Department of Electrical and Computer Systems Engineering

Technical Report MECSE-10-2006

Feature Based Navigation for UAVs

W.K. Kong, G.K. Egan and T. Cornall

MONASH UNIVERSITY

Feature Based Navigation for UAVs

W.Y. Kong, G.K. Egan and T. Cornall
Department of Electrical & Computer Systems Engineering
Monash University 3800
Melbourne, Australia

Abstract— **Most UAVs (Unmanned Aerial Vehicles) today use the GPS (Global Positioning System) for navigation. While GPS is an accurate and usually reliable aid to navigation, it can be unavailable for various reasons. The aim of this project is to develop an alternative navigation method, which can be used when GPS transmissions are not available.**

The method suggested here is called Feature Based Navigation. It works by comparing the current image taken by the on-board camera with previous reference images, usually the most recent, obtained when GPS was available. The system then compares locations of similar features extracted from the images using these to determine the location of UAV. We outline the results of the method using real data taken from the data log of a UAV in flight.

Index Terms—**UAV, GPS, navigation, features**

I. Introduction

Most UAVs in service use the GPS (Global Positioning System) as a primary mean of navigation. A GPS receiver relies on using information transmitted via radio frequency from at least three of the 24 satellites to calculate its location. Unfortunately the GPS transmissions may become unavailable for a number of natural and man made reasons. The term coined for this is "GPS denial" and because of the now widespread dependence upon GPS has become a serious research area. The system is itself owned by the USA, and until year 2000 only a less accurate version of the system was available to general public [1]. An undertaking has been given that the current level of accuracy will be maintained for public use.

The aim of this research was to develop a system capable of providing location updates for a UAV flying at a reasonable altitude when GPS transmissions are lost. For this preliminary study the system is to operate in feature rich terrain of artificial and natural features. This system is intended to operate on one of Monash University's UAVs with no additional equipment specific to the navigation task. It has as its input altitude, pitch, roll, yaw, airspeed and heading and video from the camera the latter which is tilted forward and downward. The power and payload budget of the UAV is severely constrained and so the intention was to develop a scheme amenable to implementation on the flight control computer possibly with additional FPGA-based image processing.

II. Background

There are a few alternative navigation methods to GPS that are available. While they are not designed for UAV navigation, some concepts could be adapted and used.

A. Alternative Satellite Navigation System

There are other global satellite navigation networks which are designed to perform similar task to GPS. These systems include the GLONASS developed by former U.S.S.R and the Galileo project which is currently being developed by ESA.

B. Celestial Navigation

Celestial navigation is based on locating celestial bodies. With an accurate time reference, and previously tabulated or computed orbits of the bodies it is possible to determine location using only addition and subtraction operations [2]. Unfortunately a camera trained on the sky is required.

C. Geomorphometric Navigation

This method basically involves using radar or laser scanning system to scan the ground for terrain features, which may include hills, vegetation and buildings [3]. Once the location of these features relative to the UAV are obtained a reference database is searched for a match which is then used to determine the UAV's absolute location. Similar techniques, used in submarines, are called bathymetric navigation [4].

D. Localization Using Model Image Correspondence

This method requires the capture and storage of a 3D digital model of the area to be navigated, for example a CAD model for indoor navigation or digital elevation image for the purpose of outdoor navigation. Features are extracted from images taken by on board camera and again a search is made of the model database permitting localisation of vehicle [5].

E. Feature Based Navigation

The technique proposed "Feature Based Navigation" is similar to the technique "Localisation Using Model Image Correspondence", but it does not require a database. The method works by comparing location of features on current image taken by the on-board camera with previous reference images, usually the most recent, obtained when the GPS was available. The initial position of the UAV is known so the images may be transformed into the same plane and the position of the UAV computed using the relative locations of identical features in both images. Essentially the reference image replaces the database and searching of the database is replaced by searching through the features extracted from the image extracted from reference image and the image taken by the onboard camera.

Feature based navigation was considered the most appropriate technique in this case, as it requires no additional equipment being implemented in software. Unlike Localisation Using Model Images Correspondence it does not require detailed modelling of the area over which the UAV is navigating. All that is required is a reference image in the form of an aerial photo (Figure 1) and the longitude and latitude location of the four corners all of which can be obtained in flight.

