Implementation of vision-based automatic guidance system on a fixed-wing unmanned aerial vehicle

C-S Lee and F-B Hsiao
fbhsiao@mail.ncku.edu.tw
Institute of Aeronautics and Astronautics
National Cheng Kung University
Tainan City, Taiwan

ABSTRACT

This paper presents the design and implementation of a vision-based automatic guidance system on a fixed-wing unmanned aerial vehicle (UAV). The system utilises a low-cost ordinary video camera and simple but efficient image processing techniques widely used in computer-vision technology. The paper focuses on the identification and extraction of geographical tracks such as rivers, coastlines, and roads from real-time aerial images. The image processing algorithm primarily uses colour properties to isolate the geographical track of interest from its background. Hough transform is eventually used to curve-fit the profile of the track which yields a reference line on the image plane. A guidance algorithm is then derived based on this information. In order to test the vision-based automatic guidance system in the laboratory without actually flying the UAV, a hardware-in-the-loop simulation system is developed. Description regarding the system and significant simulation result are presented in the paper. Finally, an actual test flight where the UAV successfully follows a stretch of a river under automatic vision-based guidance is also presented and discussed.

Paper No. 3761. Manuscript received 3 June 2011, revised version received 1 December 2011, accepted 7 March 2012.

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

NOMENCLATURE

A	matrix representing the image
B	matrix representing the structuring element
d	cross-track deviation distance
$f(i,j)$	value of input image at co-ordinates (i, j)
G_i,G_j	convolution results in i and j direction
G,ξ	edge's magnitude and direction
$g(i,j)$	value of input image at co-ordinates (i, j)
H,S,I	integers representing colour components of hue, saturation and intensity
h	height of image in pixels
i,j	horizontal and vertical co-ordinates of image space
m	variable in guidance algorithm
w	width of image in pixels
R,G,B	integers representing colour components of red, green and blue
T	threshold values in image segmentation
η	parameter in colour space conversion
θ,ρ	parameters describing a straight line in Hough space
ψ	aircraft's heading

Abbreviations

GPS	Global Positioning System
HIL	hardware-in-the-loop
HSI	hue-saturation-intensity
OBC	onboard computer
RGB	red-green-blue
UAV	unmanned aerial vehicle
UDP	user datagram protocol
VGS	vision-based automatic guidance system

1.0 INTRODUCTION

The image processing technique has progressed at an astounding rate over recent years due in part to the equally impressive rapid advancement of the digital computing technology. This has enabled the increasingly widespread utilisation and application of computer-vision based technology on small fixed-wing unmanned aerial vehicles (UAVs). In general, there is a growing number of research works that explore the application of image processing on various areas pertaining to UAV including attitude control[1], autonomous landing[2], ground object tracking[3,4], formation flight[5], aerial-refuelling[6], navigation and guidance[7], etc.

This paper focuses on the utilisation of computer vision and image processing as an alternative means of navigation and guidance for fixed-wing UAVs, especially in identifying and following geographical tracks such as rivers, coastlines, roads, etc. Since the advent of global positioning system (GPS), it has quickly become one of the most dominant means of *en route* navigation and guidance aid for all sorts of transportations. It is especially essential for small UAVs which have stringent restrictions in weight and space because only a small and compact GPS receiver is needed to acquire global positioning and velocity data. However, GPS is not without its limitations. In order

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

to accurately calculate the positioning solution, the GPS receiver must maintain a constant line-of-sight connection with multiple GPS satellites orbiting the Earth. In some occasions, the signal link can be rather vulnerable. It is not uncommon for GPS signals to be lost, either by perpetuated jamming or simply being in magnetically hazardous environments. If GPS is the sole navigation device onboard an UAV, a prolonged signal outage condition may jeopardise the safety of the UAV to a dangerous extent.

In addition, GPS-based guidance typically requires predefined waypoints which are determined using the latest available electronic map. If a geographical track such as a winding river is to be followed, a dense distribution of waypoints might be needed for the irregular features of the river to be followed closely. Moreover, the actual terrain may change over time due to natural elements such as typhoon, flood, erosion or human development activities. If the electronic map used to define the waypoints is not up-to-date, the resulting flight path of the UAV may deviate significantly from the desired one. Vision-based detection and guidance system essentially provides a real-time situational awareness to the UAV. One of the earlier works reported on this topic demonstrated a vision-based road-following capability on a small autonomous aircraft[8]. In particular, the road detection algorithm is based on a probabilistic approach where a so-called Bayesian pixel classification is used to separate road pixels from non-road pixels. Then, the position and the orientation of the centreline of the road are computed using line-fitting methods. The authors further use the information of the road detection to devise control strategies which include altitude-hold and lateral control that allow the UAV to follow the road.

Another interesting work addressed the problem of searching and mapping a river autonomously using an UAV based on visual feedback[9]. In essence, an off-line learning algorithm and a real-time detection algorithm that can identify the river in a given image were developed based on image-processing techniques. The exploration system transitions automatically between GPS-based tracking and vision-based tracking in an attempt to identify the river in a given area and map its co-ordinates along a specified river direction. Furthermore, the paper is substantiated by successful flight experiments. In the mean time, more recent work includes a near-infrared based vision system that distinguishes water from land and issues commands to UAV to follow the coastline or the riverbank[10]. The paper also devises a guidance algorithm based on the identified contour of the coastline. Test flights were conducted and it is shown that the UAV is able to follow the irregular coastline in real world condition. On the other hand, Zhang *et al*[11] reported an interesting approach to UAV's position and orientation estimation using vision-based system. Essentially the position and orientation estimation problem is formulated as a tracking problem solved by using an extended Kalman filter, whose state and observation models are established based on an analysis of the imaging geometry of the UAV's onboard video camera in connection with a digital elevation map of the area of flight. Simulation results are encouraging but the method incurs huge computational load and an updated digital elevation map of the flight area must be available to produce good result.

