Attitude Estimation for a Fixed-Wing Aircraft Using Horizon Detection and Optical Flow

Damien Dusha, Wageeh Boles, Rodney Walker
Australian Research Centre for Aerospace Automation (ARCAA)
Queensland University of Technology
PO Box 2434, Brisbane 4001, AUSTRALIA
d.dusha@qut.edu.au

Abstract

We develop a method for estimating the flight critical parameters of pitch angle, roll angle and the three body rates using horizon detection and optical flow. We achieve this through the use of an image processing front-end to detect candidate horizon lines through the use of morphological image processing and the Hough transform. The optical flow of the image for each candidate line is calculated, and using these measurements, we are able to estimate the body rates of the aircraft. Using an Extended Kalman Filter (EFK), the candidate horizon lines are propagated and tracked through successive image frames, with statistically unlikely horizon candidates eliminated.

Results qualitativly describing the performance of the image processing front-end on real datasets are presented, followed by an analysis of the improvement when utilising the motion model of the vehicle.

1. Introduction

Under visual flight rules, human pilots are specifically trained to control the attitude of the aircraft with respect to the horizon. With the horizon being such a valuable attitude reference to a human pilot, one would expect that an estimate of pitch and roll could be obtained through machine vision detection and processing of the horizon.

This paper makes two contributions to the field of horizon-based attitude estimation. Firstly, an emphasis is placed on the *estimation* of the attitude and secondly, particular attention is paid to the use of temporal information to prevent false matches, a contradiction to previous authors who explicity avoid track for fear of locking onto a false match [10, 2].

Aside from direct attitude estimation, this technique may have other applications such as overcoming the tranlation-rotation ambiguity in optical-flow [11] which may assist in Structure from Motion problems or to constrain the drift from integrated sensors such as gyroscopes.

The paper is organized as follows. Firstly, a brief survey on published horizon detection approaches are presented, along with their limitations and drawbacks. Following this, the image processing front-end for detecting candidate horizon lines is developed. Next, the framework for deriving the flight parameters from the horizon is presented. Finally, results are presented for processing real images sequences captured during flight tests over South-East Queensland. The paper concludes with the outlook for future work.

1.1. Existing Work

Using horizon detection for attitude determination is not a new idea. Todorovic *et.al.*[10] treated the horizon detection problem as a subset of image segmentation and object recognition, and used a percentage of the sky seen as an error signal to a flight stability controller on a Micro Air Vehicle (MAV). The resulting system was stable enough to be safely flown by an untrained operator in real time.

In contrast, Bao *et.al.* [2] use a direct edge-detection technique, followed by automatic thresholding and a Hough-like algorithm to generate a "projection statistic"' for the horizon. It claims a 99% success rate over several hours of video. Importantly, it deals only with detection, not estimation of attitude.

The image processing front-end proposed here is superficially similar to Bao's algorithm in that it uses an edge detection technique followed by a Hough transform. However, we propose a substantially different

0-7695-3067-2/07 $25.00 © 2007 Crown Copyright
DOI 10.1109/DICTA.2007.89

methodology for pre-filtering the image. In addition, it is explicitly designed to act as a front-end for further processing by the attitude estimation filter.

1.2. Image Processing Frontend

An outline of the new algorithm is shown in Figure 1. The key feature of this algorithm is that the processing is applied in parallel to each of the red, green and blue colour planes of the original image in order to exploit the observation that the horizon tends to be correlated in all channels.