Figure 1 Representative aerial photo used

III. Reference Image Extraction

As part of the feature matching process, the system has to be able to generate a reference image of what the camera should be capturing when the UAV is flying at a certain location, altitude, attitude and assumed heading (there is no compass) over the terrain represented by the reference image.

The main problem with the extraction of reference image is how to account for the effect of attitude of the UAV on the image the on board camera will capture. As has already been stated the UAV's camera is pointing forward and downward. The aircraft itself is likely to be suffering significant pitch roll and yaw due to turbulence and the acts of turning and/or climbing to new destination points during its mission. Computing the reference image is therefore not as straightforward as it may appear.

A. Coordinate System Used

There are two coordinate systems used. The global coordinate system or the coordinate system used to represent location on the reference image; positive x is aligned with due north, positive z is due east and positive y is vertically up. Origin is the bottom left hand corner of the reference image.

In the local or body coordinate system of the UAV, positive x axis is towards the nose, positive z axis is towards the right wing, and positive y axis is towards the sky when the UAV is flying upright. The origin of this coordinate system is assumed to be at the location of the on board camera. The notation for orientation of UAV is as follow, positive pitch is nose up, positive roll is right wing down and positive yaw and heading is nose to the right.

B. Reference Image Extraction - Basic Concept

With most digital cameras the viewed area is approximately rectangular although there will be distortion at the corners of the image. The size of the viewing rectangle depends on two things; the distance between the image and the camera, and the viewing angles of the camera. An image taken by a digital camera also has a certain resolution, which determines the number of pixels vertically and horizontally on every image the camera takes. Using these two ideas and assuming pixels are equally spaced across an image it is possible to calculate the physical location of a pixel in an image when it is capturing an object at a certain distance away from the camera.

This method effectively gives a vector from the camera location (assumed to be at origin in the above case) to any pixel on the viewing rectangle of a camera at a certain distance. This vector can be transformed with the Euler angle transformation matrix ($R = R_X(\psi)R_Z(\varphi)R_Y(\theta)$, where R_X is the rotation along x axis, R_Y is rotation along y axis, R_Z is rotation along z axis, R is the final transformation matrix [6]. The vector can be used simply to calculate the point on the ground which this pixel will capture.

IV. Feature Extraction and Matching

In this project it was decided to use edges as the features to be identified and matched. As the reference image may have been taken several months or years prior to the actual flight the lighting conditions will almost certainly be different. Thus the features selected should, as far as possible, be invariant under different lighting conditions.

A. Edge Detection

The edge detector chosen is called the 'Canny Edge Detector' [7].

The number of features to be matched was kept small to minimise mismatches. This was achieved by using an aggressive Gaussian filter in combination with low threshold values of T1 = 0.05 and T2 = 0.1. This set of threshold values are rather low, but the Gaussian has already suppressed most of the unwanted features leaving fragmentation of edges to a minimum [8].

B. Feature Matching

In the design of the feature matching module the following factors were considered:

- Sensor data samples are available every 0.2 Seconds
- The cruise speed of the UAV is approximately 20m/s, so the distance travelled by UAV between 2 log entries is approximately 5 metres
- Changes in roll, pitch, yaw and heading between 2 sensor readings are generally no more than 1-2 degrees

The GPS position update rate is actually every second however the autopilot usually performs simple extrapolation to provide estimated positions every 0.2 seconds.

It can be assumed that features captured by the UAV should also have moved and distorted by a small amount between two sensor readings.

The algorithm attempts to find features on the reference and camera images which have similar physical properties. Properties such as the length of the edge, the location for the centroid of the edge in the image, and Euler number of the area encircled by an edge, are all taken into account. A feature on one image is considered to have a match on the camera image if there is a feature which has similar properties, but not necessarily identical, to the properties of the feature on the reference image.

V. Calculating the UAV's Location

This process uses the location of centroid for a feature which is known on both images, to calculate a translation vector for the location of the UAV.

The first step is to establish where the centroid of the feature is on the world coordinate system. This step is relatively easy as the pixel location for centroid of the feature in the reference image is known, and transformation and translation matrix which are used to extract this image are also known. The location of the feature can be evaluated by linear equation.