In light of the above discussion, this paper presents an attempt to develop a vision-based automatic guidance system (VGS) that enables a fixed-wing UAV to identify and guide itself along a geographical track, namely a river, coastline, road, etc, using only the natural feature of the terrain without radio-based sensors. The VGS would be integrated into the Spoonbill UAV system[12] and acts as a supplemental navigation aid to the existing GPS-based navigation and guidance system. One of the biggest challenges is that the image processing algorithm and the associated system hardware have to be simple and efficient enough so that they can be implemented effectively on the Spoonbill UAV which has tight limitation in terms of weight, space, computing resources and power consumption. The remaining parts of this paper are organised as follows: First, the image

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

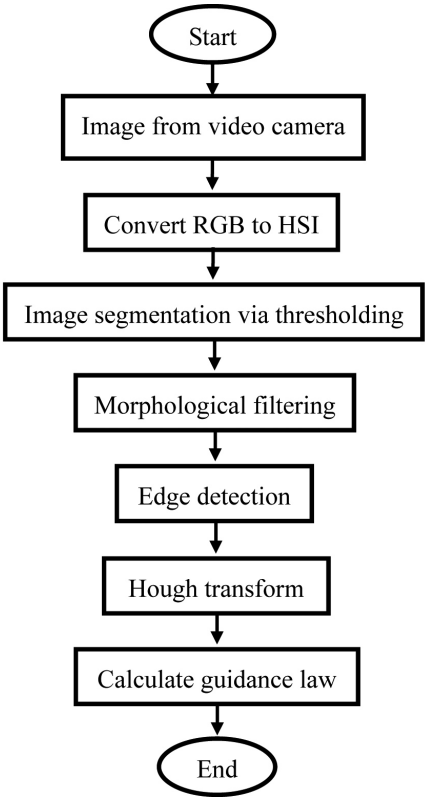


Figure 1. Flow chart of image processing procedures.

processing algorithm used to identify the geographical track is detailed. Subsequently, the hardware-in-the-loop simulation system built to test the image processing algorithm is described along with the simulation result. The next section further presents the test flight conducted in attempt to validate the VGS in real world operation. Finally, the paper is ended with some concluding remarks.

2.0 METHODOLOGY

Figure 1 depicts the proposed image processing procedures which provide automatic guidance for the UAV in terms of the reference heading based on real-time aerial images captured by the onboard video camera. First, all pixels in the digital image are converted into the HSI (hue-saturation-intensity) colour space from the typical RGB (red-green-blue) colour space. Then by selecting appropriate thresholds in terms of hue and intensity values of each individual pixel, the object of interest, e.g. the sea or the river in the image, can be segregated from the background. The thresholding process results in a binary image where the object is depicted in white while the background in black. Next, by applying a morphological filter which features the erosion and dilation techniques, the unwanted 'noise' in the binary image is cleaned up thus yielding better quality image. At this stage, the profile of the object (in white) should be clearly distinguishable from the

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

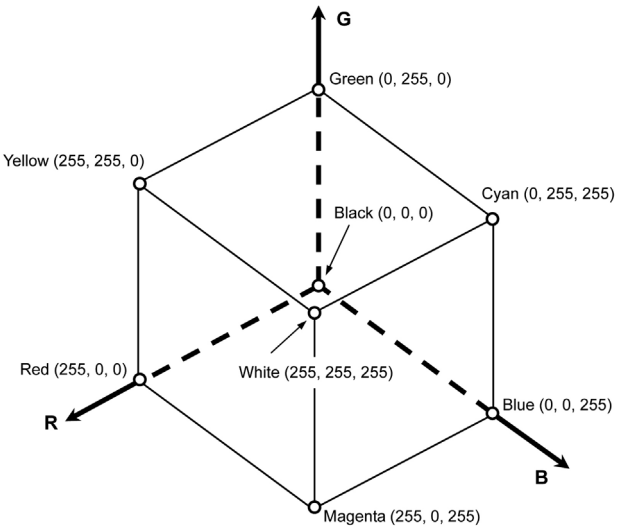


Figure 2. The RGB cube in 24-bit representation where colour values for each pixel are encoded in an RGB triplet (*R*,*G*,*B*) which represents the three-dimensional co-ordinate of the point of any given colour within the cube.

background (in black). Thus by detecting the edge of the object, the desired geographical track of the coastline or the river may be extracted from the image.

In order to obtain a straight line that fits to the geographical track which is composed of sets of edge points (pixels), an algorithm called the Hough transform is implemented. Essentially, the Hough transform detects and formulates the equation of a straight line via a voting procedure in the parametric space. Finally, since the onboard video camera is fixed on the fuselage of the UAV, there's a known relationship between the UAV's heading and the straight line obtained previously. Hence, a guidance law can be devised easily.

2.1 Colour processing

By using a video adapter (or video card), the analog signal from the video camera is captured by the onboard computer in the digital RGB format. Alternatively, a digital video camera could be used where the output would be in digital form directly. However, an analog video camera is used in the present work. The RGB colour model is an additive colour model in which red (*R*), green (*G*), and blue (*B*) light are combined together in a variety of ways to reproduce a broad array of colours. Figure 2 shows the RGB cube which portrays all the available colours — whose three axes represent the *R*, *G* and *B* weights. The RGB component values are often stored as integer numbers in the range of 0 to 255, the range that a single eight-bit byte can offer. Some distinct colours which lie at the edges of the RGB cube are illustrated in Fig. 2 as well. The RGB model is widely used in the sensing, representation, and display of images in electronic systems, such as televisions and computers. However, it is device-dependent i.e. different devices detect or reproduce a given RGB value differently. This is because the colour elements (such as phosphors or dyes) and their response to individual *R*, *G* and *B* levels vary from manufacturer to manufacturer. Moreover, another difficulty with RGB model is that the information of brightness and saturation are embedded inside the model. A standard method for dealing with this issue is to apply a nonlinear transformation on the RGB space which is linear in nature. In particular, for the purpose of our application in the vision-based

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

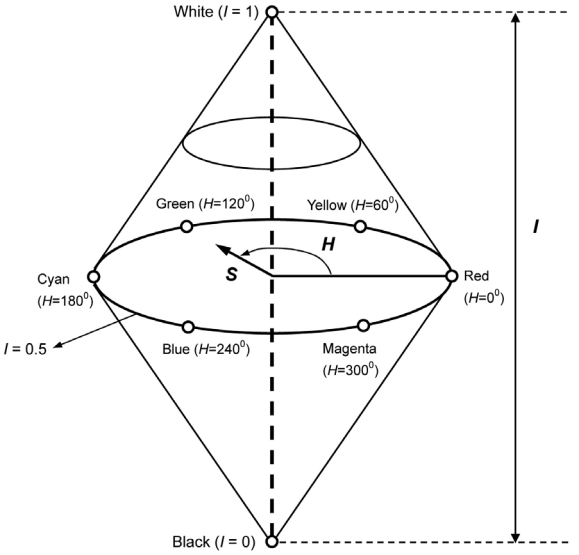


Figure 3. The HSI colour space.

guidance system, the colour values are transformed into the HSI colour space which represents colours by three properties: hue (H), saturation (S) and intensity (I). By using normalised RGB values, the corresponding HSI values can be computed via the following[13]:

$$H = \begin{cases} \eta & \text{if} \quad B \le G \\ 360 - \eta & \text{otherwise} \end{cases} \qquad \ldots (1)$$

$$S = 1 - \frac{3}{(R+G+B)}\left[\min(R,G,B)\right] \qquad \ldots (2)$$

$$I = \frac{1}{3}(R+G+B) \qquad \ldots (3)$$

where

$$\eta = \text{Cos}^{-1}\left\{\frac{\frac{1}{2}\left[(R-G)+(R-B)\right]}{\left[(R-G)^2+(R-B)(G-B)\right]^{1/2}}\right\} \qquad \ldots (4)$$

The hue component describes the colours in the form of an angle between 0° to 360° which represents colours passing from red to green, to blue and back to red again as shown in Fig. 3. The saturation component indicates how much the colour is polluted with the white colour, i.e. the property of colour that varies in passing from red to pink. On the other hand, the intensity component describes the property of colour that varies in passing from black to white, i.e. it depicts the 'brightness' of the colour. Both the saturation and intensity values range from 0 to 1. As illustrated in Fig. 3, hue is more meaningful when saturation approaches 1 and it is less meaningful when saturation approaches 0 (where all colours will be essentially white). The hue values would also be meaningless when intensity approaches 0 or 1. Furthermore, intensity also limits the saturation values. By transforming the image from the RGB colour space into the HSI space, we may indentify a particular colour of interest in terms of hue regardless of the environmental lighting

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

condition associate with the acquisition of the image. This feature enables the segregation of the object from the background based on distinct colour difference between them to be executed more effectively.