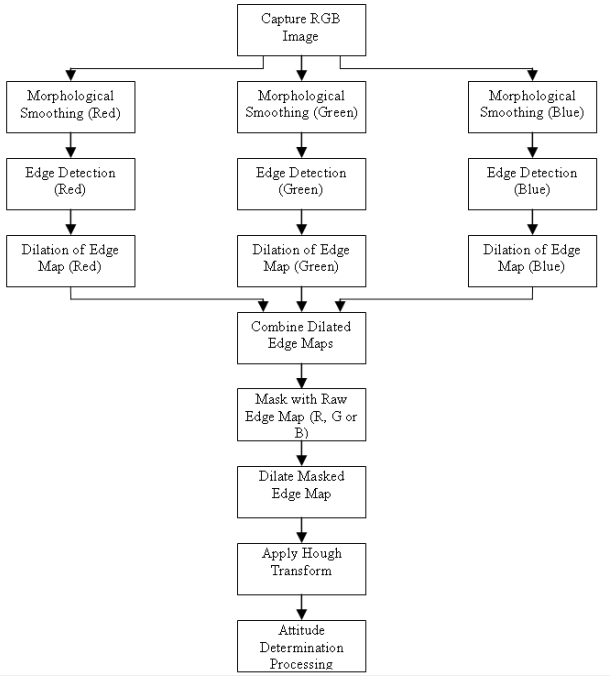


Figure 1. Image Processing Front-End for the Horizon Detection Algorithm

Morphological smoothing [8] with a large circular structuring element (relative to the size of the image) is applied to each channel of the RGB image. The filter was chosen specifically for its edge-preserving properties and the fact that spurious reponses will generally be curved as a result of the circular stucturing element and will be later rejected by the Hough transform.

Edge detection and thresholding using Sobel is then performed on each of the filtered channels. It can be visually seen in Figure 2 that the horzion tends to be reasonable well correlatd between all edge maps, whilst spurious responses tend to be localised on only one or two of the image planes. As a result, the edge maps are combined with pixelwise AND operation. Note that clutter from cloud is significantly reduced in the blue channel when compared to the other channels.

Figure 2. Result of algorithm after smoothing and edge detection, Top: Original, Red Plane, Bottom: Green Plane, Blue Plane

Once the channels are combined, the Hough Transforme is used for detecting edges that may or may not correspond to the horizon. Because of the large structuring element, even reasonably rough terrain on the horzon is well detcted by this stage.

In many instances, the strongest response from the Hough transform is sufficient to correctly detect the horizon. However, if stronger lines are present in the image, then these will typically have the largest response and will therefore result in an incorrect detection of the horizon. This effect is shown in Figure 3, where the strongest edge in one of the images corresponds to a taxiway.

(a) Sun Glare (b) Building (c) Taxiway

Figure 3. Image Processing Front End Examples

Therefore, to mitigate against false matches, we may take advantage of the motion model of the aircraft to differentiate between candidates that may or may not correspond to the horizon.

2. Deriving an Attitude Estimate from the Horizon Line

The image processing front-end presented in the preceding section is capable of detecting a number of lines in the image that are candidates for the horizon. That is, we can effectively consider the front-end to be a sensor, measuring a line on the image plane which may or may not correspond to the horizon. The measurements produced by the "sensor" need to be related to the state vector of the vehicle if the horizon is to be effectively used in the attitude estimation problem.

2.1. A Formalisation of the Horizon Attitude Estimation Problem

We define the world co-ordinate frame to be coincident with the local tangent frame at a point where the aircraft's gravity vector intersects with the Earth's surface. That is, the aircraft is assumed to be directly above the world coordinate frame. Furthermore, we assume that the axes of the camera (sensor) are coincident with axes of the body-fixed co-ordinate frame of the aircraft.

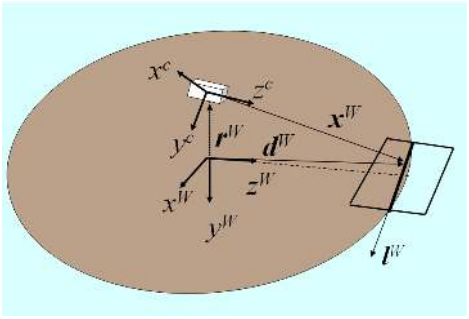


Figure 4. Definition of Coordinate System

Locally, the world is assumed to be a flat disk, passing through the origin of the world co-ordinate frame and normal to the gravity vector of the aircraft, with the "edge of the earth" corresponding to the horizon. Therefore, the horizon (if in view) will appear in the image plane as a curve. However, if the field of view of the camera is narrow, then the arc visible to the image plane can be closely approximated by (and effectively be coincident with) a secant. This is illustrated in Figure 4.

