Vision guidance of a fixed wing UAV using a single camera configuration

P.-T. Iong
peteriong@seed.net.tw
Jetpro Technology
Tainan City, Taiwan

S.-H. Chen and Y. Yang
Department of Aeronautics and Astronautics
National Cheng Kung University
Tainan City, Taiwan

ABSTRACT

In this paper a single camera vision guidance system for fixed wing UAV is developed. This system searches for and identifies a target object with known colour and shape from images captured by an onboard camera. HSV colour space and moment invariants are utilised to describe the colour and shape features of the target object. Position, area and rotation angle of the target object in the image plane are collected. This information is then processed by the Extended Kalman Filter to estimate the relative positions and attitudes of the UAV. The vision guidance system guides the UAV towards the target object automatically based on these estimated states by using a proportional controller. A Senior Telemaster aircraft model kit installed with an onboard camera and computer is used for flight test. The target object for the flight test is a white flag with a red cross. Flight simulations and flight tests results are presented in this paper, showing that the vision guidance system can recognise the target object and guide the UAV effectively.

NOMENCLATURE

A_1^{ip} area of subpart 1 of the target object in image plane co-ordinate system divided by grid lines

A_2^{ip} area of subpart 2 of the target object in image plane co-ordinate system divided by grid lines

Paper No. 3780. Manuscript received 21 October 2011, accepted 29 March 2012.

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

A_3^{ip} area of subpart 3 of the target object in image plane co-ordinate system divided by grid lines

A_4^{ip} area of subpart 4 of the target object in image plane co-ordinate system divided by grid lines

A_t^{cc} actual area of the target object in camera-centred co-ordinate system

A_t^{ip} projected area of the target object in image plane co-ordinate system

c_Q aerodynamic damping coefficient of pitch motion

d distance between the target object and the onboard camera

e lateral relative displacement between the aircraft and the target object

$\vec{\mathbf{D}}$ Euclidean norm of feature vectors

f_1 focus length parameter of X axis in image plane co-ordinate system

f_2 focus length parameter of Y axis in image plane co-ordinate system

g gravitational acceleration constant

h relative altitude between the aircraft and the target object

H_t hue of the target object

H_v height of the image captured by onboard camera

I_H hue of pixel in HSV colour space

I_S saturation of pixel in HSV colour space

I_V value of pixel in HSV colour space

i first integer index

j second integer index

$\mathbf{K}_k$ Kalman filter gain of Extended Kalman Filter

K_1 aileron control loop gain

K_2 elevator control loop gain

k third integer index

M_H hue margin of colour filtering

M_{jk} (j,k)th moment of object in index matrix

M_S saturation margin of colour filtering

O a set of connected pixels representing an object in index matrix

$\mathbf{P}_k$ covariance matrix of Extended Kalman Filter

$\mathbf{P}_k(-)$ *a priori* covariance matrix of Extended Kalman Filter

$\mathbf{P}_k(+)$ *a posteriori* covariance matrix of Extended Kalman Filter

P angular velocity along the X axis in aircraft-body co-ordinate system

$\mathbf{Q}_k$ plant noise covariance matrix of Extended Kalman Filter

Q angular velocity along the Y axis in aircraft-body co-ordinate system

$\mathbf{R}_k$ measurement noise covariance matrix of Extended Kalman Filter

Δt increment of time between each Extended Kalman Filter estimation

$\vec{\mathbf{U}}_k$ control input vector of Extended Kalman Filter

V_T velocity of the aircraft

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

w_1 dimension parameter of red cross target object
w_2 dimension parameter of red cross target object
w_3 dimension parameter of red cross target object
w_{1x}^{ip} w_1 projected along the X axis of the image plane co-ordinate system
w_{1y}^{ip} w_1 projected along the Y axis of the image plane co-ordinate system
w_{2x}^{ip} w_2 projected along the X axis of the image plane co-ordinate system
w_{3y}^{ip} w_3 projected along the Y axis of the image plane co-ordinate system
W_v width of the image captured by onboard camera
$\vec{\mathbf{X}}_k$ state vector of Extended Kalman Filter
$\hat{\mathbf{X}}_k(-)$ *a priori* state estimate of Extended Kalman Filter
$\hat{\mathbf{X}}_k(+)$ *a posteriori* state estimate of Extended Kalman Filter
$x_{unrolled}$ vertical unrolled displacement of the target object from the centre of the image plane
x_v vertical co-ordinate of pixel in the index matrix
$\overline{x}_v$ vertical co-ordinate of centre of gravity of an object in index matrix
x_t^{abc} relative position of the target object along X axis of aircraft body centred co-ordinate system
x_t^{cc} relative position of the target object along X axis of camera-centred co-ordinate system
x_0^{ip} co-ordinate of principle point along X axis of image plane co-ordinate system
x_t^{ip} relative position of the target object along X axis of image plane co-ordinate system
$y_{unrolled}$ horizontal unrolled displacement of the target object from the centre of the image plane
y_v horizontal co-ordinate of pixel in the index matrix
$\overline{y}_v$ horizontal co-ordinate of centre of gravity of an object in index matrix
y_t^{abc} relative position of the target object along Y axis of aircraft body centred co-ordinate system
y_t^{cc} relative position of the target object along Y axis of camera-centred co-ordinate system
y_0^{ip} co-ordinate of principle point along Y axis of image plane co-ordinate system
y_t^{ip} relative position of the target object along Y axis of image plane co-ordinate system
$\mathbf{Z_k}$ measurement sensitivity matrix of Extended Kalman Filter
$\overline{\mathbf{Z}}_\mathbf{k}$ measurement vector of Extended Kalman Filter
$\hat{\mathbf{Z}}_\mathbf{k}$ predicted measurement of Extended Kalman Filter
z_t^{cc} relative position of the target object along Z axis of aircraft body centred co-ordinate system
z_t^{abc} relative position of the target object along Z axis of camera-centred co-ordinate system
δ_{ail} deflection angle of the aileron
$\delta_{ail}^{command}$ deflection angle command of the aileron
δ_{ail}^{trim} deflection angle of the aileron at trim state
δ_{elev} deflection angle of the elevator
$\delta_{elev}^{command}$ deflection angle command of the elevator
δ_{elev}^{trim} deflection angle of the elevator at trim state
η_{ail} aerodynamic moment derivatives of the aileron deflection
η_{elev} aerodynamic moment derivatives of the elevator deflection
Θ pitch angle of the aircraft
$\Theta_{command}$ pitch angle command of the aircraft
θ_{oc} angle between the onboard camera axis and the aircraft fuselage axis
$\vec{\Xi}$ feature vector of object in index matrix
$\vec{\Xi}_t$ feature vector of target object

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

$\mathbf{\Phi}_t$ state transition matrix of Extended Kalman Filter
Φ roll angle of the aircraft
$\Phi_{command}$ roll angle command of the aircraft
Φ_t rotation angle of the target object
ψ yaw angle of the aircraft
$\psi_{command}$ yaw angle command of the aircraft
φ_1 first moment invariant of object in index matrix
φ_2 second moment invariant of object in index matrix
η_{jk} (j,k)th normalised moment of object in index matrix
μ_{jk} (j,k)th central moment of object in index matrix

1.0 INTRODUCTION

Inertial navigation systems (INS) and Global Positioning Systems (GPS) are two of the most commonly used instruments for navigating a UAV, and both require the co-ordinates of the target object to perform navigation. If the precise co-ordinates of the target object are unknown, additional payloads such as an onboard camera will be required to collect information from the environment. This information is sent to a ground station, where a ground operator can then analyse and determine the co-ordinates of the target object, and the navigation commands are updated and sent to the flight control system of the UAV. This method requires a long period of communication between the ground operator and the UAV. Consequently, not only is the workload of the ground operator high, but it is also not convenient to use in missions with severe communication conditions. In addition, while onboard cameras are often installed on UAV to provide a visual signal, they are seldom linked directly to automatic flight control. Nevertheless, a vision guidance system can collect information from the environment and determine the location of a target object with known colour and shape features. The flight control system can then guide the UAV towards the target object automatically, without aid from the ground operator. This is the biggest advantage of vision guidance system when compared to GPS or ILS-based navigation systems. Moreover, the vision guidance system can also be integrated with GPS or ILS-based navigation systems. For example, if the target object is known to be in a mission area without precise co-ordinates, the UAV can first be navigated to the mission area by a GPS-based navigation system. The vision guidance system will then identify the target object if it appears within the field of view and visual range of the onboard camera, and the UAV will then be guided towards the target object. The objective of this research is to develop a single camera vision guidance system for fixed wing UAV guidance. The basic limitation of this system is that the target has to be in the field of view and visual range of the onboard camera. The main tasks of the vision guidance system are as follows:

1. Identify a specific target with known colour and shape from the images taken by an onboard camera.
2. Guide the fixed wing UAV towards the specific target.