2.2 Image segmentation via thresholding

The primary task of image processing in the present work is to 'recognise' the geographical tracks in the aerial image such as rivers or coastlines, and isolate them from the background in an automated means. This may be accomplished by performing image segmentation where all pixels of the image are partitioned into two segments. In particular, the so-called thresholding method is utilised in the present work. In its simplest form, individual pixels are marked as "object" pixels if their property value is greater than some threshold value and as 'background' pixels otherwise as described by the following:

$$g(i,j)=\begin{cases}1 & if \quad f(i,j)>T\\ 0 & otherwise\end{cases} \quad \ldots (5)$$

where f is the property value of the input pixel, g is the output pixel, T is the selected threshold and (i, j) denotes the co-ordinates of the pixel in the image frame. Note that the thresholding process yields a binary image where the object is in white colour. A better variant of the thresholding criterion is

$$g(i,j)=\begin{cases}1 & if \quad T_{min}\le f(i,j)\le T_{max}\\ 0 & otherwise\end{cases} \quad \ldots (6)$$

where a pixel is labelled as 'object' if its value resides between two thresholds $[T_{min}, T_{max}]$. This enables the extraction of the object of interest with better precision, especially if the object has a colour property that is distinctly different from its background.

Figure 4(b) shows the result of the image segmentation via thresholding in terms of hue and intensity properties. Note that the original image is depicted in Fig. 4(a). The river shown in white has clearly been successfully extracted from the background. The thresholding method is the simplest method of image segmentation techniques. Its simplicity leads to a computationally efficient algorithm which may be executed effectively using the limited computing power onboard the UAV.

2.3 Morphological filtering

The next step in the image processing procedure involves removing the residual 'noise' from the previous image segmentation process as shown in Fig. 4(b) by the smudges of white pixels scattering over the image. These unwanted noisy pixels can be filtered through the application of morphological operations, namely erosion followed by dilation on the image. In formal mathematics under the context of the set theory, the erosion of image $\mathbf{A}$ by a structuring element $\mathbf{B}$, denoted by $\mathbf{A}\ominus\mathbf{B}$ is defined by[14]

$$\mathbf{A}\ominus\mathbf{B}=\left\{x\,|\,(\mathbf{B})_x\subseteq\mathbf{A}\right\} \quad \ldots (7)$$

where x is the element contained in the resulting set after an erosion operation. In practice, the erosion operation is performed by laying the structuring element on the image and sliding it across the image in a manner similar to convolution. The corresponding algorithm is described by the following sequence of steps[15]:

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

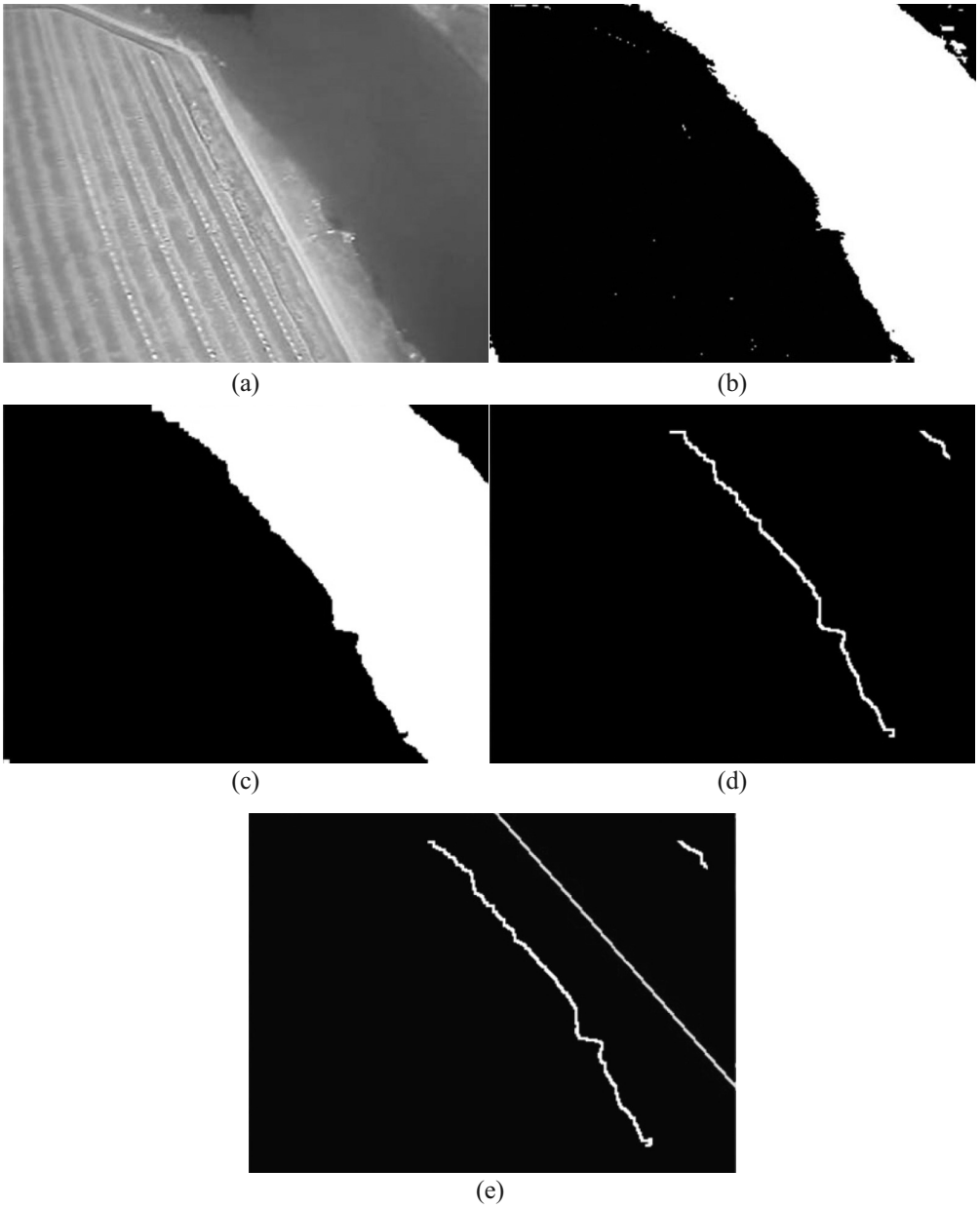


Figure 4. Image processing procedures. (a) Original aerial image. (b) Colour processing and image segmentation. (c) Morphological filtering. (d) Edge detection. (e) Hough transform.