The coordinate frame is designed to be convenient for expressing the presented formulae in terms of the image coordinates. The z-axis of the world frame is defined as the line from the origin to the centre of the arc on the horizon viewed by the camera. The y-axis points downwards toward the centre of the earth, perpendicular to the surface plane and the x-axis completes the right hand coordinate system. The z-axis of the camera is defined to be the optical axis of the camera and lies in the same plane as the z-axis of the world co-ordinate system. The x-axis of the camera frame is parallel to the top edge of the image plane of the camera (i.e. runs "right" along the image) and the y-axis completes the right-hand coordinate system.

The attitude of the aircraft is the rotation from the world frame to the camera frame. This is achieved through a pitch angle θ about the x-axis, followed by a roll angle ϕ about the z-axis. Since yaw, ψ is unobservable in this problem, we assume the angle is zero and therefore the associated rotation matrix is simply the identity matrix. Euler angles have been chosen for this formulation because of the analytic solution they yield, and since the horizon will not be in view of a projective camera when singularities occur in the rotation matrix.

For the order of rotation given (yaw-pitch-roll), the rotation of a vector from the world coordinate frame to the camera (body) fixed coordinate frame is given by [6]:

$$\mathbf{x}^c = \mathbf{R}_{\mathbf{z}}(\phi)\mathbf{R}_{\mathbf{x}}(\theta)\mathbf{R}_{\mathbf{y}}(\psi)\mathbf{x}^w \quad (1)$$

$$\mathbf{x}^c = \mathbf{R}^c_w\mathbf{x}^w \quad (2)$$

where $\mathbf{R}^c_w$ is the rotation from thje world frame to the camrea frame, the superscript of w denotes a vector expressed in the world frame, and c denotes a vector expressed in the camera frame.

Since the surface of the earth is approximated by a plane, we can describe a normal vector to the plane $\mathbf{n}^W$ as:

$$\mathbf{n}^w = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T \quad (3)$$

The horizon line can be described as a point $\mathbf{d}^W$ and a direction vector $\mathbf{l}^W$:

$$\mathbf{d}^w = \begin{bmatrix} x & 0 & d \end{bmatrix}^T \quad (4)$$

$$\mathbf{l}^w = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T \quad (5)$$

where x is an arbitrary point along the x-axis and d is the distance to the horizon along the z-axis.

Since the camera is directly above the origin of the world coordinate system, the position of the camera $\mathbf{r}^w$ can be described as:

$$\mathbf{r}^w = \begin{bmatrix} 0 & -h & 0 \end{bmatrix}^T \quad (6)$$

where h is the altitude of the aircraft above the ground. Therefore, a point on the horizon may be expressed as:

$$\mathbf{d}^w = \mathbf{x}^w + \mathbf{r}^w \quad (7)$$

where $\mathbf{x}^w$ is the position on the horizon, relative to the camera, expressed in the world frame. Expressing this in the camera frame:

$$\mathbf{x}^c = \mathbf{R}_c^w \mathbf{x}^c \tag{8}$$

And therefore the point on the horizon can be described as:

$$\mathbf{d}^w = \mathbf{R}_c^w \mathbf{x}^c + \mathbf{r}^w \tag{9}$$

The horizon, when projected onto the image plane of the camera, can be described by a point $\mathbf{u}^c$ and a direction vector $\mathbf{m}^c$:

$$\mathbf{m}^c = \begin{bmatrix} m_x & m_y & 0 \end{bmatrix}^T \mathbf{u}^c = \begin{bmatrix} u & v & f \end{bmatrix}^T \tag{10}$$

where u and v are the x- and y-coordinates (in distance units) along the image plane, f is the focal length of the camera and m_y/m_x describes the gradient of the line.

