Airborne Vision System for the Detection of Moving Objects

Reuben Strydom, Saul Thurrowgood and Mandyam V. Srinivasan

The University of Queensland, Brisbane, Australia

Abstract

This paper describes a vision-based technique for the detection of moving objects by a moving airborne vehicle. The technique, which is based on measurement of optic flow, computes the egomotion of the aircraft based on the pattern of optic flow in a panoramic image, then determines the component of this optic flow pattern that is generated by the aircraft's translation, and finally detects the moving object by determining whether the direction of the flow generated by the object is different from that expected for translation in a stationary environment. The performance of the technique is evaluated in field tests using an aerial rotorcraft.

1 Introduction

Endowing a moving aerial platform with the capacity to detect and track moving objects can significantly improve the understanding of the surrounding environment and allow for greater control such as timely and effective evasive manoeuvres, thus improving overall safety. Traditionally, active sensors such as radar have been used to detect the motion of objects in the environment. However, such systems are bulky, expensive and difficult to miniaturise. Vision provides an inexpensive, compact and lightweight alternative. Indeed, experiments with flying and walking insects suggest a number of computationally simple, vision-based strategies for the detection of moving objects by a moving agent [Srinivasan and Davey, 1995]; [Mizutani et al., 2003]; [Zabala et al., 2012].

The detection of a moving object is a relatively simple task if the vision system is stationary. In such a situation the moving object can be detected simply by subtracting successive frames. Since the background is stationary, the subtraction will register changes in pixel intensity that are generated by the moving object, thus enabling estimation of its location. A related approach, known as background removal, involves constructing a background image over several frames, and subtracting this image from the current frame to locate the object [Elhabian et al., 2008]; [Chien et al., 2002]; However, the above methods will not work when the vision system is on a moving platform, because in this situation the image of the background will also be in motion.

Techniques for detecting moving objects in the presence of moving backgrounds have been developed. One method, developed by [Bouthemy et al., 1999] uses a colour-based criterion to improve the localisation of motion boundaries. A related approach integrates colour information over a time window and uses a model based analysis to detect and track moving objects [Cornelis et al., 2008]. While these methods have proven to be effective under certain conditions, they can be compromised by changes in hue due to variations in lighting conditions.

A well-known approach for detecting moving objects is Expectation Maximization (EM), which is an iterative method to determine the maximum *a posteriori* values of motion-characterising parameters in a model.

One of these approaches developed an algorithm to detect moving objects by analysing the optic flow of a video stream obtained by a moving airborne platform, using the assumption that the scene can be described by background and occlusion layers, estimated within the EM framework [Yalcin et al., 2005]. Another EM algorithm segments the optic flow into independent motion models and estimates the model parameters to compute and articulate motion [Rowley and Rehg, 1997]. Nevertheless, this method may not have the required accuracy to detect the motion of smaller objects due to the inaccuracy in the estimations within the EM framework.

Another widely used approach for detecting moving objects involves the use of the optic flow map itself, which is the projection of object velocities onto the image plane [Barron and Beauchemin, 1995]. [Kobayashi and Yanagida, 1995] use a scheme that detects moving objects in terms of a drop in coherence of the optic flow (the image of the object moves in a direction different from that of the background). This approach can be limited by the generally poor signal-to-noise ratio of optic flow patterns. [Yamaguchi et al., 2006] present a technique that uses optic flow to estimate egomotion, and then uses feature points in three-dimensional space to determine points where the epipolar lines have a negative distance, which represents moving objects or false correspondence points.

In reality, algorithms that rely solely on optic flow information tend to be limited in their accuracy due to the poor signal-to-noise ratios that are usually associated with flow maps, which can lead to misclassification of object motion.

In this paper, we present a robust technique for the detection of moving objects by combining an algorithm for object detection with an algorithm for detecting any motions of the object that are inconsistent with the measured egomotion. We evaluate the performance of this approach through field tests using an airborne vehicle. An overview of the object motion detection algorithm is shown in Figure 1.

Authors are associated with the Queensland Brain Institute (QBI) and the School of Information Technology and Electrical Engineering (ITEE), University of Queensland, St Lucia, QLD 4072, Australia, and the ARC Centre for Excellence in Vision (email: reuben.strydom@uqconnect.edu.au).