1. If the origin of the structuring element coincides with a '0' in the image, no operation is performed; move to the next pixel.
2. If the origin of the structuring element coincides with a '1' in the image, and any of the '1' pixels in the structuring element extend beyond the object ('1' pixels) in the image, then change the '1' pixel in the image to a '0'.

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

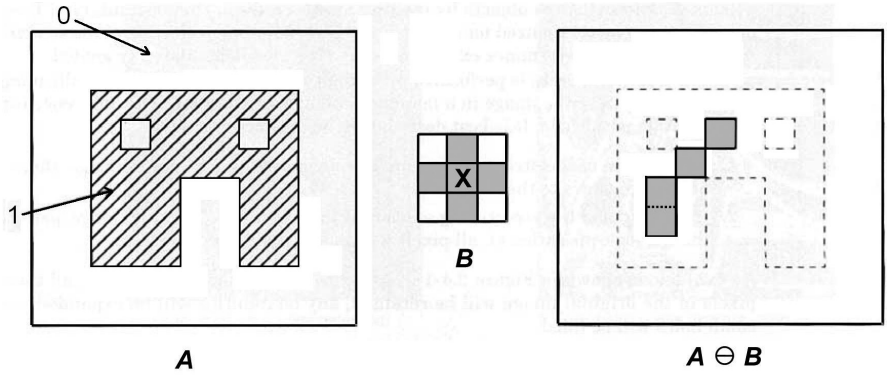


Figure 5. Erosion operation[15].

An example of the erosion operation is shown in Fig. 5. Note that the origin of the 3×3 pixels structuring element, **B** is depicted by the symbol 'X'. On the other hand, the dilation operation can be represented as a union of translates of the structuring element[14]:

$$\mathbf{A} \oplus \mathbf{B} = \bigcup_{a \in A} (\mathbf{B})_a \quad \ldots (8)$$

The dilation operation is similar to erosion, but pixels are turned into '1', instead of '0'. In practice, the following algorithm is employed[15]:

1. If the origin of the structuring element coincides with a '0' in the image, no operation is performed; move to the next pixel.
2. If the origin of the structuring element coincides with a '1' in the image, performed the OR logic operation on all pixels within the structuring element.

Figure 6 shows an example of the dilation operation. In essence, the erosion works by 'shrinking' the image while the dilation 'fills' and 'expands' it. If an erosion is followed directly by a dilation, all pixels in regions that are too small to contain the structuring element are thus eliminated while the prominent object in the image would be largely untouched. This constitutes a type of morphological filtering called opening[15]. In this case, the structuring element is often called a probe, because it is probing the image looking for small objects to be filtered out of the image. The filtered image of the shown in Fig. 4(c) clearly exhibit better quality compared to the Fig. 4(b).

2.4 Edge detection

At this stage of the image processing, the object of interest such as the river has been successfully isolated from the background into a binary image as shown in Fig. 4(c). Next, the edge demarcating the river and the background is identified using an edge detection operator. Typically, edge detection operators are based on the idea that the edge information in an image is found by looking at the relationship a pixel has with its neighbours[15]. If a pixel has neighbors with widely varying intensity levels, it may represent an edge point. Therefore, an edge can be detected by calculating the gradient of the image intensity at each point, giving the direction of the largest possible increase from light to dark and the rate of change in that direction. In the present work, the Sobel operator

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

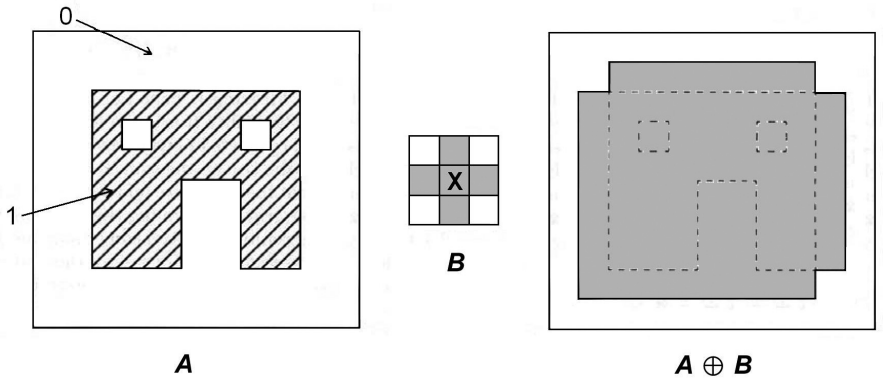


Figure 6. Dilation operation[15].

is used. It uses two 3×3 kernels which are convolved with the original image, **A** to calculate approximations of the gradient in both the horizontal axis, i and vertical axis, j:

$$G_j = \begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * \mathbf{A}; G_i = \begin{bmatrix} -1 & 0 & 1 \\ 2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * \mathbf{A} \quad \ldots (9)$$

where * denotes two-dimensional convolution operation and G_i, G_j are the convolution results along the i and j direction respectively. Consequently, the edge magnitude, G and direction, ξ can be computed via

$$G = \sqrt{G_i^2 + G_j^2}; \xi = \mathrm{Tan}^{-1}\left(\frac{G_j}{G_i}\right) \quad \ldots (10)$$

The result of edge detection is shown in Fig. 4(d) where the profile of the river is clearly distinguishable from the image. Now we may proceed to generate a reference line that conforms to the river which the UAV is supposed to follow.