Since the position of the horizon, $\mathbf{d}^w$ lies on the surface of the ground plane, it is perpendicular to the normal vector of the plane. Hence we can state that:

$$\mathbf{n}^w \bullet \mathbf{d}^w = 0 \tag{11}$$

Substituting for $\mathbf{d}^w$ yields:

$$\mathbf{n}^w \bullet (\mathbf{R}_c^w \mathbf{x}^c + \mathbf{r}^w) = 0 \tag{12}$$

The direction vector of the horizon line $\mathbf{l}^w$ lies on the plane and is therefore also orthogonal to the normal vector.

$$\mathbf{n}^w \bullet \mathbf{l}^w = \mathbf{n}^w \bullet (\mathbf{R}_c^w \mathbf{l}^c) = 0 \tag{13}$$

Equations 12 and 13 are in a form known as the *line-plane correspondence problem* proposed by Chen in [5], where the problem is to determine the rotation matrix from several corresponding lines and planes.

In order to solve the dot product equations for pitch θ and roll ϕ, an expression must firstly be developed for $\mathbf{l}^C$. To achieve this, we denote two particular points on the horizon line, $\mathbf{d}_0$ and $\mathbf{d}_1$ as:

$$\mathbf{d_0}^w = \begin{bmatrix} x_0 & 0 & d \end{bmatrix}^T \tag{14}$$

$$\mathbf{d_1}^w = \begin{bmatrix} x_1 & 0 & d \end{bmatrix}^T \tag{15}$$

With the difference between the two points being a scalar multiple of the horizon line vector:

$$\mathbf{l}^w = k\left(\mathbf{d}_0^w - \mathbf{d}_1^w\right) \tag{16}$$

Therefore, the positions of points $\mathbf{d}_0$ and $\mathbf{d}_1$ relative to the camera are:

$$\mathbf{x_0}^w = \mathbf{d_0}^w - \mathbf{r}^w \tag{17}$$

$$\mathbf{x_1}^w = \mathbf{d_1}^w - \mathbf{r}^w \tag{18}$$

Introducing the equations for a calibrated perspective camera:

$$\mathbf{u} = \begin{bmatrix} u \\ v \\ f \end{bmatrix} = \frac{f}{Z} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{f}{Z} \mathbf{x}^c \tag{19}$$

where f is focal length of the camera, u and v are the x- and y-coordinates respectively of the position of a point on the image plane and X, Y and Z are the coordinates of the observed point. Rearranging gives:

$$\mathbf{x}^c = \frac{Z}{f} \mathbf{u} \tag{20}$$

Substituting into the equation for the horizon line yields:

$$\mathbf{l}^W = k(\mathbf{x_0}^w - \mathbf{x_1}^w) = k\mathbf{R}_c^W \left(\frac{Z}{f} (\mathbf{u_0} - \mathbf{u_1}) \right) \tag{21}$$

The difference between the two horizon points projected onto the image plane is a scalar multiple of direction vector, describing the line on the image plane.

$$k_2 \cdot \mathbf{m}^c = (\mathbf{u_0} - \mathbf{u_1}) \tag{22}$$

Hence the horizon line and the projection onto the image plane can be related by:

$$\mathbf{l}^W = k k_2 \frac{Z}{f} \mathbf{R}_c^W \mathbf{m}^c \tag{23}$$

Substituting into the dot product:

$$\mathbf{n}^w \bullet (k \cdot k_2 \frac{Z}{f} \mathbf{R}_c^W \mathbf{m}^c) = 0 \tag{24}$$

Since $k \cdot k_2 Z/f$ is scalar, then we may write:

$$\mathbf{n}^w \bullet (\mathbf{R}_c^W \mathbf{m}^c) = 0 \tag{25}$$

Directly substituting for $\mathbf{n}$, $\mathbf{R}_c^W$ and $\mathbf{m}^c$ yields:

$$-1 cos(\theta) \left(m_x \sin(\phi) + m_y \cos(\phi) \right) = 0 \tag{26}$$

Assuming $\cos(\theta)$ is not zero, solving for ϕ gives:

$$\phi = atan \left(\frac{-m_y}{m_x} \right) \tag{27}$$

Which is the intuitive result that the roll angle is dependant on the gradient of the horizon line on the image plane. Recalling the other dot product:

$$\mathbf{n}^w \bullet (\mathbf{R}_c^w \mathbf{x}^c + \mathbf{r}^w) = 0 \tag{28}$$

And substituting the projective camera equations,

$$\mathbf{n}^w \bullet (\mathbf{R}_c^w \frac{Z}{f} \mathbf{u} + \mathbf{r}^w) = 0 \tag{29}$$