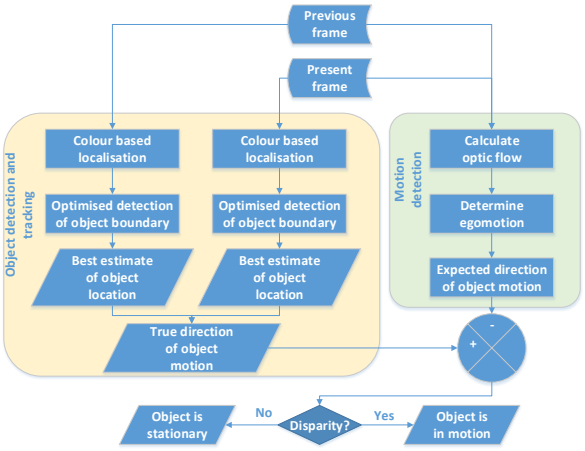

Figure 1. Overview of algorithm.

2 Object detection

As we mention above, the reliability with which one can determine whether an object is moving can be enhanced if the object can be detected consistently, i.e. segmented from the background, so that its location in the image can be determined accurately and reliably. In our case the object is a red ball, 55cm in diameter, viewed against a background that consists of a green football field when the ball is on the ground, or against trees or the sky when it is in the air. Detection of the ball involves two steps: (i) A colour filtering algorithm is used to distinguish the red pixels of the ball from pixels of other colours that represent the background (e.g. ground, trees or sky). This segmentation provides an approximate location of the ball in the image. (ii) The absolute gradient of the image is computed, as illustrated in Figure 2. The boundary of the ball is then detected by summing the pixel intensities in the gradient image along the boundary of a circle. The radius, and the *x* and *y* positions of this circle in the image are optimised to obtain the maximum summed value. This optimization is performed on the view sphere, where the projection of the ball is always a circle, rather than in image space where the projected image can appear elliptical and even non-convex. This circle represents a 'best estimate' of the position and diameter of the image of the ball.

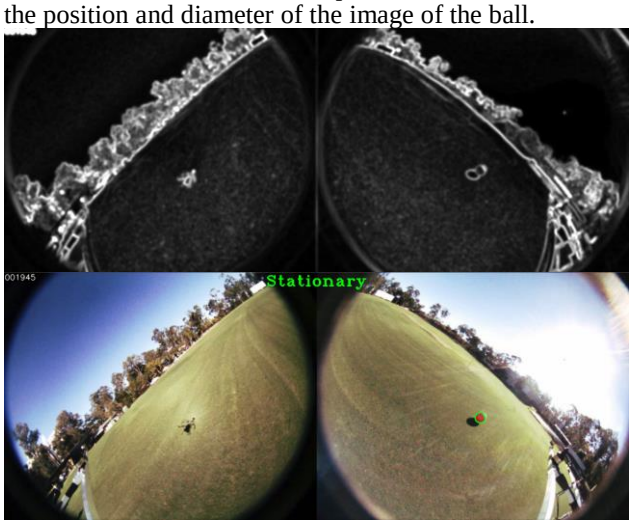

Figure 2. Illustration of technique for determining the centre of the object. Top: Gradient image computed over all channels of the RGB image to determine the radius and the center coordinates of the image of the ball. Bottom: Image displaying the estimated fit from the gradient image, shown by green circle around red ball.

3 Detection of object motion

If the motion of the aircraft is known (or can be inferred), the motion of an object in the environment can be detected by determining whether the image of the object moves in a direction that is different from that expected if the object were stationary [Irani and Anandan 1998]; [Srinivasan, 2011]. In principle, the motion of the aircraft (egomotion) could be inferred by using information from its on-board IMUs. However, IMU signals can be noisy and lead to unreliable results. In this study we present a solution that uses purely visual information.

Here we infer egomotion by computing the pattern of optic flow that is generated in a panoramic image of the environment, captured by the on-board vision system. The procedure for determining whether an object is moving consists of the following four steps:

1. Compute the optic flow in the panoramic image.
2. Determine egomotion from the measured pattern of optic flow.
3. Obtain the pattern of optic flow corresponding to the translatory component of the egomotion (the translatory flow field) for a stationary environment. This will be a pattern of vectors that are directed radially outwards from a point that represents the 'pole' of the flow field, and corresponds to the direction of the translatory component of the aircraft's egomotion (black arrows in Figure 3)
4. Determine whether the object is moving by comparing the direction of motion of the object (red arrow, Figure 3) with the expected pattern of optic flow. A discrepancy between these directions indicates that the object is in motion. Note that the object can move in three dimensions: its motion does not need to be restricted to a plane.