Vision systems for UAV navigation have been studied widely in recent years and one common topic is autonomous tracking or landing for unmanned rotorcraft, because this platform is very suitable for a vision system. Autonomous landing of rotorcraft is achieved by recognising the pattern of the landing pad either by corner extractions[1,2] or object features[3,4]. Translation and orientation of the rotorcraft relative to the landing pad is then determined and utilised for automatic

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

flight control. The vision system can be also integrated with inertial navigation to limit the errors of the inertial sensors[5]. However, there are limited amounts of research about implementing vision system on fixed wing unmanned aircraft. One study utilises a vision system to detect the horizon from captured images[6,7] and the roll angle and pitch percentage of micro air vehicles are then determined by the position and orientation of the horizon. The lateral stability augmentation system is a proportional-integral feedback control of roll angle error. Pitch angle is not uniquely related to pitch percentage, so pitch percentage feedback acts as a limiter to prevent the horizon from moving out of the field of view of the onboard camera. In another work the vision system is utilised to guide an unmanned glider to fly through a window[8,9]. Captured images are sent to a ground station for image processing, state estimation and control commands calculations. The position and area of the window in the image plane is determined as well as the rotation angle of the window, which is determined by corner detection. The relative positions and altitudes of the glider are then estimated by the Extended Kalman Filter, although the altitude error is ignored in this research. Control commands are calculated by the ground station from the estimated states and sent to the glider by radio signal. This technique has been extended to landing on an airfield with four beacons[10]. The Vision system can also be used to estimate relative positions and velocities between two UAVs to achieve automatic formation flight[11].

The vision guidance system developed in this paper contains object recognition subsystem and vision guidance subsystems. HSV colour space developed by Smith[12] and moment invariants developed by Hu[13] are descriptors for the colour and shape features of the target. The object recognition subsystem compares the features of the target object with the features of the object candidates in the captured images to identify the target object. The location and area of the identified target object in the image plane are then calculated and its rotation angle is determined by a signature analysis method modified from that used for circuit board inspection. The relative positions and attitudes of the UAV are then estimated by the Extended Kalman Filter. Automatic flight control is achieved by use of a proportional controller. This vision guidance system is tested by flight simulations and flight tests, and provides two main features for guidance of UAV. First, the object recognition system allows a flexible choice of target. Once the colour, shape and area of the target are known, it can be recognised by the vision guidance system. The rotation angle detection algorithm in this research only requires the shape of the target to be symmetric for the placing of a grid. Moreover the vision guidance system developed in this paper is fully implemented on a Senior Telemaster model aircraft, and the calculations for objection recognition and guidance are all executed onboard, without aid from a ground station.

2.0 OBJECT RECOGNITION

The block diagram of the vision guidance system is shown in Fig. 1. The object recognition subsystem identifies the target object from images captured by the onboard camera. In this research the target object is assumed to be a two dimensional pattern with the colour and shape features known. In actual application, the colour and shape features of the two dimensional projections of buildings or vehicles can be used for this subsystem. The format of the digital raw images captured by the onboard camera is RGB colour format. A specific colour in RGB colour space is composed of different intensities of red, green and blue. This is not the same as how a person perceives colour in real life. Thus, a more perceptive colour space, the HSV colour space, is utilised to describe colour features in this research. HSV colour space is a hexcone model, and H, S, V stand for hue, saturation and value. Hue describes the colour, such as red or yellow, saturation is defined as the intensity of a specific colour, and value stands for the brightness of the

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

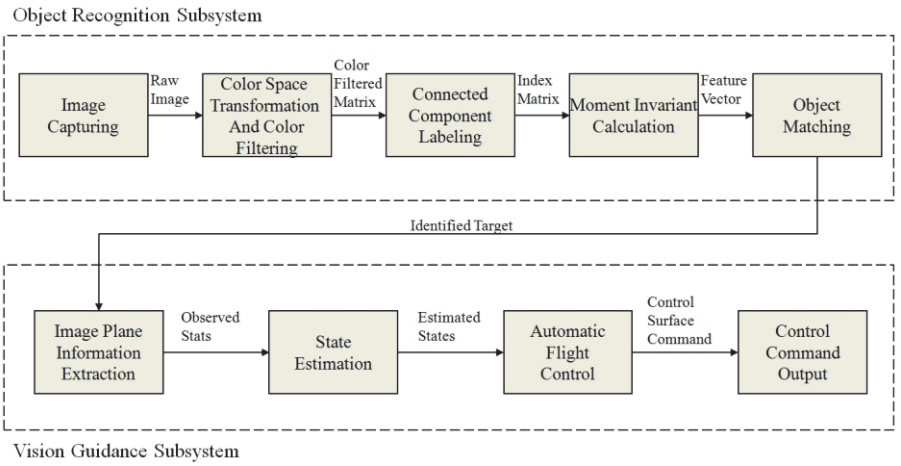


Figure 1. Block diagram of the object recognition subsystem and the vision guidance subsystem.

colour. The algorithm for transforming RGB colour space to HSV colour space was developed by Smith[12]. Every pixel in the raw images is transformed from RGB colour space to HSV colour space and represented by the co-ordinates $I_H(x_v, y_v)$, $I_S(x_v, y_v)$, $I_V(x_v, y_v)$ in HSV colour space. After the colour space transformation, the colour features of each pixel are compared to those of the target object by using the following criteria:

$$\left| I_H(x_v, y_v) - H_t \right| < M_H \quad \ldots (1)$$

$$I_S(x_v, y_v)\, I_V(x_v, y_v) > M_S \quad \ldots (2)$$

H_t denotes the hue of the target object. M_H and M_S represent the hue margin and saturation margin. Equation (1) determines whether the hue of a pixel is similar to the hue of the target object. Equation (2) determines how far the pixel is located away from the central axis of the HSV hexcone. The hue of the pixel is poorly defined near the central axis of the HSV hexcone, so pixels which do not satisfy Equation (2) are rejected. This is called the colour filtering operation. Pixels which satisfy both criteria are labeled as interesting pixels in a new colour filtered matrix and otherwise as uninteresting. Uninteresting pixels represent pixels with colour features not similar with the target object, and they will not be processed by subsequent operations. This strategy allows the computation workload to concentrate on pixels with similar features to the target object, improving the systems calculation time. The colour filtered matrix will then be processed by the connected component labeling operation, which generates an index matrix from the colour filtered matrix. In this index matrix, interesting pixels in the colour filtered matrix that are next to each other are labeled by a unique index, while uninteresting pixels in the colour filtered matrix remain unchanged in the index matrix. As a result, each group of interesting pixels is categorised by an index that is different from the other groups. The connected component labeling algorithm utilised in this paper was developed by Rosenfeld and Pfaltz[14]. This algorithm requires only two passes when building the index matrix, which is achieved by building an index equivalent table during the first pass of the connected component labeling algorithm. The index equivalent table is a table storing information about index pairs in neighbors. Before the second pass is performed, indices belonging to the same object but not in the neighborhood are detected by checking every

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

recorded index pair in the index equivalent table. After relationships between index pairs are resolved, all indices belonging to the same object will be renamed to the same new index. In the second pass of the connected component labeling algorithm, the indices in the index matrix are renamed based on the results of the index equivalent table. After the operations of the connected component labeling, an index matrix with a unique index associated with each object which consists of interesting pixels is obtained. The shape feature of the target object is now ready to be implemented for identification. The describers of the shape features are moment invariants. Moment invariants are descriptors for shape feature which remain constant under variations in translation, rotation and size, as developed by Hu[13]. Hsia[15] suggests dealing with the logarithm of the magnitudes of moment invariants since they usually have a very large dynamic range. Since moment invariants are developed for a continuous image, the effects of image digitalisation on errors of moment invariants were later investigated by Teh and Chin[16], whose results showed that when the pixel size of an object drops under 20, the percentage error rises dramatically. Under this consideration of image digitalisation, interesting pixel groups with less than twenty such pixels are rejected from the calculation of moment invariants. The form of moment invariants used in this research is provided by Haralick and Shapiro[17]. Let O be an object in the index matrix. For each pair of non-negative integers (j, k), the (j, k)th moment of O is given by:

$$M_{jk}(O) = \sum_{(x_v, y_v) \in O} x_v^j y_v^k \quad \ldots (3)$$

The area of the object is $M_{00}(O)$. The centre of gravity of O can be obtained by:

$$\bar{x}_v = \frac{M_{10}(O)}{M_{00}(O)} \quad \ldots (4)$$

$$\bar{y}_v = \frac{M_{01}(O)}{M_{00}(O)} \quad \ldots (5)$$

The central (j,k)th moments of O are defined by:

$$\mu_{jk} = \sum_{(x_v, y_v) \in O} (x_v - \bar{x}_v)^j (y_v - \bar{y}_v)^k \quad \ldots (6)$$

The normalised central (j,k)th moment of O is:

$$\eta_{jk} = \frac{\mu_{jk}}{\mu_{00}^{\frac{j+k}{2}+1}} \quad \ldots (7)$$

The first and second moment invariants which are invariant to translation, scaling and rotation are:

$$\varphi_1 = \eta_{20} + \eta_{02} \quad \ldots (8)$$

$$\varphi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad \ldots (9)$$

The logarithms of first and second moment invariants of each object in the index matrix form a feature vector Ξ.

$$\vec{\Xi} = \begin{bmatrix} \log_{10}\varphi_1 \\ \log_{10}\varphi_2 \end{bmatrix} \quad \ldots (10)$$

The feature vector remains constant for a specific shape under variations of translation, rotation

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press



(a) Raw Image



(b) Color Filtered Matrix Visualized by Binary Image



(c) Index Matrix Visualized by Binary Image



(d) Raw Image with Target Recognized

Figure 2. Example object recognition results. (a) Raw image (b) Colour filtered matrix visualised by binary Image (c) Index matrix visualised by binary Image (d) Raw image with target recognised.

and size caused by variations of distance and attitude of the aircraft. The feature vector Ξ_i represents the feature vector of the *i*th object in the index matrix, while the feature vector Ξ_t represents the feature vector of the known target object. Each feature vector Ξ_i will be compared to the feature vector Ξ_t, and the object with the Ξ_i that best matches Ξ_t will be identified as the target object. A minimum distance classifier is used in this procedure. The Euclidean norm $\mathbf{D}_i$ of Ξ_i to Ξ_t will be calculated for each object, assuming there are *N* objects in the index matrix.

$$\vec{\mathbf{D}}_i = \left\| \vec{\boldsymbol{\Xi}}_i - \vec{\boldsymbol{\Xi}}_\mathbf{t} \right\| \qquad i = 1,2,...,N \qquad \ldots (11)$$

For the *j*th object identified as target object, the following condition must be satisfied.

$$\vec{\mathbf{D}}_j < \vec{\mathbf{D}}_i \qquad \mathrm{i} = 1,2,...,\mathrm{N}, \quad i \neq j \qquad \ldots (12)$$

After the above object recognition operations, the object which best matches the shape of the known target object in the index matrix will be identified as the target object. Since objects in the index matrix represent objects with a colour similar to the known target object in the raw image, the target object identified in the index matrix also matches the colour of the known target object. Figure 2 shows an example object recognition result with the raw image, visualised colour filtered matrix and visualised index matrix with object under 20 pixels rejected.