In order to substitute this equation, an expression for Z must be found. After rearranging the relative position of the camera and a point on the horizon:

$$\mathbf{x}^c = \mathbf{R}_w^c(\mathbf{d}^w - \mathbf{r}^w) \tag{30}$$

Directly substituting for $\mathbf{R}_w^c$, $\mathbf{d}^w$ and $\mathbf{r}^w$ results in the third index of the $\mathbf{x_3}^c$ (i.e. the Z component) being:

$$Z = d\cos(\theta) - h\sin(\theta) \tag{31}$$

Substituting and solving for θ yields:

$$\theta = arcrtan\left(\pm\frac{-hf + ud\sin(\phi) + vd\cos(\phi)}{df + uh\sin(\phi) + vh\cos(\phi)}\right) \tag{32}$$

If the distance to the horizon is much greater than the height of the aircraft (that is, $d >> h$) then the expression for the pitch angle reduces to:

$$\theta = arctan\left(\pm\frac{u\sin(\phi) + v\cos(\phi)}{f}\right) \tag{33}$$

Which is dependant on the roll angle and the position on the image plane that the horizon falls. For the zero roll case, this expression is easily verified using simple trigonometry.

2.2. Deriving the Body Rates from Optical Flow on the Horizon Line

The horizon line is by no means the only information available from an image sequence. Optical flow has been used in a vast array of applications from Structure from Motion [12] through to an autonomous landing of a UAV [9].

We use the classic optical flow equations for rigid body motion [1] in order to take advantage of the decoupling between the translational and rotational velocities:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \frac{f}{Z}\begin{bmatrix} 1 & 0 & \frac{-u}{f} \\ 0 & 1 & \frac{-v}{f} \end{bmatrix}\begin{bmatrix} \dot{r_x}^w \\ \dot{r_y}^w \\ \dot{r_z}^w \end{bmatrix} + \begin{bmatrix} \frac{uv}{f} & -(f+\frac{u^2}{f}) & v \\ (f+\frac{v^2}{f}) & \frac{-uv}{f} & -u \end{bmatrix}\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{34}$$

If the point observed in the image lies on the horizon, then the distance along the focal axis is very large and hence the contribution from the translational velocity is negligible. Therefore, it can be assumed that on the horizon, the optical flow is influenced only by body rotations:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} \frac{uv}{f} & -(f+\frac{u^2}{f}) & v \\ (f+\frac{v^2}{f}) & \frac{-uv}{f} & -u \end{bmatrix}\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{35}$$

In Figure 5, the influence of the translational motion on the optical flow is clearly evident when comparing the optical flow measured on the horizon (Figure 5a) and the optical flow for on the false horizon on the edge of the road (Figure 5c). On the latter, the translational motion of the aircraft results in the road moving "downwards" in the image and is easily detected by the statistical test presented in the following section.

Figure 5. Optical flow at candidate horizons selected by the Hough Transform

3 Estimation, Temporal Tracking and Data Association

As was previously shown, the image processing front-end can be susceptible to falsely detecting the horizon if a strong line or edge is detected in an image. Consequently, additional information must be utilised to distinguish the true horizon in the image apart from the spurious response that may occur from time to time.

The solution we propose is not to simply choose a single candidate at each frame, but to track candidates over time and to statistically choose the most likely candidate, based on a motion model of the camera.

3.1 The Attitude Estimation Filter

We choose the state vector of the aircraft to consist of the roll angle ϕ, pitch angle θ and body rates of the aircraft. We assume the motion model to be a non-linear Markov process, perturbed by uncorrelated zero-mean Gaussian noise. Similarly, we assume that the measurements are a non-linear function of the state, corrupted by uncorrelated zero-mean Gaussian noise. That is:

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{n}(k)) \tag{36}$$

$$\mathbf{z}(k) = \mathbf{h}(\mathbf{x}(k), \mathbf{w}(k)) \tag{37}$$

Where $\mathbf{x}(k)$ is the state vector of the vehicle and $\mathbf{n}(k)$ is an uncorrelated zero-mean Gaussian random vector

with diagonal covariance matrix $\mathbf{Q}$, $\mathbf{z}(k)$ is the measurement vector at time k and $\mathbf{w}$ is a zero-mean Gaussian noise vector with a diagonal covariance matrix $\mathbf{R}$.