2.5 Hough transform

The Hough transform is an algorithm that will take a collection of edge points, as found by the edge detector described in previous section, and determine all the lines which these edge points lie[15]. This is done by transforming the line of interest from the image space (i-j plane) into a parameter space called the Hough space via the following equation:

$$\rho = j\mathrm{Cos}\theta + i\mathrm{Sin}\theta \quad \ldots (11)$$

Now the line is defined by ρ, the perpendicular distance from the origin to the line, and θ, the angle between the j-axis and the ρ-line (See Fig. 7). In other words, each pair of ρ and θ represents a particular line on the image. Subsequently, in order to reduce the search time, the ρ-θ parameter space can be quantised into specific number of blocks as shown in Fig. 7. The sizes of the increments for each axis are denoted as $\Delta\rho$ and $\Delta\theta$, respectively, and they corresponds to the resolution of the quantisation of the line. Note that the range of θ is ± 90 and ρ ranges from 0 to

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

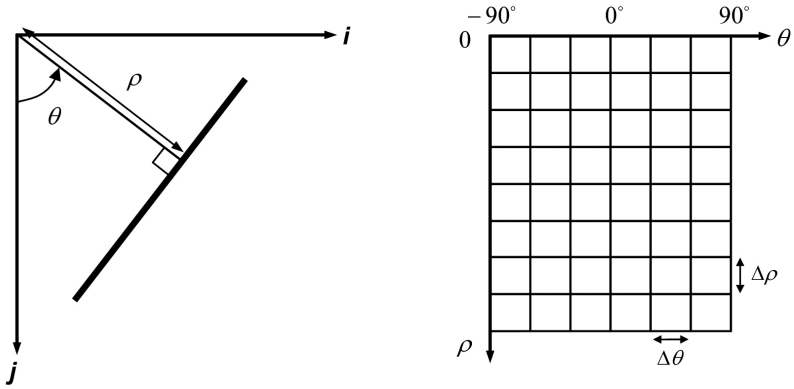


Figure 7. Hough transform from image space (left) to parameter space (right).

$\sqrt{h^2+(w/2)^2}$ where h and w are the height and width of the image in terms of pixels, respectively. In order to fit a line on the edge points shown in Fig. 4(d), the following algorithm is employed:

1. Define the desired increments $\Delta\rho$ and $\Delta\theta$, and quantise the ρ-θ space accordingly.
2. For a particular edge point, insert the respective i, j values into the equation $\rho = j\text{Cos}\theta + i\text{Sin}\theta$. Then, for each value of θ in the quantised space (from –90° to 90°), solve for ρ.
3. Repeat Step 2 for every edge point in the image and record the counts of each quantised block that has been 'inserted'.
4. The quantised block (ρ-θ pair) with maximum number of counts corresponds to a line that is most correlated with the edge points.

The line defined by that particular pair of ρ and θ represents the dominant line on the image of the edge. The result of Hough transform is shown in Fig. 4(e) where the straight yellow line represents the ρ-θ pair with the highest counts in the parameter space. Recall that the objective of the image processing in this work is to generate an automatic guidance law for the automatic flight control system to follow. By tracking the reference line, it is possible for the UAV navigate itself along the river or any other geographical track that is identifiable using the image processing algorithm.

2.6 Calculation of guidance law

After the image processing procedure has successfully identified the geographical track of interest, a simple guidance law may be devised for the UAV's automatic flight control system. Essentially, the guidance law provides the reference heading, ψ_{ref} which the UAV is supposed to follow. The result of Hough transform gives us ample information to formulate the calculation of the reference heading as shown in Fig. 8. With the known value of θ, the heading of the identified track, ψ_{track} is obtained via

$$\psi_{track} = \begin{cases} 90^\circ - \theta, & \theta \geq 0^\circ \\ -90^\circ - \theta, & \theta < 0^\circ \end{cases} \quad \ldots (12)$$

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

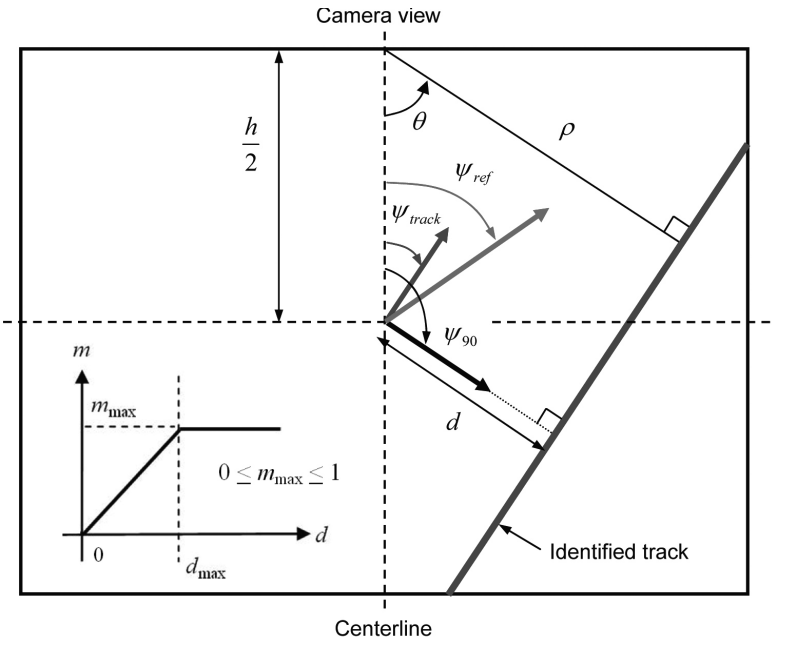


Figure 8. Guidance law.

In addition, the cross-track deviation, d of the UAV and the track can also be computed using the known ρ value by

$$d = \rho - \frac{h}{2}\mathrm{Cos}\theta \qquad \ldots (13)$$

where h is the height of the image therefore d is given in the unit of pixel. Generally, the ultimate goal of the guidance law is to bring the UAV's heading to approach ψ_{track} while driving d to zero. This can be achieved with the following reference heading

$$\psi_{ref} = (1-m)\cdot\psi_{track} + m\psi_{90} \qquad \ldots (14)$$

where

$$\psi_{90} = \begin{cases} \psi_{track} + 90^{\circ}, & \psi_{track} \geq 0^{\circ} \\ \psi_{track} - 90^{\circ} & \psi_{track} < 0^{\circ} \end{cases} \qquad \ldots (15)$$

Note that m is a parameter related to the cross-track deviation, d and the plot shown in Fig. 8 is one used in practice. The proper values of m_{max} and d_{max} can be determined through simulations and flight tests. It is easy to realise that when $d = 0$, $\psi_{ref} = \psi_{track}$ and the UAV would be asked to follow the heading of the track. Also, since the vertical centreline of the camera view coincides with the UAV's heading, the reference heading computed via the above formulation is relative to the UAV's heading itself. Eventually, when the UAV is aligned with the track (a successful guidance), the line representing the track in the image would coincide with the vertical centreline where $\psi_{ref} = \psi_{track} = 0^{\circ}$.