3.1 Computation of optic flow

Optic flow vectors are computed in the panoramic image at the video frame rate (25 Hz) in an evenly sampled grid of approximately 400 points using a pyramidal block-matching method. Image matching is performed using a sum of absolute difference on 7x7 image blocks, searching a 7x7 area, with sub-pixel refinement accomplished through equiangular fitting [Shimizu and Okutomi, 2003]. Note that this is computed over a pyramid of images, which indicates a larger effective search range.

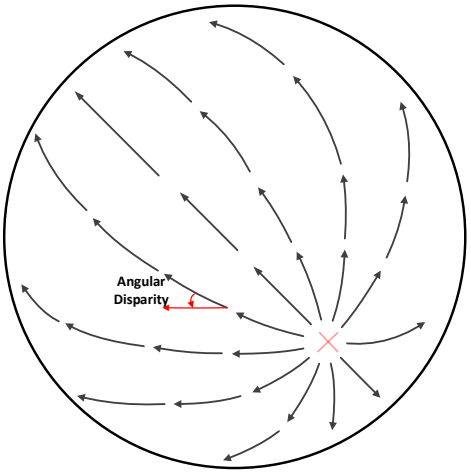

Figure 3. Optic flow vector field.

3.2 Determination of egomotion

We assume that, (i) the object in question occupies a small fraction of the panoramic field of the vision system, so that the computation of egomotion is not significantly affected by the possible motion of the object. (ii) The aircraft is moving over a plane, i.e. the environment in the vicinity of the aircraft is approximately flat.

The egomotion is computed by using an iterative algorithm to derive the best-fitting translational (**T**) and rotational (**Ω**) vectors that describe the aircraft's motion between successive video frames.

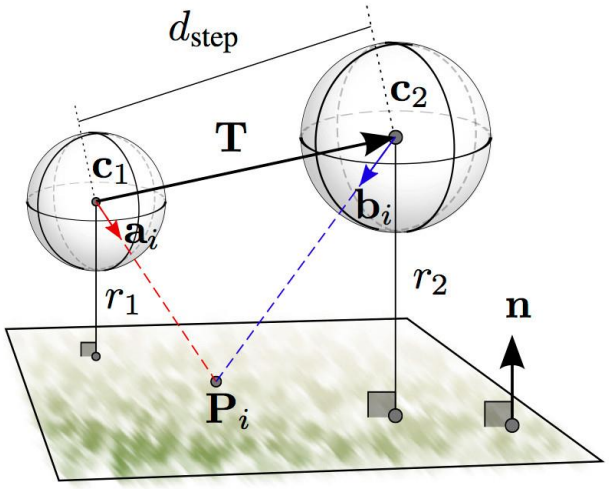

Figure 4 Measurement of egomotion (translation and rotation) of view sphere over a plane. The sketch illustrates movement of the vision system over a plane, translating over a distance **T** from position $\mathbf{c}_1$ to position $\mathbf{c}_2$. The unit vectors directed at a point $\mathbf{P}_i$ on the ground plane from positions $\mathbf{c}_1$ and $\mathbf{c}_2$ are respectively $\boldsymbol{a}_i$ and $\mathbf{b}_i$. Details in text.

The procedure for computing egomotion from optic flow is as follows. As illustrated in Figure 4, we wish to estimate the motion of the vision system between the two positions $\mathbf{c}_1$ and $\mathbf{c}_2$, as described by a translation vector (**T**), and a rotation vector (**Ω**). Consider a set of points $\mathbf{P}_i$ on the ground plane, which have unit view vectors $\boldsymbol{a}_i$ when the vision system is at $\mathbf{c}_1$, and unit view vectors $\mathbf{b}_i$ when the vision system is at $\mathbf{c}_2$. The vectors $\mathbf{b}_i$ are obtained by measuring the optic flow that is generated by the points $\mathbf{P}_i$ when the vision system moves from $\mathbf{c}_1$ to $\mathbf{c}_2$. **n** is the unit vector defining the normal to the ground plane, in the direction facing the vision system, in the rotational frame at position $\mathbf{c}_2$.

Consider the unit vector $\boldsymbol{a}_i$', which represents the unit vector $\boldsymbol{a}_i$ rotated through an angle **Ω** is given by

$$\boldsymbol{a}_i' = rodrigues(-\boldsymbol{\Omega})\boldsymbol{a}_i \qquad (1)$$

where *rodrigues* is Rodrigues' rotation formula that converts **Ω** into a 3x3 rotation matrix.