The next step involves calculating the location of UAV based on the location of the feature in the camera image. As physical location for the centroid of this feature in global coordinates is known from previous calculation, and from the flight log the transformation matrix *R* and translation vector *TL* can be calculated, this mean the viewing vector *v'* is also known. Then it becomes a straightforward linear equation to solve. This is shown in figure 2.

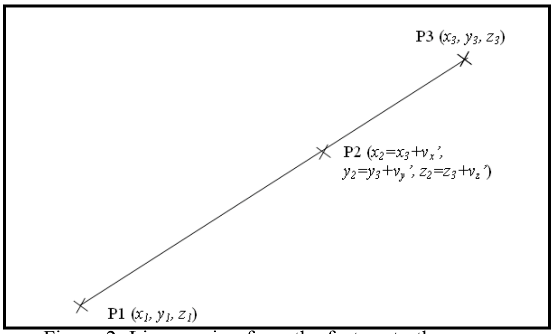

Figure 2: Line passing from the feature to the camera

The location of P1 is known based on calculation for the location of the feature in global coordinate system described above. The value of y_3 which is based on the altitude of the UAV is known, and because the view vector *v'* which is based on the location of the feature in the camera image and the transformation matrix *R* is known the value for v_x', v_y' and v_z' are also known. The aim is to solve for x_3 and z_3, which is the location of the camera.

With the values of x_3 and z_3 known, it is a simple matter of converting them to latitude and longitude.

Strictly only one feature is needed to calculate the value x_3 and z_3, but in the actual implementation the x_3 and z_3 for three matching features were calculated and averaged to improve the accuracy of the system.

$$m = \frac{y_3 - y_1}{y_3 + v_y'} = \frac{y_3}{y_3 + v_y'}$$

$$z_3 = \frac{(mv_z' + z_1 - mz_1)}{(1-m)}$$

$$x_3 = \frac{(mv_x' + x_1 - mx_1)}{(1-m)}$$

VI. Result and Discussion

Two experiments were conducted to test the performance of this system. The first involves using the system to calculate the location of the UAV in next time step for 125 entries which are recorded in the flight log. These 125 entries represent 25 seconds of flight time. This test is similar to an "open loop" test, aiming to see how with accurate information the system will be able to perform when different data is fed into it. The next test is a "closed loop" test and involves using the result of previous calculation as the input for the next calculation. This is what would happen when the GPS signal is lost; the input to the system will depend on the measurement taken by on board sensor as well as the location at the previous time step. Again this test was conducted for 125 entries of the flight log which represents 25 seconds of flight time. The flight log sequences used in both tests were identical.

The "open loop" test results for the system running every 1.0 second and 0.2 second demonstrated that the system with a few exceptions is capable of calculating the location of the UAV with errors of less then 25 metres for the test sequences. The system is able to on average calculate the location of the UAV with an error of 6.68 metres when it is operating with accurate information which is 0.2 second before the current point.


Figure 2 Experimental UAV with down and forward looking camera

There is always likely to be some error due to inaccuracy in the raw data such as the measurement of location by the GPS system, linear estimation of longitude and latitude location, and inaccuracy in measurement for the corners of the image. So an average error of 6.68 metres is quite acceptable as most commercial GPS systems in our experience, depending on aircraft velocity and the number of available satellites, have an uncertainty of 4-20 metres. The large errors in selected points are likely to be due to the simple feature matching algorithm used to determine if two features are matching. It seem that at some points there are too many features in the filtered images and it caused the matching algorithm to register two features which are mismatched to be matching features. This causes the system to treat the features as having shifted by a large amount thus causing an error in the calculation for the location of the UAV. It is shown that by reducing the time span between each calculation the average errors and the frequency of points with large errors decrease, which mean when the system is operating at 0.2 second per calculation the matching algorithm performs much better and frequency of such a mismatch decreases. This is because the features are no longer shifted as much and the stresses place on the matching algorithm have lessened, but still at some points the matching algorithm does fail, and this can only be fixed by implementing a more sophisticated matching algorithm.

The "closed loop" test results also show some promising results with the system being able to track the actual location of the UAV with an error of less then 20 metres for most of the duration of the test when it is operating at 0.2 second per calculation. The average error for the test is 7.75 metres. This amount of error is within the limits of the accuracy of the GPS unit. But what is worrying about the result is the fact that the error of the system seems to be increasing as more points are processed. This means in the current state this system is only valid as a navigation tool for about half a minute, beyond that it is unlikely that it will be able to provide accurate information about the location of the UAV.