The non-linear process model we employ is a simple constant velocity model perturbed by random accelerations. If we assume that the body rates are approximately constant over the sampling interval Δt, then the state transition equations are:

$$\begin{bmatrix} \phi(k+1) \\ \theta(k+1) \\ \omega_x(k+1) \\ \omega_y(k+1) \\ \omega_z(k+1) \end{bmatrix} = \begin{bmatrix} \phi(k) + \Delta t \left(\dot{\phi}(k) \right) \\ \theta(k) + \Delta t \left(\dot{\theta}(k) \right) \\ \omega_x(k) \\ \omega_y(k) \\ \omega_z(k) \end{bmatrix} + \begin{bmatrix} n_\phi \\ n_\theta \\ n_{\omega_x} \\ n_{\omega_y} \\ n_{\omega_z} \end{bmatrix} \tag{38}$$

Where:

$$\dot{\phi}(k) = (\omega_x(k) \sin(\phi(k)) \tan(\theta(k)) + \omega_y(k) \cos(\phi(k)) \tan(\theta(k)) + \omega_z(k))$$

and

$$\dot{\theta}(k) = \omega_x(k) \cos(\phi(k)) - \omega_y(k) \sin(\phi(k))$$

The measurement equations consist of the direct observations of the pitch and roll from the horizon, and i optical flow observations on the horizon. Therefore, the measurement vector $\mathbf{z}(k)$ is of length $2(i+1)$ and is related to the states via the linear equations:

$$\begin{bmatrix} \phi \\ \theta \\ \dot{u}_i \\ \dot{v}_i \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{u_i v_i}{f} & -(f + \frac{u_i^2}{f}) & v_i \\ 0 & 0 & (f + \frac{v_i^2}{f}) & \frac{-u_i v_i}{f} & -u_i \end{bmatrix} \begin{bmatrix} \phi \\ \theta \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{39}$$

Since the process model is non-linear, we use the Extended Kalman Filter (EKF) as the estimator. The EKF requires the Jacobian of the process model, which is readily calculated using a software package such as the symbolic toolbox in MATLAB. The prediction and update equations follow the well-known Joseph form of the EKF [4].

3.2 Data Association

Data association between the state filter and the measurements is performed in innovation space by using the normalised innovation square [3] of the innovation vector, defined as:

$$\gamma = \mathbf{v}^T \mathbf{S}^{-1} \mathbf{v} \tag{40}$$

Where $\mathbf{v}$ is the innovation vector and $\mathbf{S}$ is the innovation covariance matrix. If the innovation vector is white and Gaussian, then γ is a chi-squared random variable with degrees of freedom equal to the length of $\mathbf{v}$. Hence, if there is unmodelled motion contained in the observations (e.g. a translational component), then the resulting test statistic will be large.

Referring back to Figure 5(a), the 24 optical flow vectors result in an innovation vector of length 50, and a of 28.16, providing overwhelming evidence to accept the null hypothesis that the line lies on the horizon ($p = 0.9946$). Figure 5(b), with a test statistic of 68.44 and 52 degrees of freedom (DoF) provides some evidence to reject the null hypothesis ($p = 0.0628$). Figure 5(c), where the translation motion is clearly evident, has a test statistic of 464.84 with 56 DoF resulting with $p = 0$.

4 Implementation and Results

The algorithm described in this paper has been implemented in MATLAB and C. The image precessing front-end had been implemented using the OpenCV libraries and is capable of processing webcam-sized images in real time at 15Hz.

4.1 The Image Processing Front-End

The datasets used to evaluate the algorithm are from the University of Florida [7] (webcam data over a 2.4GHz link), data collected during flights out of Archerfield Airport (Point Gray Flea), and static image data collected at Mary Cairne Cross, Queensland (Point Gray Dragonfly)[1]. Data was collected at different times of day and under different lighting and weather conditions.