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

3.0 VISION GUIDANCE

The task of the vision guidance subsystem is to guide the UAV towards the target object. After the target object is identified, information such as its position, size and rotation angle can be obtained from the index matrix. The index matrix represents real world objects projected on the image plane based on the onboard camera parameters. The camera model used in this research is based on the pinhole camera model, where the camera aperture is described as a point, and light from an object projects through this point onto the image plane. The distance between the image plane and the hole is the focus length. The pinhole camera model does not include the effect of lenses, and hence the effects of distortion or unfocused objects caused by lenses are neglected. The pinhole camera model used in this research is a simplified model of the one developed by Heikkilä and Silvén[18], with lens distortions neglected and the principle point assumed to be at the centre of the image plane. The remaining focus length and scale factor are combined to two parameters for vertical and horizontal directions of the image plane. The position of a target object with co-ordinates in camera-centred co-ordinate system (x_t^{cc}, y_t^{cc} and z_t^{cc}) projected on the image plane is then determined by:

$$x_t^{ip} = \frac{f_1 z_t^{cc}}{x_t^{cc}} + x_0^{ip} \quad \ldots (13)$$

$$y_t^{ip} = \frac{f_2 y_t^{cc}}{x_t^{cc}} + y_0^{ip} \quad \ldots (14)$$

x_t^{ip} and y_t^{ip} are the co-ordinates of the target object projected on the image plane co-ordinate system. f_1 and f_2 represent the combined parameters, including the focus length and scale factor of the vertical and horizontal directions. x_0^{ip} and y_0^{ip} are the principle point co-ordinates of the camera, which are assumed to be equal to half the height and width of the image. The superscript '*ip*' denotes measuring in the image plane co-ordinate system, and the superscript '*cc*' donates measuring in the camera-centred co-ordinate system. In this research, the parameters f_1, f_2, x_0^{ip} and y_0^{ip} are determined by the MATLAB camera calibration toolbox developed by Bouguet[19]. Equations (13) and (14) relate the relative position of the target object with its projected co-ordinate on the image plane. But it is not sufficient to solve the three dimensional relative positions of the target object by these two equations. Some vision systems of ground robots utilise a dual-camera configuration to acquire four equations for determining the three dimensional relative positions of the target object. However, a dual-camera configuration means increasing both the payload and image process workload. It is thus preferred to use a single camera configuration in this research due to restrictions on aircraft payload and onboard computer calculation speed. Assuming the area of the target object is also known, the third equation required to solve the three dimensional relative positions of the target object is then obtained by relating the area of the target object to its projected area on the image plane.

$$A_t^{ip} = \frac{f_1 f_2}{(x_t^{cc})^2} A_t^{cc} \quad \ldots (15)$$

Wagter, Proctor and Johnson[8,9] pointed out the roll motion of the UAV affects the location of the target object in the image, and thus it has to be restored to its unrolled state. The roll angle of the UAV is thus required for this calculation. It is detected from the rotation angle of the window target, which is obtained by determining the four corner locations of the window. This method is only

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

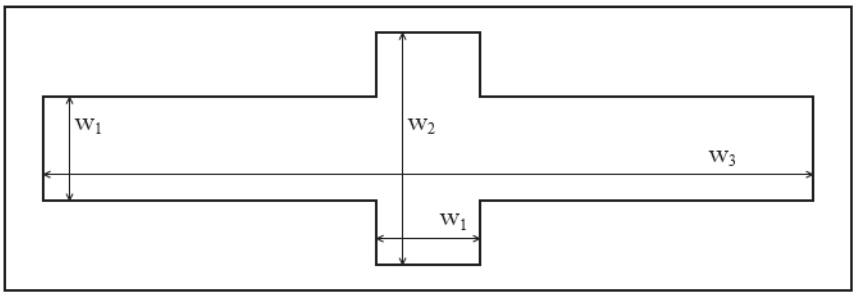


Figure 3. Notations of dimensions of the red cross target object.

suitable for square or rectangular target objects, and in actual applications the shapes of two dimensional projections of buildings or vehicles may be more complicated. The rotation angle detection method in this research is modified from the signature analysis method used for inspecting circuit boards. This modification allows it to be used with target objects at variable distances. This is achieved by varying the grid size according to the distance obtained by Equation (15). This method only requires the shape of the target object to be symmetrical for the placing of a grid. Compared to the horizon detection method for determining roll angle[6,7], there are advantages and disadvantages with the method presented here. The advantage is that only the target object is required to be within the field of view of the onboard camera, while for the horizon detection method, the horizon and the target object both need to be within the field of view. The disadvantage is that the rotation angle detection method used in this research requires the target object to be placed on the ground horizontally, but this is normally the case for most buildings and vehicles, so this is not a severe restriction. This rotation angle detection method is demonstrated with a red cross target object. Figure 3 shows this target object along with its dimensions. The rotation angle detection procedures are as below:

1. The distance of the target object from the camera x_t^{cc} in the camera-centred co-ordinate system is first estimated by solving Equation (15).

$$x_t^{cc} = \sqrt{f_1 f_2 \frac{A_t^{cc}}{A_t^{ip}}} \quad \ldots (16)$$

2. The dimension parameters of the red cross target object w_1, w_2 and w_3 are projected on the image plane co-ordinate system. The results w_{1x}^{ip}, w_{1y}^{ip}, w_{2x}^{ip} and w_{3y}^{ip} are then calculated by substituting w_1, w_2, w_3 and x_t^{cc} into Equations (17) to (20).

$$w_{1x}^{ip} = \frac{f_1 w_1}{x_t^{cc}} \quad \ldots (17)$$

$$w_{1y}^{ip} = \frac{f_2 w_1}{x_t^{cc}} \quad \ldots (18)$$

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

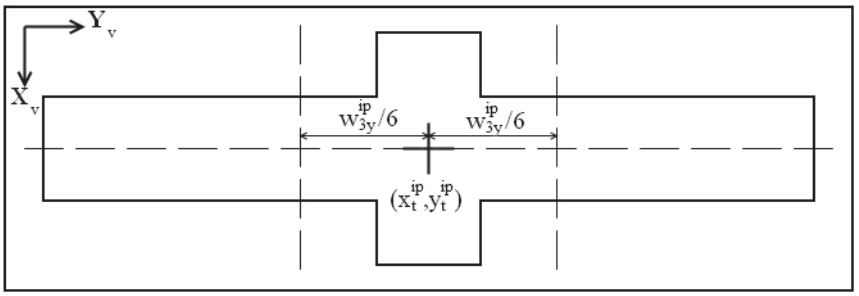


Figure 4. Grid lines placed on target object.

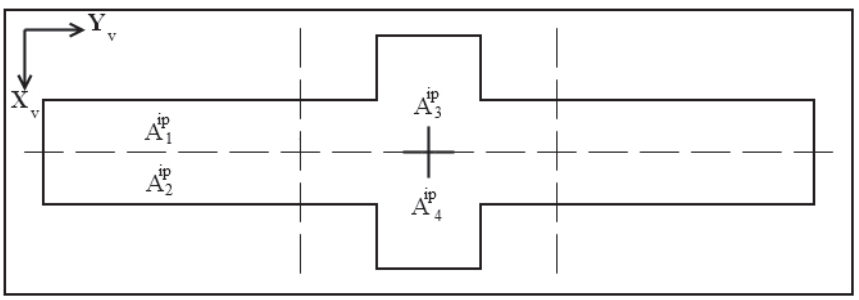


Figure 5. Notations of areas.

$$w_{2x}^{ip} = \frac{f_1 w_2}{x_t^{cc}} \quad \ldots (19)$$

$$w_{3y}^{ip} = \frac{f_2 w_3}{x_t^{cc}} \quad \ldots (20)$$

3. A grid is placed on the centre (x_t^{ip}, y_t^{ip}) of the red cross target object in the image plane co-ordinate system. The horizontal grid line passes through x_t^{ip} and is parallel to the Y axis of the image Plane co-ordinate system. There are two vertical grid lines parallel to the X axis of the image plane co-ordinate system. The interaction point of the vertical grid lines and the horizontal grid line is at $\pm w_{3y}^{ip}/6$ from y_t^{ip}. Figure 4 shows the grid placed on the red cross target object in the image plane.
4. The target object is divided into six parts by the grid lines. The areas of four of these parts are utilised to determine the rotation angle of the target object. The area of these four parts is calculated by counting the number of connected pixels in that part, with the area of each part named as A_1^{ip}, A_2^{ip}, A_3^{ip} and A_4^{ip}, as shown in Fig. 5.