For any given translation (**T**) and rotation (**Ω**) we can then estimate the unit view vectors $\mathbf{s}_i$ at the location $\mathbf{c}_2$ to be $\mathbf{s}_i = \boldsymbol{a}_i'$ when $\boldsymbol{a}_i'$ is directed toward the sky (this would be a point that is effectively infinitely far away, and is therefore unaffected by translation);

and

$$\boldsymbol{s}_i = \frac{\boldsymbol{a}_i'}{\boldsymbol{a}_i'.(-\boldsymbol{n})} - \boldsymbol{T} \qquad (2)$$

when $\boldsymbol{a}_i'$ is directed toward the ground plane (this would be a point that is a finite distance away, and is affected by the translation). The term $\boldsymbol{a}_i'\cdot(-\mathbf{n})$ in the denominator is a factor that normalizes the visual motion of the point with reference to a ground plane that is at a unit distance from the vision system.

The aircraft's motion is then computed from the pairs of unit vectors ($\boldsymbol{a}_i$, $\mathbf{b}_i$) by finding the best rotation (**Ω**) and the best translation (**T**) that minimise the objective function

$$\frac{\arg min}{\boldsymbol{\Omega},\boldsymbol{T}} \sum_{i=1}^{n} \left\| \frac{\boldsymbol{s}_i}{\|\boldsymbol{s}_i\|} - \boldsymbol{b}_i \right\| \qquad (3)$$

The optimization is implemented by direct iterative search of the 6-DOF parameter space, using the derivative free gradient descent algorithm, BOBYQA [Powell, 2009], as implemented in the NLopt library [Johnson, 2010]. This iterative algorithm operates at the video frame rate of 25/sec.

The approach described here is analogous to that used by [Corke et al., 2004] for visual odometry in terrestrial vehicles, but is extended here to 3-D motion with 6 degrees of freedom.

3.3 Computing the translatory component of egomotion

The egomotion algorithm described in section 3.2 above delivers the motion of the aircraft in 6 degrees of freedom, in the form of a three-dimensional rotation vector (**Ω**) and a three-dimensional translational vector (**T**). The translational vector specifies the direction and magnitude of the translational motion of the aircraft between successive frames. The estimated translational magnitude corresponds to having an altitude of one unit at position $\mathbf{c}_1$.

3.4 Detection of object motion

As explained above, the object is deemed to be moving if the direction of its motion in the image is different from that corresponding to the computed translatory flow field for a stationary environment. This computation is carried out as follows.

The position of the object in the image is first transformed into a vector that describes its location in the 3D view sphere attached to the aircraft body frame. The vector defining the two dimensional component of the normalised vector **o** in the plane perpendicular to **T** is defined as:

$$\boldsymbol{o}' = \boldsymbol{o} - \boldsymbol{t}(\boldsymbol{t}\cdot\boldsymbol{o}) \qquad (4)$$

Where **o** is the normalised vector of **O** and **t** is the normalised vector of **T**.

Once the projected location of the object is determined, the direction of its motion (θ) is given by the inverse cosine of the new object vector (**o'**) projected onto the **Up** vector, as shown in Figure 5 and described by the equation:

$$\boldsymbol{\theta} = \boldsymbol{cos}^{-1}(\boldsymbol{o}'\cdot\boldsymbol{Up}) \qquad (5)$$