To a certain degree this is to be expected, as each result will inevitably contain within it some error; using the previous result to calculate the location of the current time step is effectively like adding a random translation factor to the input thus putting stress on the feature matching algorithm. As the amount of input error accumulates the output also shows a corresponding increase in error, until total failure occurs.

A more sophisticated feature-matching algorithm is needed to resolve this problem. We can also obtain improvement by dynamically adjusting the threshold of the parameters for Canny algorithm. While the algorithm performs acceptably in identifying major edges and keeping them intact, there are still instances where edges are broken up in places where they should not be, and in some cases mismatching edges; this is most likely due to problems with threshold values selected. It may be possible to adjust the threshold based on the image content giving some improvement. We are continuing to explore parameters which will improve our initial results.

Currently each image registration and resulting localisation takes approximately 8 seconds on a contemporary PC workstation.

VII. FUTURE WORK

There are two major issues, which will need to be addressed before this system can be used for UAV navigation in real-time. These are improvements with the feature matching algorithm and the time needed to extract the reference image; possible solutions for these two issues are discussed briefly below.

A. Real time operation

The current algorithm involves repeated multiplication of a floating point vector to a floating point matrix, followed by an operation which involve a few multiplications, addition and division to solve a linear equation. These are all very repetitive operations, and can be optimised by converting them to integer operations and ideally implementing the whole operation in a FPGA.

B. Feature matching

Matching edges has limitations particularly with natural features. We could change the matching method from the idea that two features are matching if they pass a list of criteria, to deeming them to match if they exceed some threshold in the number of matches found along the candidate edges.

Once these two improvements are made to the system, the method of Feature Based Navigation as discussed in this paper should be viable for navigation of a UAV under GPS denial.

VIII. CONCLUSION

The aim of this research was to provide an alternative navigation method to a UAV when GPS transmissions are not available. A number of alternative methods were explored which took into account the limited computational resources onboard a small UAV. A method called Feature Based Navigation was devised and tested. The method works by comparing location of features on two images, a camera image taken by the on board camera of the UAV during flight, and a reference image of what the UAV should currently "see". The location of the UAV is calculated based on locations of similar features that are in the two images.

As seen from this paper, Feature Based Navigation is intended to provide permit navigation under GPS denial. Currently the system can successfully track the location of the UAV in close loop for at least 20 seconds without significant errors.

ACKNOWLEDGMENT

We wish to thank the Members of the Aerobotics Group at Monash University [9] and in particular Mr. Raymond Cooper the Chief Test Pilot for his assistance in obtaining log files and in practical considerations relating to UAV flight.

REFERENCES

[1] Andrzej Michalski, *The Accuracy of Global Positioning Systems*, IEEE Instrumentation & Measurement Magazine, March 2004, p. 56 – 61

[2] *Celestial Navigation*, http://en.wikipedia.org/wiki/Celestial_navigation, 2005

[3] Andy Shaw, Dave Barnes and Phil Summers, *Landmark recognition for localization and navigation of aerial vehicles,* International Conference on Intelligent Robots and Systems 2003, 2003

[4] F. Pappalardi, S.J. Dunham, M.E. LeBlang, T.E. Jones, J. Bangert and Kaplan, *Alternative to GPS*, Proceeding of OCEANS 97 Conference, 1997.

[5] Raj Talluri, *Mobile Robot Self-Location Using Model-Image Feature Correspondence*, IEEE Transactions on Robotics and Automation, 1996

[6] Martin John Baker, *Maths - Euler Angles*, http://www.euclideanspace.com/maths/geometry/rotations/euler/, 2005

[7] Bob Fisher, Simon Perkins, Ashley Walker and Erik Wolfart, *Feature Detector - Canny Edge Detector,* http://www.cee.hw.ac.uk/hipr/html/canny.html, 1994

[8] Bob Fisher, Simon Perkins, Ashley Walker and Erik Wolfart, *Spatial Filter – Gaussian Smoothing,,* http://www.cee.hw.ac.uk/hipr/html/gsmooth.html, 1994

[9] *Monash Aerobotics*, [online], Available: http://www.ctie.monash.edu.au/hargrave/aerobotics.html, 2002, (Accessed February 2006).