5. The rotation angle Φ_t of the target object is determined by the following equations.

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

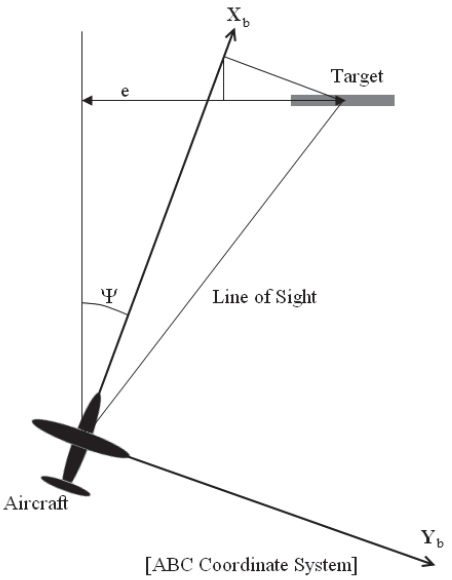


Figure 6. Lateral geometry of the line of sight from the aircraft to the target.

When $A_1^{ip} > A_2^{ip}$

$$\Phi_t = -\mathrm{Cos}^{-1}\left(\frac{w_{1x}^{ip} w_{3y}^{ip}}{3(A_3^{ip} + A_4^{ip} - w_{1y}^{ip}(w_{2x}^{ip} - w_{1x}^{ip})}\right) \quad \ldots (21)$$

When $A_1^{ip} < A_2^{ip}$

$$\Phi_t = \mathrm{Cos}^{-1}\left(\frac{w_{1x}^{ip} w_{3y}^{ip}}{3(A_3^{ip} + A_4^{ip} - w_{1y}^{ip}(w_{2x}^{ip} - w_{1x}^{ip})}\right) \quad \ldots (22)$$

After the rotation angle is determined, the unrolled location of the target object on the image plane can then be determined by the following equations, in which W_v and H_v represent the width and height of the image, respectively:

$$x_{unroll} = (x_t^{ip} - \frac{H_V}{2})\mathrm{Cos}\Phi_t - (y_t^{ip} - \frac{W_V}{2})\mathrm{Sin}\Phi_t \quad \ldots (23)$$

$$y_{unroll} = (x_t^{ip} - \frac{H_V}{2})\mathrm{Sin}\Phi_t + (y_t^{ip} - \frac{W_V}{2})\mathrm{Cos}\Phi_t \quad \ldots (24)$$

The Extended Kalman Filter is then used to estimate the flight dynamic variables for automatic flight control. There are nine estimated states, namely distance, velocity, pitch rate, pitch angle, relative altitude, roll rate, roll angle, yaw angle and relative lateral displacement. The model for updating the *a priori* states is a simplified discrete flight dynamic model. The UAV is assumed to be operating under co-ordinated turn conditions.

$$d_k = d_{k-1} - V_{Tk-1}\Delta t \quad \ldots (25)$$

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

$$V_{Tk} = V_{Tk-1} - g\Theta_{k-1}\Delta t \quad \ldots (26)$$

$$Q_k = Q_{k-1} + c_Q Q_{k-1}\Delta t + \eta_{elev}(\delta_{elevk} - \delta_{elev}^{trim})\Delta t \quad \ldots (27)$$

$$\Theta_k = \Theta_{k-1} + Q_{k-1}\Delta t \quad \ldots (28)$$

$$h_k = h_{k-1} + V_{Tk-1}\Theta_{k-1}\Delta t \quad \ldots (29)$$

$$P_k = P_{k-1} + \eta_{ail}(\delta_{ailk} - \delta_{ail}^{trim})\Delta t \quad \ldots (30)$$

$$\Phi_k = \Phi_{k-1} + P_{k-1}\Delta t \quad \ldots (31)$$

$$\Psi_k = \Psi_{k-1} + \frac{g\Phi_{k-1}}{V_{Tk-1}}\Delta t \quad \ldots (32)$$

$$e_k = e_{k-1} - V_{Tk-1}\Psi_{k-1}\Delta t \quad \ldots (33)$$

To guide the aircraft towards the target object, the angle between the aircraft heading and the line of sight to the target must be minimised. The lateral geometry of the unrolled aircraft and the target object is shown in Fig. 6. The superscript 'abc' denotes measurement in the aircraft body centred co-ordinate system. Using Equation (14) and (24) with assumption of small Φ_t yields:

$$y_{unroll} = \frac{f_2 y_t^{cc}}{x_t^{cc}} \approx \frac{f_2 y_t^{cc}}{d} \quad \ldots (34)$$

From the geometry of Fig. 6, the following equation can be obtained.

$$y_t^{cc} \approx y_t^{abc} = \frac{e - x_t^{abc}\mathrm{Sin}\Psi}{\mathrm{Cos}\Psi} \approx \frac{e - \dfrac{d}{\mathrm{Cos}\theta_{oc}}\mathrm{Sin}\Psi}{\mathrm{Cos}\Psi} \quad \ldots (35)$$

Substituting Equation (35) into Equation (34) results in the following equation:

$$y_{unroll} = \frac{f_2 e}{d\mathrm{Cos}\Psi} - \frac{f_2 \mathrm{Tan}\Psi}{\mathrm{Cos}\theta_{oc}} \quad \ldots (36)$$

Figure 7 shows the longitudinal geometry of the unrolled aircraft and a target object. The angle between the aircraft heading and the line of sight to the target consists of a component contributed by the pitch angle Θ and a component contributed by the difference in altitude between the aircraft and the target object. Using Equations (13) and (23) with small Φ_t.

$$x_{unroll} = \frac{f_1 z_t^{cc}}{x_t^{cc}} \approx \frac{f_1 z_t^{cc}}{d} \quad \ldots (37)$$

From the geometry of Fig. 7, the following equation can be obtained.

$$z_t^{cc} = \frac{h + x_t^{cc}\mathrm{Sin}(\Theta - \theta_{oc})}{\mathrm{Cos}(\Theta - \theta_{oc})} \quad \ldots (38)$$

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

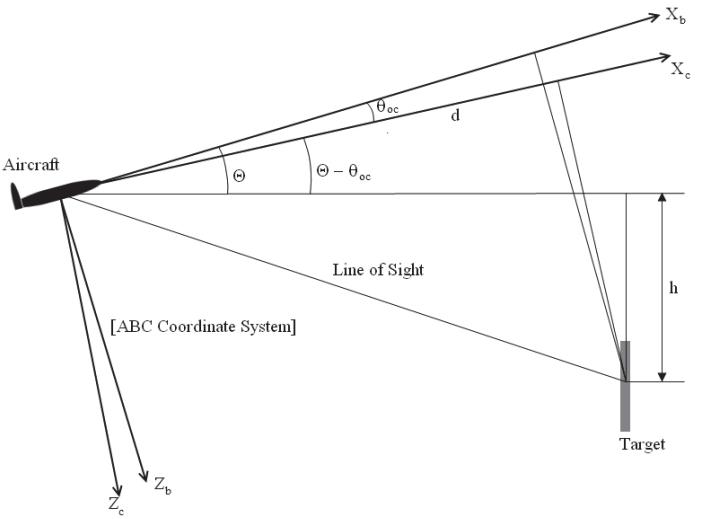


Figure 7. Longitudinal geometry of the line of sight from the aircraft to the target.

Substituting Equation (38) into Equation (37) results in the following:

$$x_{unroll} = f_1 \frac{h}{d\mathrm{Cos}(\Theta - \theta_{oc})} + f_1 \mathrm{Tan}(\Theta - \theta_{oc}) \quad \ldots (39)$$

Equation (15) is then rewritten as:

$$A_t^{ip} = \frac{f_1 f_2 A_t^{cc}}{(x_t^{cc})^1} \approx \frac{f_1 f_2 A_t^{cc}}{(d\mathrm{Cos}\theta_{oc})^2} \quad \ldots (40)$$

The rotation angle of the target object and the roll angle of the aircraft are related by the following equation.

$$\Phi_t \approx \Phi \mathrm{Cos}\theta_{oc} \quad \ldots (41)$$

Equations (36), (39), (40) and (41) form the four measurement equations of the state estimation. Before discussing the procedures of the state estimation, the following definitions are required:

State vector: $\vec{\mathbf{X}}_k = \begin{bmatrix} d_k \\ V_{Tk} \\ Q_k \\ \Theta_k \\ h_k \\ P_k \\ \Phi_k \\ \Psi_k \\ e_k \end{bmatrix}$

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

Control input vector: $\vec{\mathbf{U}}_k = \begin{bmatrix} \delta_{elevk} \\ \delta_{ailk} \end{bmatrix}$

Measurement vector: $\vec{\mathbf{Z}}_k = \begin{bmatrix} x_{unrollk} \\ y_{unrollk} \\ A_t^{ip}{}_k \\ \Phi_{tk} \end{bmatrix}$

Covariance matrix: $\mathbf{P}_k$
Plant noise covariance matrix: $\mathbf{Q}_k$
Measurement noise covariance matrix: $\mathbf{R}_k$

The state estimation procedures are as follows:

1. After the target object is identified the first time, the initial state vector is generated. V_{T0} and h_0 are initialised by an estimated flight speed and altitude when the aircraft approaches the target. In this research, they are estimated as 50ft/sec and 50ft. d_0, Θ_0, $\mathbf{\Phi}_0$ are initialised by using Equations (40), (39) and (41), respectively. The lateral relative position between the camera and the target object e_0 and the yaw angle ψ_0 is coupled in Equation (36), So it is not possible to estimate their values in this equation. An initial guess of $\psi_0 = 0$ is utilised to estimate e_0 from Equation (36).
2. The estimation starts from the next time step after Δt seconds. Δt is not a fixed value since the time required for object recognition is not fixed. Δt is determined by comparing the CPU clock time of the onboard computer between two vision guidance cycles.
3. The *a priori* state estimate $\hat{\mathbf{X}}_k(-)$ is obtained by substituting $\hat{\mathbf{X}}_{k-1}(+)$ and $\vec{\mathbf{U}}_{k-1}$ into Equations (25) to (33), respectively.
4. The predicted measurement $\hat{\mathbf{Z}}_k$ is obtained by substituting $\hat{\mathbf{X}}_{k-1}(+)$ into Equations (36), (39), (40) and (41).