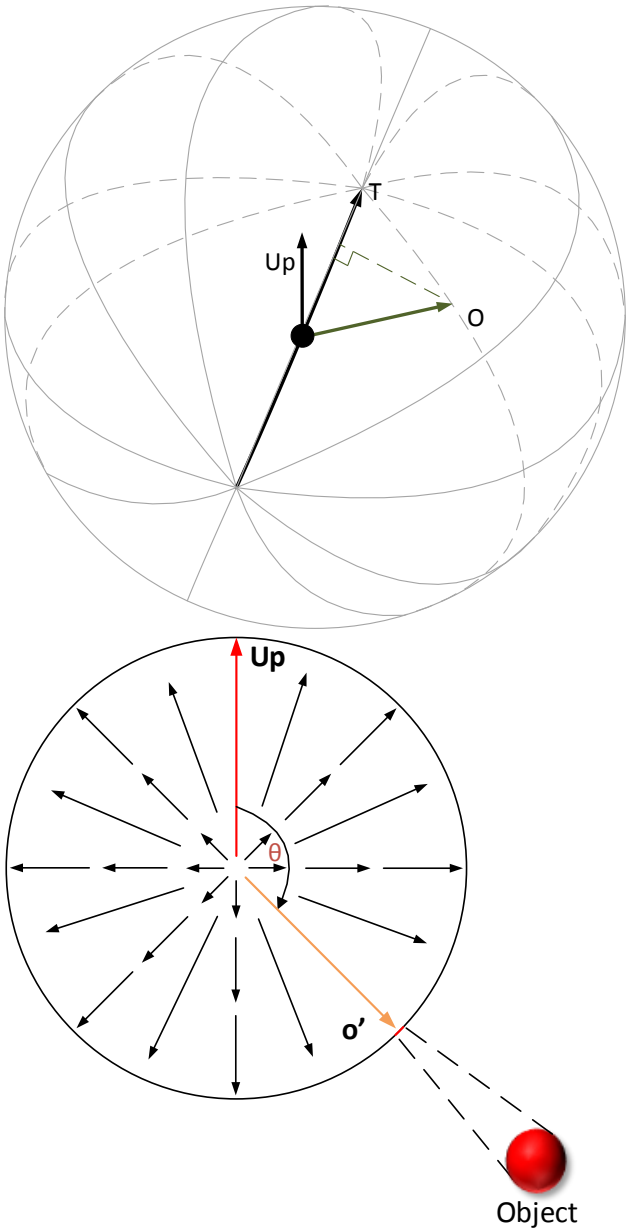

Figure 5. Visual projection representation. Top: A diagram of the rotorcraft translating with an object in the view. Bottom: A diagram to demonstrate the direction of flow vectors of the environment corresponding to the translatory motion, where the orange vector (**o'**) represents the **O** vector in the top diagram in the plane perpendicular to the translation vector.

The plane within which θ is calculated is constantly recomputed from frame to frame according the new translation vector of the rotorcraft. In each frame, the object is deemed to be moving if the direction of its motion between the previous frame and the current frame has deviated significantly from the direction of the epipolar line (θ) in the previous frame i.e. there is significant motion perpendicular to that epipolar line. It must be noted, however, that noise in egomotion estimation introduces noise in the estimate of θ. The standard deviation of this angular error was determined experimentally, to set a threshold disparity for the reliable detection of object motion (see in later section).

4 Rotorcraft

Figure 6. Rotorcraft used in flight tests

4.1 Vision system

The vision system does not need additional sensors apart from a pair of fish-eye cameras, capturing images such as those shown in Figure 2. The computer currently used on the rotorcraft (Figure 6) is a PC104 that uses an Intel core 2 Duo 1.5GHz dual core processor.

The two cameras are angled down by 45 degrees and pointed slightly towards each other to enable computation of stereo in a narrow region of the frontal visual field, if required (not used in this application). Each fish-eye camera has an approximately 180 degree field of view (FOV), which, after the two camera images are combined and remapped, gives the vision system a FOV of approximately 360 degrees (azimuth) by 150 degrees (elevation) [Thurrowgood et al., 2010].

5 Field tests

Figure 7. Photo of a flight test.

Test flights were conducted over a number of days with varying light conditions to assess not only the effectiveness of the algorithm for the detection of a moving object, but also to test the robustness to changes in light intensities, weather conditions and signal to noise ratio (SNR). An example of the test setup is shown in Figure 7.

5.1 Detecting motion disparity

The method uses the angular disparity (Δθ) between the predicted motion of the object and the actual motion of the object to detect motion of the object. The disparity is computed over a number of frames and a running mean is computed, in order to obtain a more reliable estimate. By calculating the value for θ at both c1 and c2 as indicated in

Figure 4, the angular disparity can be computed from the following equation:

$$\Delta\theta = \theta_2 - \theta_1 \quad \textbf{(6)}$$

This method determines that an object is moving when the following condition is met:

$$\Delta\theta > Threshold \quad \textbf{(7)}$$

To determine the optimum angular disparity threshold, histograms of positive responses were calculated for stationary and moving objects, as shown in Figure 8. The optimal threshold was set to the value where the stationary and moving histograms overlap, which was found to be 0.18 radians as indicated by the blue dashed. However, through some experimental analysis, it was found that a conservative value of 0.25 (indicated by black dashed line) radians yielded the most accurate performance.

