmanned air vehicle: Vision based motion estimation and nonlinear control. *Asian Journal of Control*, 1999.

[16] C.S. Sharp, O. Shakernia, and S.S. Sastry. A vision system for landing an unmanned aerial vehicle. 2001.

[17] G. Schweighofer and A. Pinz. Robust pose estimation from a planar target. *PAMI*, 2006.

[18] Andrew Miller, Mubarak Shah, and Don Harper. Landing a uav on a runway using image registration. In *IEEE Conference on Robotics and Automation*, 2008.

[19] T. D. Cornall, G. K. Egan, and A. Price. Aircraft attitude estimation from horizon video. In *IEE Electronics Letters*, 2006.

[20] S. M. Ettinger, M. C. Nechyba, P. G. Ifju, and Martin Waszak. Vision-guided flight stability and control for micro air vehicles. In *IEEE Int. Conference on Intelligent Robots and Systems*, 2002.

[21] Sinisa Todorovic and Michael C. Nechyba. Sky/ground modeling for autonomous mav flight. In *IEEE International Conference on Robotics and Automation*, 2003.

[22] S. Todorovic and M. C. Nechyba. A vision system for intellignet mission profiles of micro air vehicles. In *IEEE Transactions on Vehicular Technology*, 2004.

[23] Damien Dusha, Wageeh Boles, and Rodney Walker. Fixed-wing attitude estimation using computer vision based horizon detection. *12th Australian International Aerospace Congress*, 2007.

[24] James R. Bergen, P. Anandan, Keith J. Hanna, and Rajesh Hingorani. Hierarchical model-based motion estimation. In *ECCV*, 1992.

[25] Steve Mann and Rosalind W. Picard. Video orbits of the projective group: A simple approach to featureless estimation of parameters. *IEEE Transaction on Image Processing*, 1997.

[26] Derek Hoiem, Alexei A. Efros, and Martial Hebert. Putting objects in perspective. In *CVPR*, 2006.

[27] J. Sivic, B. Kaneva, A. Torralba, S. Avidan, and W.T. Freeman. Creating and exploring a large photorealistic virtual space. 2008.

[28] Yaser Sheikh, Niels Haering, and Mubarak Shah. Shape from dynamic texture for planes. In *CVPR*, 2006.

[29] A. Criminisi and A. Zisserman. Shape from texture: Homogeneity revisited. 2000.

[30] Jana Kosecka and Wei Zhang. Video compass. In *ECCV*, 2002.

[31] Marc Pollefeys, Luc Van Gool, Maarten Vergauwen, Frank Verbiest, Kurt Cornelis, Jan Tops, and Reinhard Koch. Visual modeling with a hand-held camera. *IJCV*, 2004.

[32] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *CVPR*, 2009.