5. The linear approximate state transition matrix and measurement sensitivity matrix are obtained by:

$$\mathbf{\Phi}_{k-1} \approx \begin{bmatrix} 1 & -\Delta t & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -g\Delta t & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \Delta t & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \Theta_{k-1}\Delta t & 0 & V_{Tk-1}\Delta t & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \Delta t & 1 & 0 & 0 \\ 0 & -\dfrac{g\Phi_{k-1}\Delta t}{(V_{Tk-1})^2} & 0 & 0 & 0 & 0 & \dfrac{g\Delta t}{V_{Tk-1}} & 1 & 0 \\ 0 & -\Psi_{k-1}\Delta t & 0 & 0 & 0 & 0 & 0 & -V_{Tk-1}\Delta t & 1 \end{bmatrix} \quad \ldots (42)$$

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

$$\mathbf{H}_k \approx \begin{bmatrix} -\dfrac{f_1 h_k}{(d_k)^2} & 0 & 0 & f_1 & \dfrac{f_1}{d_k} & 0 & 0 & 0 & 0 \\ -\dfrac{f_2 e_k}{(d_k)^2} & 0 & 0 & 0 & 0 & 0 & 0 & -f_2 & \dfrac{f_2}{d_k} \\ -\dfrac{2f_1 f_2 A_t^{cc}}{d_k^3 (\mathrm{Cos}_{,\,oc})^2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad \ldots (43)$$

6. The *a priori* covariance matrix is obtained by:

$$\mathbf{P}_k(-) = \mathbf{\Phi}_{k-1}\mathbf{P}_{k-1}(+)\mathbf{\Phi}_{k-1}{}^{\mathrm{T}} + \mathbf{Q}_{k-1} \quad \ldots (44)$$

7. The Kalman filter gain is obtained by:

$$\overline{\mathbf{K}}_k = \mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}}[\mathbf{H}_k\mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}} + \mathbf{R}_k]^{-1} \quad \ldots (45)$$

8. If the target object does not disappear from the image plane, the *a posteriori* state estimated will be obtained by:

$$\hat{\mathbf{X}}_k(+) = \hat{\mathbf{X}}_k(-) + \overline{\mathbf{K}}_k(\vec{\mathbf{Z}}_k - \hat{\mathbf{Z}}_k) \quad \ldots (46)$$

If the target object disappears from the image plane for less than or equal to three cycles, the *a posteriori* state estimated will be obtained by:

$$\hat{\mathbf{X}}_k(+) = \hat{\mathbf{X}}_k(-) \quad \ldots (47)$$

9. The *a posteriori* covariance matrix is obtained by:

$$\mathbf{P}_k(+) = (\mathbf{I} - \overline{\mathbf{K}}_k\mathbf{H}_k)\mathbf{P}_k(-) \quad \ldots (48)$$

10. Go Back to step 2 and repeat until the target object is lost from the field of view for more than three cycles.

The symbol (–) represents an *a priori* estimation. The symbol (+) represents an a posteriori estimation. These estimated flight dynamic states will be utilised for feedback control. To guide the aircraft towards the target object, the tactic is to keep the velocity vector of the aircraft aligned with the line of sight between the aircraft and the target object. When the angle-of-attack and the sideslip angle are small, the velocity vector of the aircraft almost aligns with the fuselage reference axis. If the onboard camera is installed so that the angle θ_{oc} between the camera axis and the fuselage reference axis is small, this method will also keep the target object in the field of view of the onboard camera.

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

The objective of lateral control is to minimise the angle between the aircraft heading and the line of sight to the target. The required yaw angle command is obtained from Fig. 6.

$$\Psi_{command} = \mathrm{Tan}^{-1}\frac{e}{x_t^{abc}\mathrm{Cos}\Psi - y_t^{abc}\mathrm{Sin}\Psi} \approx \mathrm{Tan}^{-1}\frac{\frac{e}{d}}{\frac{\mathrm{Cos}\Psi}{\mathrm{Cos}\theta_{oc}} - \frac{y_t^{abc}}{d}\mathrm{Sin}\Psi} \quad \ldots (49)$$

Substituting Equation (35) into Equation (49) and using small angle approximation results the following equation.

$$\Psi_{command} \approx \mathrm{Tan}^{-1}\frac{\hat{e}_k(+)}{\hat{d}_k(+)\mathrm{Cos}\hat{\Psi}_k(+) - \hat{e}_k(+)\mathrm{Tan}\hat{\Psi}_k(+) + \hat{d}_k(+)\mathrm{Sin}\hat{\Psi}_k(+)\mathrm{Tan}\hat{\Psi}_k(+)} \quad \ldots (50)$$

The yaw angle command is achieved by controlling the aileron to perform a turn, and the Equation (32) is utilised to calculate the roll angle required to achieve the yaw angle command.

$$\Phi_{command} = \frac{\hat{V}_{Tk}(+)}{g\Delta t}(\Psi_{command} - \bar{\Psi}_k(+)) \quad \ldots (51)$$

The aileron command is then determined by a proportional feedback controller:

$$\delta_{ail}^{command} = K_1(\Phi_{command} - \hat{\Phi}_k(+)) \quad \ldots (52)$$

To point the aircraft fuselage reference axis towards the target object, the required pitch angle is:

$$\Theta_{command} = -\mathrm{Tan}^{-1}\frac{\hat{h}_k(+)}{\hat{d}_k(+)} \quad \ldots (53)$$

The elevator command is then determined with a proportional feedback controller:

$$\delta_{elev}^{command} = K_2(\Theta_{command} - \hat{\Theta}_k(+)) \quad \ldots (54)$$

After the control surface commands are determined, they will be first added to their trim values, and then be recorded for the next state estimation and sent to the control surface actuators.

4.0 DEVICES USED IN THE EXPERIMENT

The aircraft used in this research is the Senior Telemaster model kit produced by Hobby-Lobby International. It is constructed mainly of balsa and maple, and a radio receiver and servo set produced by JR Radios is installed on the aircraft for remote control. The weight of the Senior Telemaster without fuel and batteries is 4·9kg, and the maximum take-off weight is 10kg. The detailed specifications of the Senior Telemaster are listed in Table 1, while Fig. 8 shows a photo and Fig. 9 shows the positions of the devices installed on it. The onboard computer used in this research is the speedMOPSlcdPM produced by Kontron Embedded Computers. The

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

Table 1
Specifications of Senior Telemaster

Wing span	95in	Length	63in
Wing area	1,336in^2	Empty weight	4·9kg
Wing dihedral angle	3·6°	Maximum take-off weight	10kg
Wing aerofoil	NACA-4412	Engine	OS MAX AX 120
Wing incidence angle	4·5°		



Figure 8. Senior Telemaster.

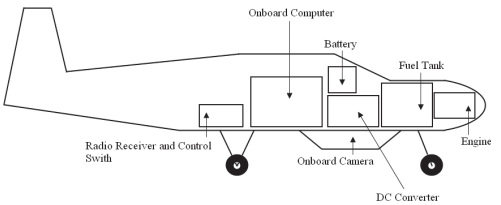


Figure 9. Positions of devices installed on Senior Telemaster.

speedMOPSlcdPM is a PC-104 CPU module which can also be used as a stand-alone CPU module, requiring a 5 volt power supply. A Hi-speed 432 model disk-on-module produced by Power Quotient International Limited is mounted on the onboard computer to store the operating system and the vision guidance system. The Hi-speed 432 model disk-on-module requires a 5 volt power supply. The onboard camera connected to the speedMOPSlcdPM also drains power from the speedMOPSlcdPM through the USB port. The total power consumption of the above devices is expected to be less than 20 watt, and power is supplied by an 11·1 volt 3300 mAh lithium-ion polymer battery. The input voltage is transferred from 11·1 to 5 volt by an SDS-035A model DC to DC switching power supply produced by Sunpower Technology Corporation. The onboard camera used in this research is a Logitech Quickcam Messenger produced by Logitech International. This is a CMOS digital camera that connects to the onboard computer through the USB port. The camera is mounted in a camera box, and the camera box is installed under the fuselage of the Senior Telemaster. The angle between the camera axis and the fuselage reference axis is 10°. The camera is set to transmit a 320×240 resolution, 24 bit RGB colour image to the onboard computer, and raw images are stored in three dimensional matrices (320×240×3 matrices). During the flight experiment, the onboard computer will start to send control surface commands after the target object is identified, with these commands in the format of an RS232 signal. This RS232 signal will be transferred into a pulse width modulation (PWM) signal by the Mini SSC II servo controller produced by Scott Edwards Electronics Incorporation. The resolution of the control surface angle is 0·36°, which is based on the restrictions of the Mini SSC II servo controller. During take-off and landing of the Senior Telemaster, it is remote controlled by the ground operator, who can switch the aircraft to follow their commands or to follow those from the onboard computer using the gear channel of the radio transmitter. The control switch module