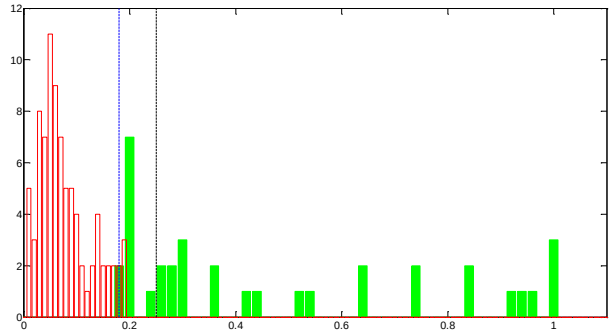

Figure 8. Histograms of positive responses for stationary (red) and moving (green) objects, obtained from field tests. The threshold (0.18) determined for angular disparity is indicated by the vertical blue line, and the actual value used in the tests (0.25) by the vertical black line.

It should be noted that this algorithm is intended for operation in conditions of relatively uniform egomotion without large, abrupt rotations of the aircraft.

5.2 Performance of algorithm

Performance was evaluated by first calculating the true positives (TP) and true negatives (TN) (where a positive is a moving object) and the false positives and false negatives for various assumed threshold levels. Once these were found, a Receiver Operating Characteristic (ROC) was plotted using a 'control' flight test, where the motion of the object was determined by inspection in the video. Figure 9, upper panel, shows the accuracy for various thresholds. The area under the ROC plot gives an indication of the overall performance of the system, where an area of 1 represents perfect performance. Table 1 specifies the accuracy of the method and the area under the ROC. The accuracy is defined as:

$$Accuracy = \frac{TP+TN}{Total\ number\ of\ samples} \quad \textbf{(8)}$$

Performance metric	Percentage
Maximum accuracy	94.4%
% area under ROC	92.8%

Table 1. Performance metrics

Figure 9. Performance in the detection of a moving object as a function of the angular disparity threshold (upper panel) and as revealed by ROC analysis (lower panel). TPR: True Positive Rate; FPR: False Positive Rate.

The upper panel of Figure 9 reveals good accuracy at angular disparity thresholds above 0.2, with a slight, progressive reduction in accuracy as the threshold is increased beyond 0.4. The lower panel of Figure 9 shows that the ROC plot, which also indicates good overall performance - the area under the ROC curve is close to 1 (see Table 1).

5.3 Flight 1: Stationary object

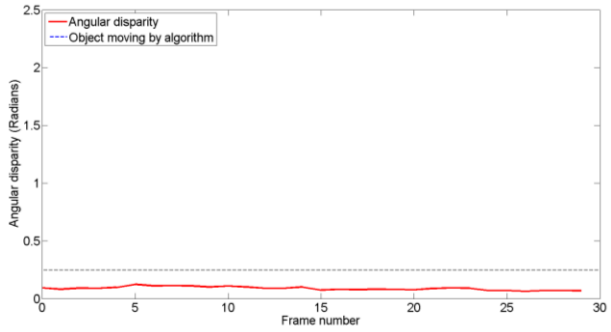

Figure 10. Flight 1: Field test with a stationary object. The horizontal dashed line represents the angular disparity threshold of 0.25 radians.

In this test, the rotorcraft was flown over a stationary ball. A constant angular disparity threshold of 0.25 radians was used to determine if the object was stationary or moving. It can be seen from Figure 10 that the detection algorithm was able to correctly analyse that the object was stationary. This confirms the suitability of the threshold value selected from the data of Figure 8.

5.4 Flight 2: Stationary to moving object

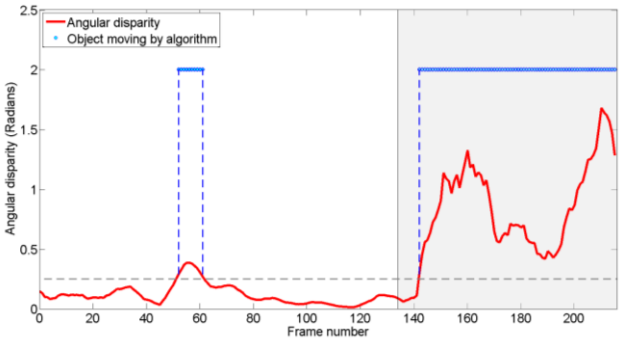

Figure 11. Flight 2: Field test in which an object is initially stationary, and then moves. The horizontal dashed line represents the angular disparity threshold of 0.25 radians. The grey box denotes the period when the object was moving, as determined by manual inspection of the video sequence. The blue circles depict the frames in which the object was deemed to be moving by the algorithm.