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

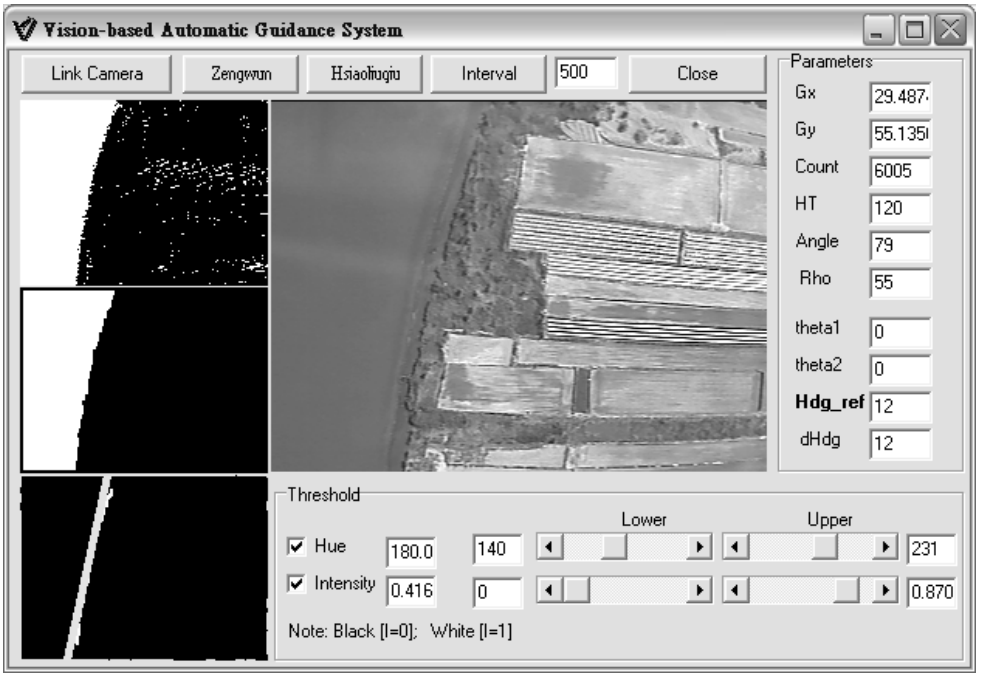


Figure 9. Snapshot of the actual program that executes the image processing algorithm and computes the guidance law.

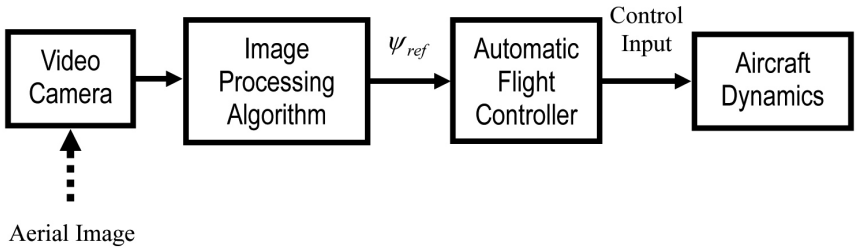


Figure 10. Block diagram of the vision-based automatic guidance system.

In practice, the image processing algorithm is implemented on the Spoonbill UAV system using a graphical computer program composed using Borland C++ Builder (see Fig. 9). Basically, this program acquires aerial image from the onboard video camera, executes the image processing algorithm and calculates the reference heading. Subsequently, this guidance information would be fed to the existing automatic flight control system of the UAV at a rate of 2Hz. It is noteworthy that the vision-based guidance system provides the outer-loop control which is independent from the inner-loop control and stabilisation which may be composed of any automatic flight controller (see Fig. 10). In fact the guidance law devised previously is quite similar to the existing GPS-based guidance law on the Spoonbill UAV system[12].

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

3.0 HARDWARE-IN-THE-LOOP SIMULATION SYSTEM

Before the vision-based automatic guidance algorithm is validated in actual test flights, simulations were conducted in the laboratory to minimise the risk associated with the actual deployment of the proposed guidance law in automatic flight. In order to achieve higher integrity and fidelity in simulation results, a hardware-in-the-loop (HIL) simulation system is developed exclusively for the Spoonbill UAV system.

3.1 System architecture and setup

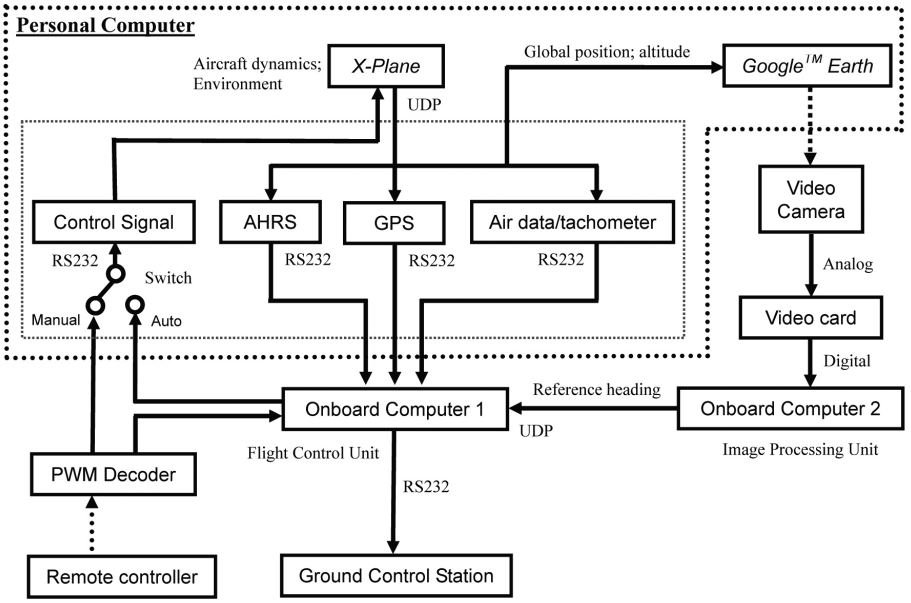
There are two onboard computers (OBCs) on the Spoonbill UAV; one of them (OBC 1) is responsible for the sensor-data acquisition and flight control tasks while the other (OBC 2) executes the vision-based guidance algorithm. Both OBCs (PC/104 embedded-PC type computer modules) are incorporated into the simulation loop so that the simulation resembles the UAV in flight to a higher degree. In the mean time, a commercially available flight simulator, X-Plane by Laminar Research is used to provide the aircraft dynamics and a three-dimensional virtual flight environment. The software allows the user to design and build customised air vehicle models and incorporate them into the simulation environment. Hence, a full-scale Spoonbill model is constructed with the same physical properties as the actual air vehicle. Furthermore, through a UDP (User Datagram Protocol) link, we can establish a real-time interaction with the virtual UAV by extracting flight data from and transmitting control signals to it. As shown in Fig. 11, all flight data acquired from X-Plane are sent to OBC 1 in data packages and channels that are exactly the same with the actual onboard system. Similar to the actual scenario, the virtual UAV can be flown manually or automatically by flipping a knob on the remote controller.