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

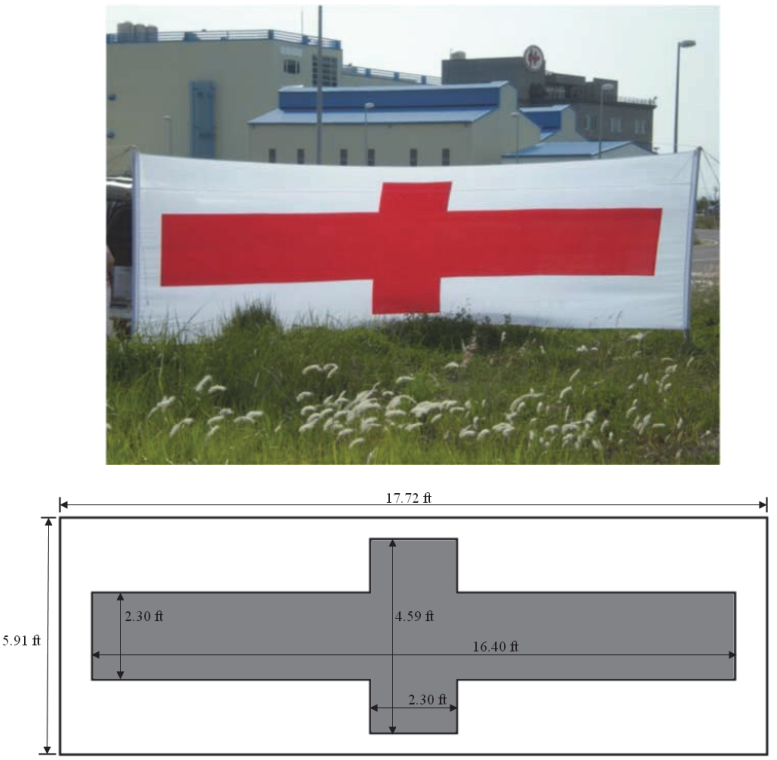


Figure 10. Photo and dimension of the target object.

consists of a quad bilateral switch, which is a TC4066 chip produced by Toshiba Semiconductor, powered by a 4·8 volt nickel-cadmium battery.

The operating system of the onboard computer is Windows XP, and the program for the vision guidance system is coded in Visual C++ 6.0. The camera driver provided by Logitech includes some basic camera controls, such as automatic exposure time and automatic white balance, and these are set to their default values. An ActiveX component — VideoOCX, developed by Marvelsoft, is implemented to work as a bridge between the vision guidance system program and the Logitech camera driver. VideoOCX includes additional functions for camera control, such as image capture and display. The image capture function is utilised to capture images from the onboard camera and store them in a 320×240×3 matrix, and the vision guidance system can then process the matrix and identify the target object. The image display function of VideoOCX is utilised to display images captured by the onboard camera on the user interface. Every image captured by the camera is also saved in the JPG format, and are used for analysis after experiments. The vision guidance system will start the estimation process once the target object is recognised, and will reset the state estimation if the target object disappears from the field of view of the onboard camera for more than three consecutive frames.

The target object used in the ground and flight experiments in this research is a white flag with a red cross, as shown in Fig. 10. During the ground and flight experiments, the white flag is supported by two flag poles, which are fixed on the ground by ropes and nails.

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

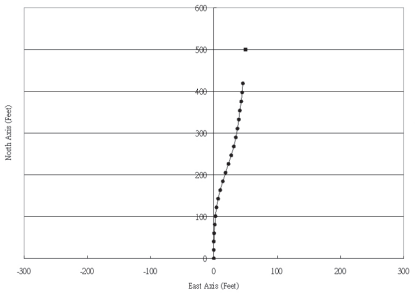


Figure 11. Top view of a UAV flight path in flight simulation.

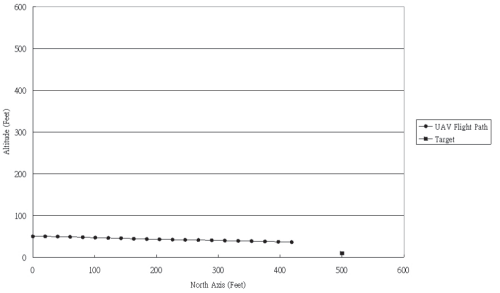


Figure 12. Side view of UAV flight path in flight simulation.

5.0 RESULTS AND DISCUSSION

In this research, the vision guidance system is tested by flight simulations and flight experiments. The nonlinear flight dynamic model is numerical integrated by using the Runge-Kutta algorithm. The position and size of the target object in the image plane are obtained by co-ordinate transformation equations and camera model equations. This flight simulation is coded using FORTRAN, and the sample time of the vision guidance system during flight simulation is assumed to be 0·4 seconds. The observed target object position, size and rotation angle are corrupted by Gaussian white noise to simulate the noisy condition in actual operating environment. In this flight simulation, the aircraft is flying at initial trim condition with 50ft/sec airspeed and 50ft altitude. A target object is placed 500ft ahead of the aircraft with 50ft lateral displacement, while the altitude of the object is 10ft. The target object is identified at 0 seconds, and the first state estimation and automatic flight control start from 0·4 seconds. The vision guidance system is forced to be unable to acquire the target between 2·0 and 2·8 seconds to simulate the situation of object recognition failure. Figure 11 shows the top view of the aircraft flight path in the north-east-down co-ordinate system, where Fig. 12 shows the side view of the same path. Figure 13 shows the time history of aileron and elevator commands. The estimated states: distance from target, pitch angle, relative altitude, roll angle, yaw angle and relative lateral displacement, are plotted with the true simulated states in Figs 14 to 19. From Figs 11 and 12, the results of the flight path simulation show that the vision guidance can guide the aircraft towards the target. In this process, the lateral guidance is more effective than longitudinal guidance, and this is because the relative lateral displacement (Fig. 19) converges much faster than the relative altitude (Fig. 16). The difference between the assignment of the initial guess value of the yaw and pitch angle is probably one of the reasons. In the lateral case, from Equation (36), the horizontal unrolled displacement of the target object is determined by the variables of distance to target (d), relative lateral displacement(e) and yaw angle (ψ). The initial guess value of the yaw angle is set to be zero. This means the X axis of aircraft body centred co-ordinate system at the moment of target first acquired is set to be the reference axis for yaw angle during the subsequent estimation. This renders the initial error of yaw angle. The relative lateral displacement is then estimated under the condition of zero initial error of yaw angle. Thus it converges faster taking advantages of this mechanism. In the longitudinal case, from Equation (39), the vertical unrolled displacement of the target object is determined by the variables of distance to target (d), relative altitude (h) and pitch angle (Θ). In this case the initial values of relative altitude and pitch angle are both unknown. It is also inadequate to assign the Y axis of aircraft body centred co-ordinate system at the moment of target first acquired to be the reference axis for pitch angle during the subsequent estimation because the estimation of pitch angle

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

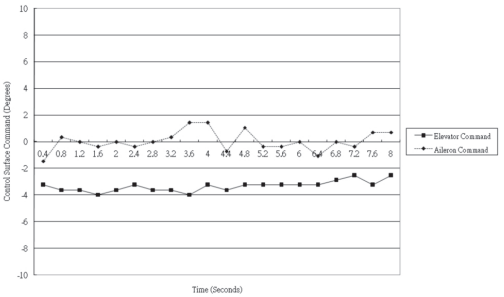


Figure 13. Control surface command time history in flight simulation.

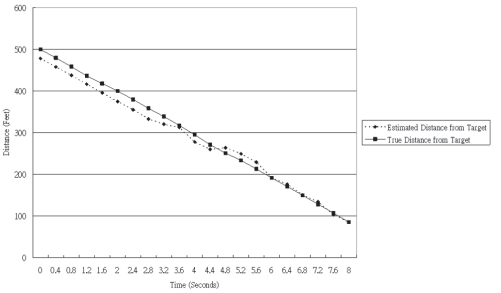


Figure 14. Estimation of distance from target in flight simulation.

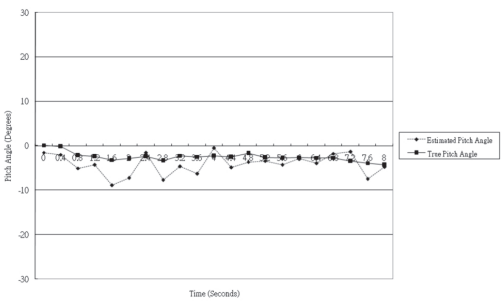


Figure 15. Estimation of pitch angle in flight simulation.

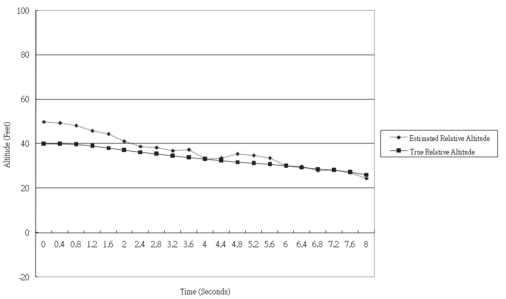


Figure 16. Estimation of relative altitude in flight simulation.

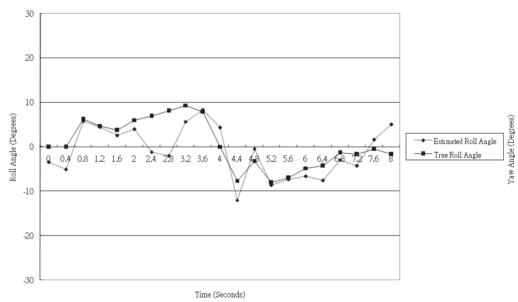


Figure 17. Estimation of roll angle in flight simulation.