In this field test (Figure 11) the rotorcraft translates for a number of frames while the object is stationary, after which the object is moved in three dimensions for a number of frames, while the aircraft continues to translate, under large noise in the images due to low light conditions. We observe that the transition from 'stationary' to 'moving' is detected accurately by the vision system.

There was a slight delay in determining that the ball was moving. This delay depends upon the level of threshold that is chosen, where a compromise has to be set between rapidity of response and the rate of false positive responses. Furthermore, the detection delay is increased if we choose to compute the mean disparity over a number of frames. Another contributor to the delay is that the optic flow is computed over a variable time step. Between frames 52 and 61, the algorithm produced false positives as the rotorcraft was experiencing turbulence with large rotations.

5.5 Flight 3: Intermittently moving object

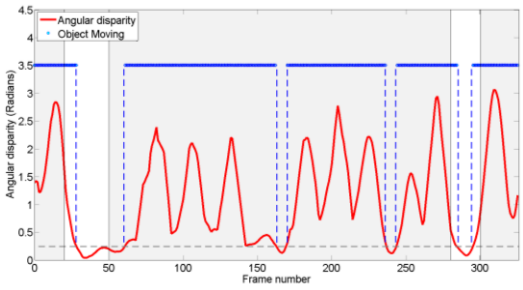

Figure 12. Flight 3: Field test for the detection of a moving object. The grey boxes represent epochs when the ball was moving, as determined by manual inspection of the video sequence. The horizontal dashed line represents the angular disparity threshold of 0.25 radians. The blue circles depict the frames in which the object was deemed to be moving by the algorithm.

In this test (Figure 12), a red ball was moved from one side of the aircraft to the other in three dimensions as the aircraft was translating forwards in conditions of good visibility. The detection algorithm has to account for the ball's transition from moving to stationary and from stationary to moving during the time windows when the ball changes direction.

The oscillatory behaviour in the angular disparity reflects the acceleration and deceleration of the object. The true motion of the ball was deceleration followed by an acceleration at the end of each minimum of the curve, and was zero during the non-shaded regions.

It can be seen in Figure 12 that between frames 150 and 250 that the algorithm produced false negatives due to the relatively conservative threshold that was used. However, Figure 12 demonstrates that the algorithm has good overall performance in detecting a moving object from an airborne platform.

6 Limitations of the method

One of the limitations of the proposed algorithm is that it will not detect an object when the object is translating along a current epipolar line as one cannot determine whether the flow vector is induced by the objects' self-motion or is due to the (currently untracked) relative range to the object. This could be overcome to some degree by optic flow information to compute the relative range to the object over multiple frames, under the assumption that the object is not moving, and monitoring whether this additional assumption is violated.

The reliability with which the angular disparities of the flow vectors are computed is coupled to the limitations of the egomotion determination. As the noise in the optic flow increases, the overall reliability of the system decreases. For example, noise in the computation of optic flow is increased under the low light level conditions, due to increase in image noise.

7 Conclusion

This paper describes a method for detecting the self-motion of objects around a UAV. It is shown that the technique, which combines object detection and tracking with estimation of egomotion from optic flow, enables accurate and robust determination of whether an object in the environment is stationary or moving from the view of a moving platform. The technique is capable of detecting object motion in three dimensions.

Future developments and research will include using the capacity to detect moving objects to create autonomous mid-air collision avoidance systems, and reducing the current limitations of the system at detecting object motion along epipolar lines.

Acknowledgments

Sincere thanks to Michael Knight for flying the rotorcraft on demand and to Dean Soccol for his mechanical expertise. This research was supported by ARC Discovery Grant DP0559306, ARC Centre of Excellence in Vision Science Grant CE0561903, Boeing Defence Australia Grant SMP-BRT-11-044 and by a Queensland Smart State Premier's Fellowship.

References

[Aggarwal and Shah, 1996] Jake K. Aggarwal and Shishir Shah. Intrinsic parameter calibration procedure for a (high-distortion) fish-eye lens camera with distortion model and accuracy estimation. Pattern Recognition 29(11):1775--1788. November 1996.

[Barron and Beauchemin, 1995] John L. Barron and Steven S. Beauchemin. The Computation of Optical Flow. ACM Computing Surveys (CSUR), 27(3):433--466, September 1995.

[Bouthemy et al., 1999] Patrick Bouthemy, Ronan Fablet and Marc Gelgon. Moving Object Detection Color Image Sequences Using Region-Level Graph Labeling. International Conference on Image Processing. Kobe, Japan, 1999.