On the other hand, the HIL system also permits the testing of the vision-based automatic guidance system in the comfort of a laboratory. However, although the X-Plane software provides a three-dimensional terrain environment, the resolution is rather coarse. Therefore, a data-link is established between the Google™ Earth software and the flight simulator where the position (longitude and latitude) and altitude of the UAV is transmitted to the global mapping software at some update rate (see Fig. 11). The Google™ Earth software displays satellite image which correspond to the updated position and altitude of the UAV. By pointing a video camera at the image, we are able to simulate the scenario of the aerial image being captured by the onboard video camera in real flight. The image is acquired by OBC 2 via a video card where it would undergo image processing algorithm. Eventually, the guidance information is sent to OBC 1 through an UDP link. It is noteworthy that since both X-Plane and Google™ Earth provide global positioning and mapping information, the simulation of the vision-based guidance algorithm can be done on any actual geographical location on the Earth.

3.2 Simulation result

In conjunction with the planned actual flight test, the HIL simulation of the vision-based automatic guidance system (VGS) is performed along a portion of the Zengwun River, which is located in the outskirt of Tainan City, in the southern part of Taiwan. The yellow line in Fig. 12 shows the trajectory of the UAV navigating itself along the Zengwun River under VGS. Note that the inner-loop flight control and stabilisation of the UAV is provided by the existing linear-quadratic-Gaussian controllers[16] while the altitude and airspeed are maintained at approximately 300m and 100km/h respectively. In actual test flights of the Spoonbill UAV, the flying altitude is typically restricted below 305m for safety and regulatory reasons. Figure 12 further

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press



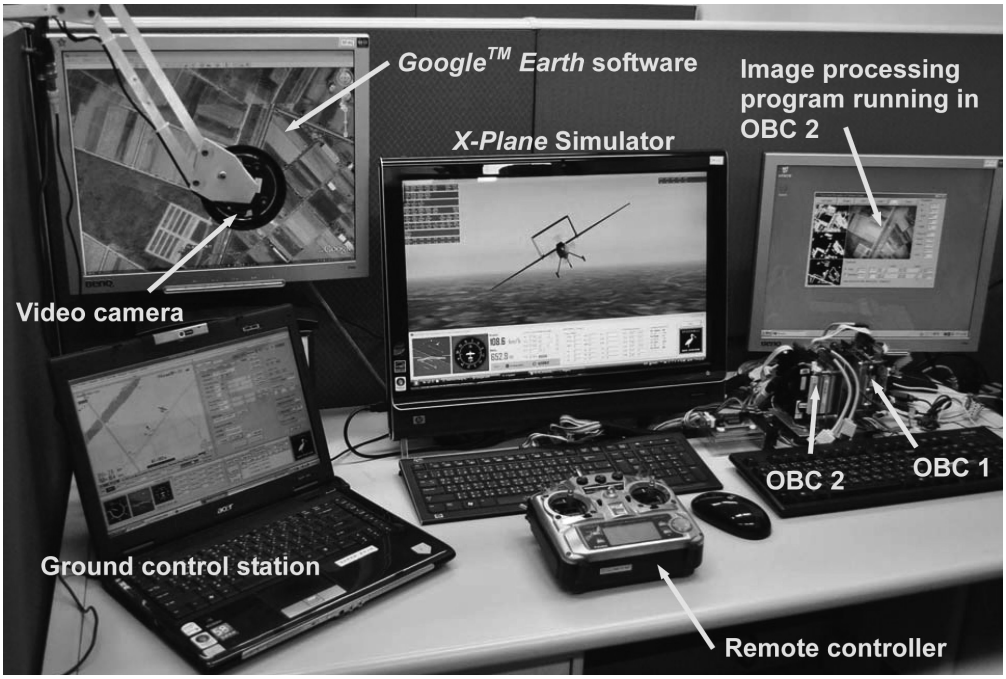


Figure 11. Hardware-in-the-loop simulation system.

illustrates that the UAV essentially follows the south-side bank of Zengwun River because the camera's field-of-view cannot cover the entire river at that particular flying altitude. The result shows that the VGS works effectively and the UAV continues to fly along the riverbank over a ground track of approximately 11·7km without significant deviation from the river.

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

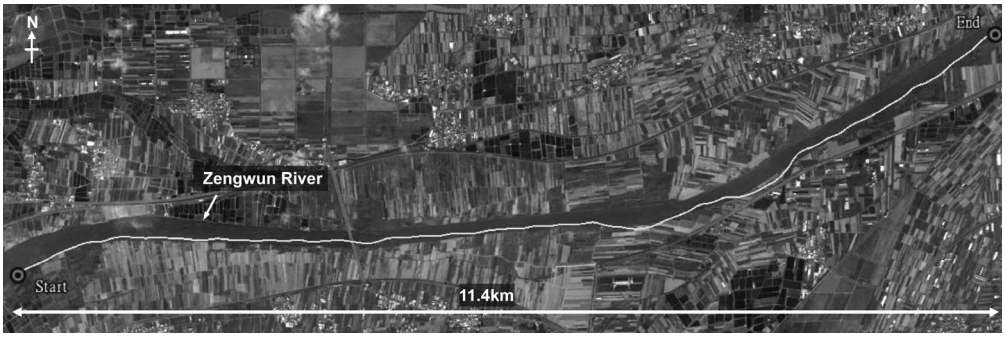


Figure 12. Hardware-in-the-loop simulation result showing trajectory of the UAV flying along a long portion of Zengwun River using vision-based automatic guidance system.



Figure 13. Spoonbill UAV with a gimbaled video camera under the fuselage.

4.0 FLIGHT TEST RESULT

The Spoonbill UAV used in this work is shown in Fig. 13 and detail description regarding the airframe and onboard system can be found in Lee *et al*[12]. The aircraft weighs 30kg and carries two onboard computers. Two onboard computers are required in practice because one computer is not powerful enough to run the image processing and flight control calculations simultaneously. Also discernible in Fig. 13 is a gimbaled video camera carried under the fuselage. The video camera used in this work is a commonly available low-cost analogue camera with horizontal resolution of 420 TV lines. In addition, its lens has field-of-view of 70deg, focal length of 3·6mm and *f*-number of 2·0. It is noteworthy that the analog video signal is digitised into a 320 × 240 pixels frame for subsequent image processing in the VGS. As shown in Fig. 14, the video camera is

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

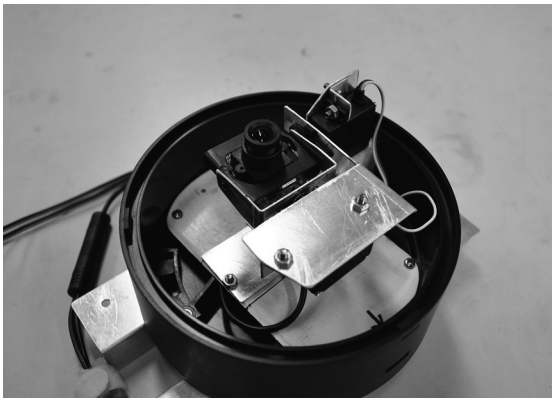


Figure 14. Video camera mounted on a two-axis gimbal with the transparent dome removed.