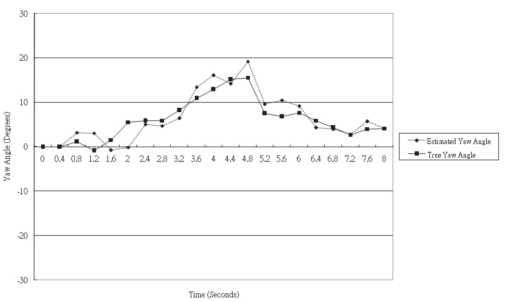


Figure 18. Estimation of yaw angle in flight simulation.

between the aircraft and the horizon is important for flight safety. In this research, the single camera navigation configuration without aid of other sensors, the initial values of pitch angle and relative altitude cannot be acquired. So the initial relative altitude is guessed to be 50ft and the initial pitch angle is determined from Equation (39). Both values may differ from the actual initial relative altitude and pitch angle. This causes the slower convergence speed of longitudinal dynamic states comparing to lateral dynamic states. This situation can be resolved by integrating with other attitude sensor to provide initial value for the estimation. The forced missing of target recognition between 2·0 and 2·8 seconds doesn't have much impact on the estimation of relative lateral displacement, yaw angle, relative altitude and pitch angle. From Fig. 17 it can be seen that the roll angle estimation is serious affected. But it recovers quite fast after the target is recognised again.

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

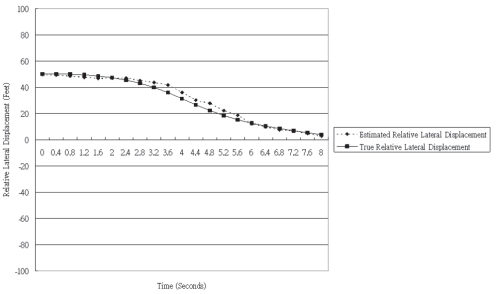


Figure 19. Estimation of relative lateral displacement in flight simulation.

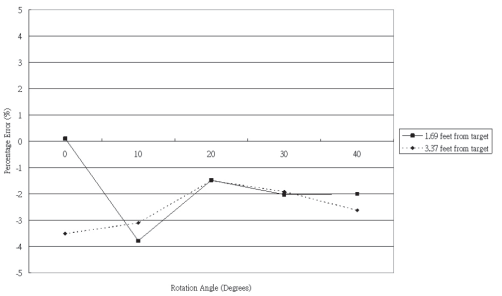


Figure 20. Average percentage error of target area detection in ground test.

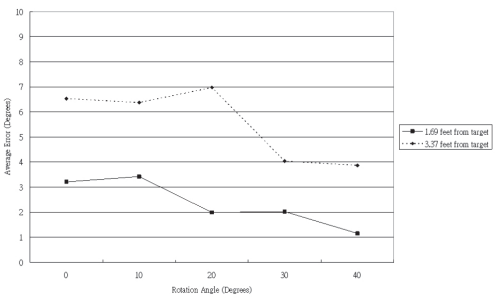


Figure 21. Average error of target rotation angle detection.

The target area and rotation angle detection algorithms presented in Section 3 were tested in a ground experiment. Since the flatness of the target object in Fig. 10 is easily affected by wind, an outdoor experiment is not ideal for verifying the target area and rotation angle detection algorithms. Instead this test was performed indoor. A scale-down 'red cross' target object with its size equal to 1/1,225 of the original target object was created. This scaled-down target object was reproduced with rotation angles from 0 to 40° for this experiment. The onboard camera was tested at two spots which were 1·69ft and 3·37ft from the scale-down target object. Fifty object recognition cycles were performed for each rotation angles and each distances. The area of the scale-down target object was 0·03506ft^2. The area of the scale-down target object after projected on the image plane was obtained by Equation (40). The average percentage errors of target area detection are plotted in Fig. 20. The average errors of target rotation angle detection are plotted in Fig. 21. Figure 20 shows the average percentage error of target area detection is within 5%. The error range is acceptable. The main reasons of target area detection error are probably caused by camera noise. The colour features of some pixels of the target object are affected which causes them rejected by colour filtering procedure. The resulting target area in the index matrix is thus smaller than its actual area in the image plane. When the area of the target object is smaller in the image plane, the effects of camera noise are more obvious. This is shown by the average percentage error at 3·37ft larger than the average percentage error at 1·69ft in Fig. 20. Camera noise also affects target rotation angle detection because the target rotation angle detection algorithm is based on area detection of subparts of the target object. Figure 21 shows the error of the target rotation angle detection is larger when the camera is farther from the target. This is because the error of area detection of subparts is more

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

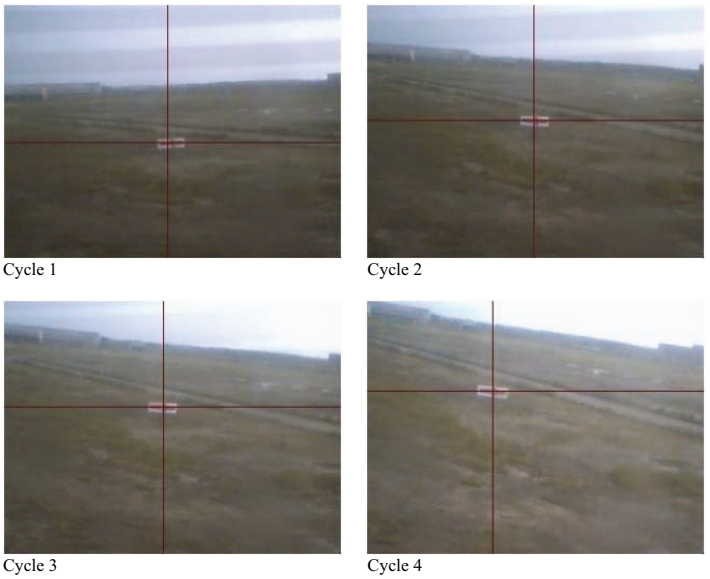


Figure 22. Images captured and processed in flight tests.

sensitive when the camera is farther from the target. Figure 21 also shows the rotation angle detection error is larger when the target rotation angle is near zero. This is because the slope of the inverse cosine term of the target rotation angle detection Equation (21) and Equation (22) is steep when the target rotation angle is near zero. Little changes of areas of subparts of the target object cause large changes of rotation angle.

A series of flight tests have been performed to test the vision guidance system. During the flight test, The Senior Telemaster was manually controlled to take off and fly near to the target object, the ground operator then switched the aircraft into automatic flight mode. The Senior Telemaster would then be guided by the vision guidance system for a short time after the target object was recognised by the vision guidance system. Before the Senior Telemaster crashed into the target object, the ground operator switched the aircraft back into manual flight mode. Figure 22 shows a sequence of flight test images processed and stored by the vision guidance system during one of this automatic flight. The crossed lines in the images represent the locations where the target object was recognised to be. Other data stored on the onboard computer in this automatic flight test are shown in Fig. 23 to Fig. 26.

This test lasted for 6·219 seconds, and 15 vision guidance cycles were executed. Each vision guidance cycle is given a code name in numerical order, from 'Cycle 1' to 'Cycle 15', for demonstration in the following figures. In Fig. 23, the operation time of each vision guidance cycles ranged from 0·25 to 0·7 seconds. The average operation time of each cycle is 0.323 seconds, which is close to the assumed value of 0·4 seconds during the flight control system design and flight simulations. The in-flight image processed by the vision guidance system in Fig. 22 shows that the target object was recognised successfully during this automatic flight. The vision guidance system also tried to guide the aircraft and kept the target object centred in the field of view. Comparing the lateral position of the target object in Fig. 22, the estimated relative lateral displacement in Fig. 24 and the estimated yaw angle in Fig. 25, it can be seen that the lateral