Flight tests at Archerfield, Queensland have been processed and appear to be relatively robust to changing light, scenery and glare. There are, however, noticeable failures where there is a strong line, such as the dirt runway seen in Figure 3. Failures also occur where the blue/green haze over the horizon cause a number of spurious edges to be detected, which are sometimes identified as candidate lines.

Overcast conditions in Figure 6 were captured at Mary Cairn Cross, Queensland , where the front-end successfully handled varying lighting conditions throughout the day.

The footage in Figure 7 was taken from a horizon-detection stabilized flight at the University of Florida using an MAV. Despite the extreme degradation caused

[1]Dataset courtesy of Ryan Carnie

Figure 6. Front-end processing of the Mary Cairn Cross dataset

by the video link and poor quality of the camera, the horizon detection algorithm is able to successfully detect the horizon in the vast majority of the frames. Failures occur where the horizon is heavily distorted by the optics, in the presence of severe noise or when the horizon is not present in the image.

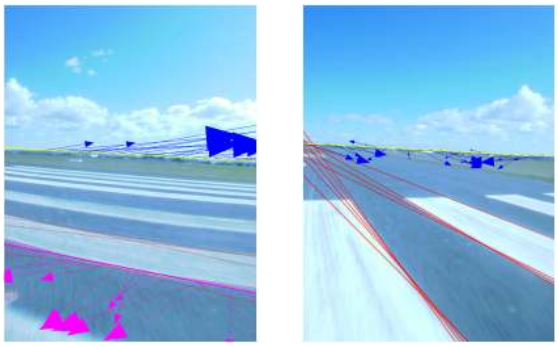
Figure 7. Front-end processing of the UFL Dataset

4.2 The Use of Temporal Tracking

There are a number of instances where the image processing front-end failed because of the strong response of spurious edges. Because the camera parameters are approximately known in the Archerfield dataset, image sequences known to cause spurious responses were re-processed using the additional temporal data association and tracking described in this paper.

To demonstrate the use of the temporal motion residual method of selecting the horizon, a dataset of the aircraft turning onto the runway and taking off. This dataset has been deliberately chosen as a test case because it has been previously shown that the image processing front-end can fail when other strong edges are present in the scene. In this case, the runway threshold markings are often dominant in the image, creating many responses from the front-end.

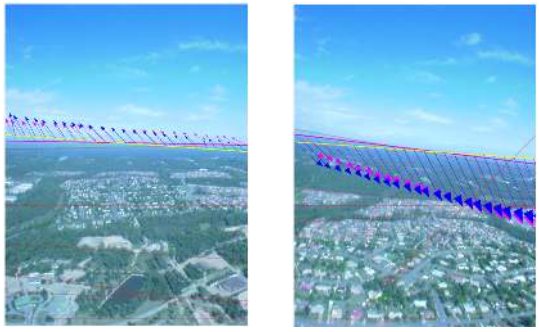
For the take-off dataset, the advantage of using the residuals of the motion is clearly evident in the reduction of cases of false detection due the front-end detecting another strong line in the image. sing the front-end only, 90.7% of the frames had the horizon detected



(a) (b)

Figure 8. Horizon detection algorithm based on both temporal tracking and the image processing front-end. Yellow denotes the output of the motion algorithm

correctly. With the use of temporal tracking, this increased to 98.8%. A typical example of this is show in Figure 8. The horizon which was detected based on motion residuals is shown in yellow, with blue associated optical flow vectors. The pink line shows the strongest response from the image processing front-end with its associated flow vectors. The faint red lines are other horizon candidates that have been dismissed by the algorithm. Despite the relatively poor optical flow on the horizon, the motion residuals algorithm is sufficiently robust against responses away from the horizon.