[Corke et al., 2004] Peter Corke, Dennis Strelow and Sanjiv Singh. Omnidirectional visual odometry for a planetary rover. IEEE/RSJ International Conference on Intelligent Robots and Systems. Sendai, Japan, 2004.

[Cornelis et al., 2008] Nico Cornelis, Luc Van Gool, Bastian Leibe and Konrad Schindler. Coupled Object Detection and Tracking from Static Cameras and Moving Vehicles. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(10):1683--1698. October 2008.

[Chien et al., 2002] Shao-Yi Chien, Shyh-Yih Ma and Liang-Gee Chen. Efficient Moving Object Segmentation Algorithm Using Background Registration Technique. IEEE Transactions on Circuits and Systems for Video Technology 12(7):577--586. July 2002.

[Elhabian et al., 2008] Sumaya H. Ahmed, Shireen Y. Elhabian and Khaled M. El-Sayed. Moving Object Detection in Spatial Domain using Background Removal Techniques - State-of-Art. Recent Patents on Computer Science 1(1):32--34. January 2008.

[Irani and Anandan 1998] Mzchal Irani and P. Anandan. A unified approach to moving object detection in 2D and 3D scenes. Pattern Analysis and Machine Intelligence, IEEE Transactions on 20(6):577--589. June 1998.

[Johnson, 2010] Steven G. Johnson. The NLopt nonlinear-optimization package. 2010.

[Kobayashi and Yanagida, 1995] Hisato Kobayashi and Masaru Yanagida. Moving Object Detection by an Autonomous Guard Robot. 4th IEEE International Workshop on Robot and Human Communication. Tokyo, Japan, 1995.

[Koenderink, 1986] Jan J. Koenderink. Optic Flow. Vision Research, 26(1):161--179, 1986.

[Mizutani et al., 2003] Akiko Mizutani, Javaan S. Chahl and Mandyam V. Srinivasan. Motion camouflage in dragonflies. Nature 423(6940):604--604. June 2003.

[Powell, 2009] Michael J. D. Powell. The BOBYQA algorithm for bound constrained optimization without derivatives. Department of Applied Mathematics and Theoretical Physics. Cambridge England, technical report NA2009/06, 2009.

[Rowley and Rehg, 1997] Henry A. Rowley and James M. Rehg. Analysing Articulated Motion Using Expectation-Maximization. IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Juan, Puerto Rico, 1997.

[Shimizu and Okutomi, 2003] Masao Shimizu and Masatoshi Okutomi. Significance and attributes of subpixel estimation on area-based matching. Systems and Computers in Japan 34(12):1--10. November 2003.

[Srinivasan, 2011] Mandyam V. Srinivasan. Honeybees as a Model for the Study of Visually Guided Flight, Navigation, AND Biologically Inspired Robotics. Physiological Reviews 91(2):413--460. April 2011.

[Srinivasan, 2011] Mandyam V. Srinivasan. Visual control of navigation in insects and its relevance for robotics. Current Opinion in Neurobiology 21(4):535--543.

[Srinivasan and Davey, 1995] Mandyam V. Srinivasan and Matthew Davey. Strategies for active camouflage of motion. Proceedings of the Royal Society of London. Series B: Biological Sciences 259(1354):19--25. January 1995.

[Yalcin et al., 2005] Hulya Yalcin, Martial H. Herbert, Robert T. Collins, Michael J. Black. A Flow-Based Approach to Vehicle Detection and Background Mosaicking in Airborne Video. IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Diego, California 2005.

[Yamaguchi et al., 2006] Koichiro Yamaguchi, Takeo Kato and Yoshiki Ninomiya. Vehicle Ego-Motion Estimation and Moving Object Detection using a Monocular Camera. 18th International Conference on Pattern Recognition. Hong Kong, August, 2006.

[Thurrowgood et al., 2010] Saul Thurrowgood, Richard JD Moore, Daniel Bland, Dean Soccol, and Mandyam V. Srinivasan. UAV attitude control using the visual horizon. In Twelfth Australasian Conference on Robotics and Automation, Brisbane, December 2010.

[Zabala et al., 2012] Francisco Zabala, Peter Polidoro, Alice Robie, Kristin Branson, Pietro Perona and Michael H. Dickinson. A Simple Strategy for Detecting Moving Objects during Locomotion Revealed by Animal-Robot Interactions. Current Biology 22(14):1344--1350. June 2012.