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

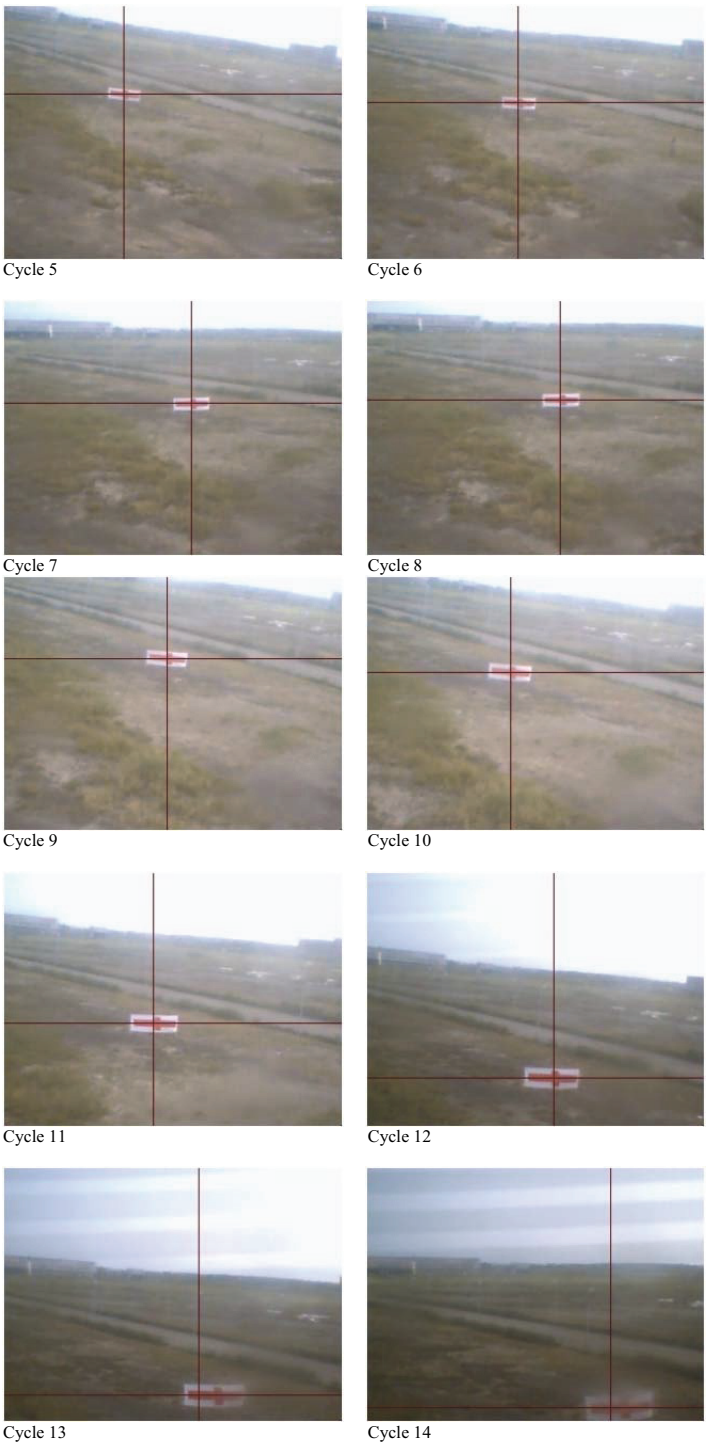


Figure 22 (continued). Images captured and processed in flight tests.

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press



Cycle 15

Figure 22 (continued). Images captured and processed in flight tests.

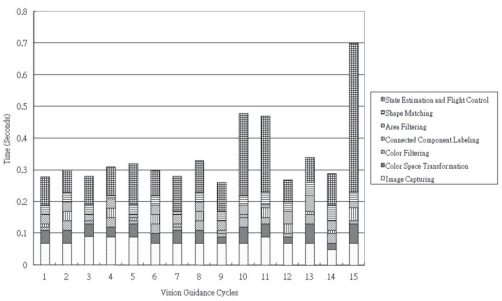


Figure 23. Operation time of vision guidance cycles in flight test.

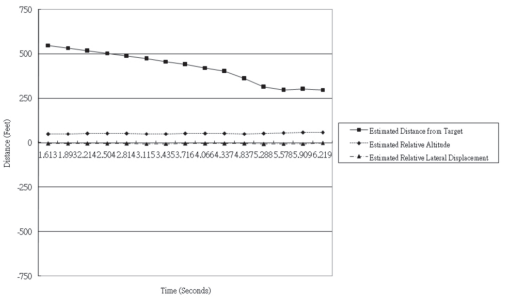


Figure 24. Estimation of distance from target, relative altitude and relative lateral displacement in flight test.

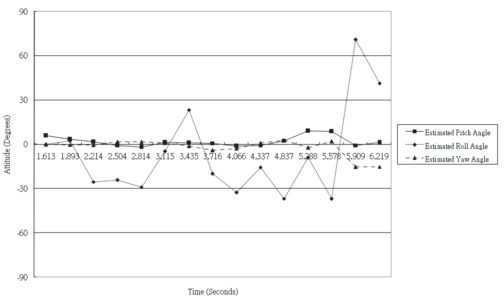


Figure 25. Estimation of attitude angles in flight test.

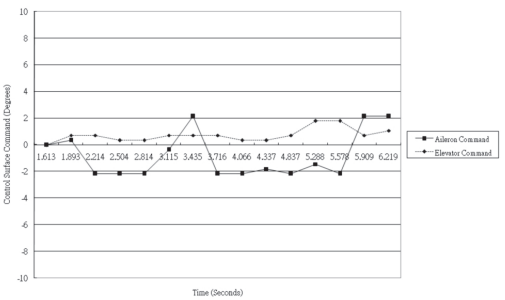


Figure 26. Control surface command time history in flight test.

guidance performed quite good in keeping lateral flight direction under gust disturbances. The estimate of roll angle is also acceptable in the initial 12 cycles. Staring from Cycle 13 the target object moved into the lower right corner of the camera, which was probably corrupted by fuel particle from the engine exhaust of the aircraft. Most of the blurred pixels in the right side of the target object were then probably discarded by the vision guidance system. As a result the centre of the target object is recognised with an offset to the left. This also made the grid for rotation angle detection placed at an incorrect position and largely affected the accuracy of roll angle estimation. For the longitudinal flight control, thought the elevator was given commands to pitch down to point towards the target. The reduction of relative altitude was not significant. The first reason is due to the slow convergence speed of longitudinal dynamic states estimation discussed above. The second reason is probably due to lack of automatic airspeed control.

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

6.0 CONCLUSIONS

The vision guidance system developed in this research integrated image processing, state estimation and automatic flight control. It is able to identify a target with known colour, shape and area but without knowledge of precise location. The choice of the target is flexible, as longitudinal and lateral flight dynamic states and relative positions are estimated from image plane information. Once the target appears in the field of view of the onboard camera and is recognised by the vision guidance system, the UAV will be guided automatically towards it. This system is fully implemented on a Senior Telemaster model aircraft without aid from a ground station, with object recognition and vision guidance performed on every image frame captured by the onboard camera. Flight simulation and flight test results show that the vision guidance system works as intended. Though there is the need to improve the longitudinal guidance, this is mainly restricted from the single camera configuration. With advancements of sensors and processors technology, it is quite possible to fulfill better image only guidance by multi camera configuration in short future. This system can also complement GPS navigation by providing the ability to search for a target without knowledge of its location, and thus it is very suitable to serve as a terminal guidance system in an integrated GPS/vision system.

REFERENCES

1. SHARP, C.S., SHAKERNIA, O. and SASTRY, S.S. A vision system for landing an unmanned aerial vehicle, 2001, IEEE International Conference on Robotics and Automation, Seoul, Korea, pp 1720-1727.
2. SHAKERNIA, O., SHARP, C.S., VIDAL, R., SHIM, D.H., MA, Y. and SASTRY, S. Multiple view motion estimation and control for landing an unmanned aerial vehicle, 2002, IEEE Conference of Robotics and Automation, Washington DC, USA, pp 2793-2798.
3. SARIPALLI, S. Visually guided landing of an unmanned aerial vehicle, *IEEE Transactions on Robotics and Automation*, 2003, **19**, (3), pp 371-381.
4. SARIPALLI, S. and SUKHATME, G.S. Landing on a moving target using an autonomous helicopter, 2003, International Conference on Field and Service Robotics, Mt Fuji, Japan.
5. WU, A.D., JOHNSON, E.N. and PROCTOR, A.A. Vision-aided inertial navigation for flight control, *J Aerospace Computing, Information, and Communication*, 2005, **2**, (9), pp 348-360.
6. ETTINGER, S.M., NECHYBA, M.C., IFJU, P.G. and WASZAK, M. Vision-guided flight stability and control for micro air vehicles, 2002, 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems EPFL, Lausanne, Switzerland, pp 2134-2140.
7. KEHOE, J.J., CAUSEY, R., ABDULRAHIM, M. and LIND, R. Waypoint navigation for a micro air vehicle using vision-based attitude estimation, 2005, AIAA Guidance, Navigation, and Control Conference, San Francisco, USA.
8. WAGTER, C.D., PROCTOR, A.A. and JOHNSON, E.N. Vision-only aircraft flight control, 2003, AIAA Digital Avionics Conference, Indianapolis, USA.
9. PROCTOR, A.A. and JOHNSON, E.N. Vision-only aircraft flight control methods and test results, 2004, AIAA Guidance, Navigation, and Control Conference and Exhibition, Providence, Rhode Island, USA.
10. PROCTOR, A.A. and JOHNSON, E.N. Vision-only approach and landing, 2005, AIAA Guidance, Navigation and Control Conference and Exhibition, San Francisco, California, USA.
11. OH, S.M. and JOHNSON, E.N. Relative motion estimation for vision-based formation flight using unscented Kalman filter, 2007, AIAA Guidance, Navigation and Control Conference and Exhibition, Hilton Head, California, USA.
12. SMITH, A.R. Color gamut transform pairs, *ACM SIGGRAPH Computer Graphics archive*, **12**, (3), 1978, pp 12-19.
13. HU, M.K. Visual pattern recognition by moment invariants, *IRE Transactions on Information Theory*, **IT-8**, 1962, pp 179-187.
14. ROSENFELD, A. and PFALTZ, J.L. Sequential operations in digital picture processing, *J Association for Computing Machinery*, 1966, **13**, pp 471-494.

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press

15. HSIA, T.C. A note on invariant moments in image processing, *IEEE Transactions on Systems, Man and Cybernetics*, 1981, **SMC-11**, (12), pp 831-834.
16. TEH, C.H. and CHIN, R.T. On digital approximation of moment invariants, *Computer Vision, Graphics, and Image Processing*, 1986, **33**, pp 318-326.
17. HARALICK, R.M. and SHAPIRO, L.G. *Computer and Robot Vision*, 1992, Addison-Wesley Longman Publishing Co, Boston, USA.
18. HEIKKILÄ, J. and SILVÉN, O. A four-step camera calibration procedure with implicit image correction, 1997, *IEEE Computer Vision and Pattern Recognition*, Puerto Rico, pp 1106-1112.
19. BOUGUET, J.Y. Camera Calibration Toolbox for Matlab, URL: http://www.vision.caltech.edu/bouguetj/calib_doc/ [cited 14 April 2006].

https://doi.org/10.1017/S0001924000007922 Published online by Cambridge University Press