(a) (b)

Figure 9. Similar, but not identical responses from the front-end and motion model

Where there are other candidates lines close the horizon, the performance of the algorithm is mixed. A second dataset of the aircraft in straight and level flight has been analysed on a day where the horizon is partially obscured by haze. In this instance, multiple candidates close to the horizon are detected, and

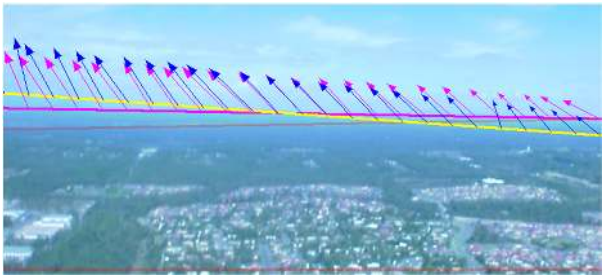


Figure 10. Closeup of the horizon

therefore have similar optical flow vectors. As a result, although the strongest front-end response correctly detected the horizon more than 90% of the time, the motion residuals did not associate the correct candiate in 41% of those cases.

On the surface, the latter results appear not to bode well for the motion residuals. In all cases where the front end resulted in a "correct" detection, the motion residuals responded to a line that was also near the horizon and will therefore result in only a small attitude error. A typical case where this occurs in shown in Figures 9 and 10

Current work is considering a multiple model approach to the problem where multiple horizon hypotheses are tracked through time, allowing information from two "close" candidates to evolve over time and allow better data association decisions to be made.

5 Conclusions

This paper has presented an algorithm for estimating the attitude of a fixed wing aircraft from the horizon using machine vision. The algorithm is based on detecting lines in an image which may correspond the horizon, followed by testing the optical flow against the measurements expected by the motion filter.

The results show that in scenarios where it is possible for the front-end to latch onto strong object other than the horizon, utilising motion properties is an effective means for the detection of the horizon.

However, there are still cases where motion properties have not been able to select the best candidate where other candidates are close to the horizon. Work is currently under way on a multiple mode approach where multiple horizon hypotheses are maintained over time instead of a single filter where only a single measurement is associated and assimilated into the filter.

Work is also ongoing on validating the estimation of the attitude from the aircraft, with preparation for flight trials with an attitude truth underway. Testing will be conducted over a range of conditions and scenes to examine the robustness of the entire algorithm to changes in lighting, weather and different cameras.

References

[1] G. Adiv. Inherent ambiguities in recovering 3-d motion and structure from a noisy flow field. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 11(5):477–489, 1989.

[2] G. Bao, Z. Zhou, S. Xiong, X. Lin, and X. Ye. Towards micro air vehicle flight autonomy research on the method of horizon extraction. In *Instrumentation and Measurement Technology Conference, 2003. IMTC '03. Proceedings of the 20th IEEE*, volume 2, pages 1387–1390 vol.2, 2003.

[3] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan. *Estimation with applications to tracking and navigation*. Wiley, New York, 2001.

[4] R. G. Brown and P. Y. C. Hwang. *Introduction to random signals and applied Kalman filtering : with MATLAB exercises and solutions*. Wiley, New York, 3rd edition, 1997.

[5] H. Chen. Pose determination from line-to-plane correspondences: existence condition and closed-form solutions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):530–541, 1991.

[6] D. Dusha, W. Boles, and R. Walker. Fixed-wing attitude estimation using computer vision based horizon detection. In *Proceedings 12th Australian International Aerospace Congress*, 2007.

[7] S. Ettinger, P. Ifju, and M. C. Nechyba. Vision-guided flight for micro air vehicles (mavs), 2005.

[8] R. C. Gonzalez and R. E. Woods. *Digital image processing*. Prentice Hall, Upper Saddle River, N.J., 2nd edition, 2002.

[9] M. V. Srinivasan, S. Zhang, and J. S. Chahl. Landing strategies in honeybees, and possible applications to autonomous airborne vehicles. *Biological Bulletin*, 200:215–221, 2001.

[10] S. Todorovic, M. Nechyba, and P. Ifju. Sky/ground modeling for autonomous mav flight. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 1, pages 1422–1427 vol.1, 2003.

[11] G.-S. Young and R. Chellappa. Statistical analysis of inherent ambiguities in recovering 3-d motion from a noisy flow field. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 14(10):995–1013, 1992.

[12] M. Zucchelli. *Optical Flow Based Structure from Motion*. Phd dissertation, Kungl Tekniska Hogskolan (Royal Institute of Techology, Stockholm), Stockholm, 2002.