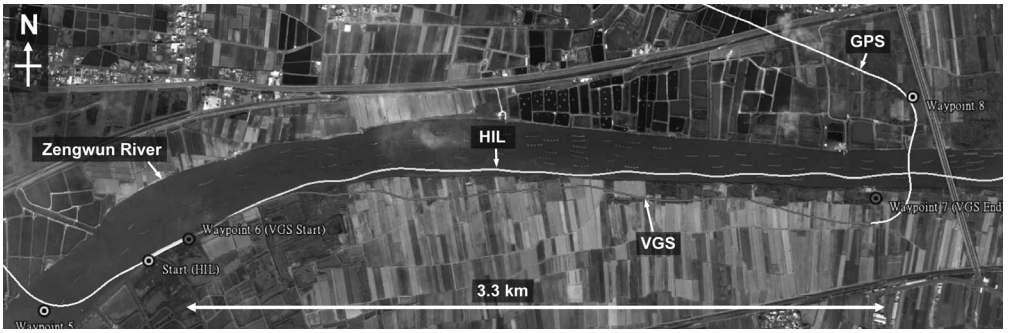


Figure 15. Trajectories of the Spoonbill UAV in actual test flight (red) and in HIL simulation (yellow) while flying under VGS.

mounted on a two-axis rotating-platform which permits the camera to rotate along the pitch and roll axes relative to the aircraft body-frame co-ordinates system. Hence, through a controller with aircraft attitude feedback, the video camera is able to maintain a constant orientation regardless of the dynamic motion of the aircraft, e.g. directly points towards the ground surface. This feature is imperative for the proposed image processing algorithm of the VGS to identify the geographical track effectively.

After numerous successful tests in the HIL simulation system, a series of flight tests was conducted in order to validate the effectiveness of the vision-based automatic guidance system (VGS) in real environment. The test was planned to be performed along a portion of Zengwun River because it is close to our usual flight test site and more importantly, it is within the flying range of the Spoonbill UAV. Nevertheless, the river is approximately 8km to the south from the takeoff/landing site. Therefore, the UAV is preprogrammed to fly automatically towards the river using GPS (Global Positioning System) guidance following a flight path connected by preselected waypoints. When the UAV reaches waypoint 6 (see Fig. 15), the VGS would be automatically engaged and take over the outer-loop control of the flight control system. As the UAV flies along the riverbank and reaches waypoint 7, the VGS would be disengaged automatically and the GPS guidance would resume. Then the UAV would follow the remaining waypoints and return to the takeoff/landing area.

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

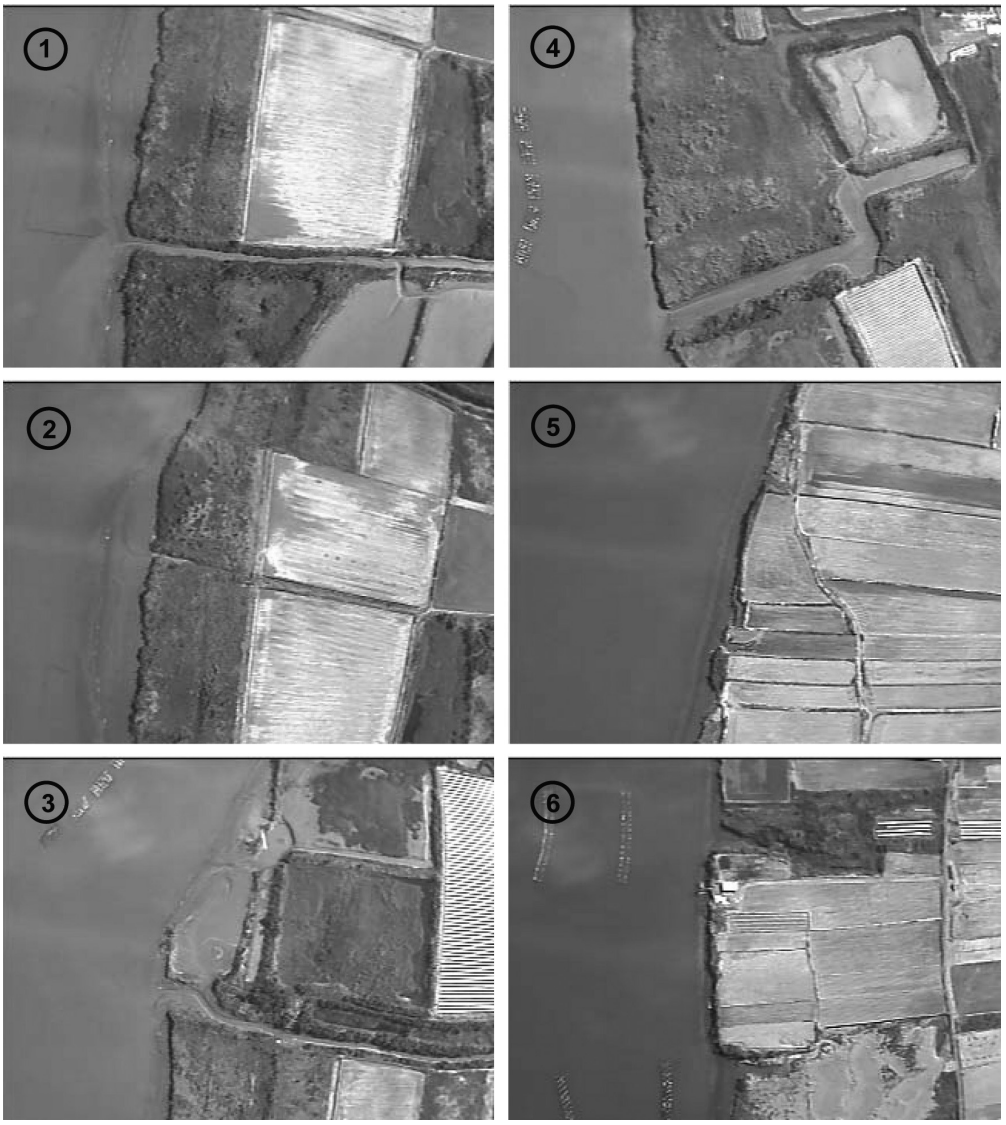


Figure 16. Sample images captured by the onboard video camera during the VGS flight along the south-side riverbank of Zengwun River (Number on each image marks the temporal sequence of the images).

Figure 15 shows the result of the test flight conducted in the morning of 29 October 2010. The trajectory of the UAV during VGS flight is depicted in red while the white line represents the flight path flown under GPS guidance. Note that the UAV was flying in an anticlockwise direction. The entire flight lasted for 12min 40sec. while the nominal altitude and airspeed of the UAV were 306m and 109km/h, respectively. In addition, the HIL simulation result previously shown in Fig. 12 is also embedded into Fig. 15 for comparison (shown in yellow line). Figure 15 shows that there is a noticeable discrepancy between the trajectory of simulation and actual flight. Also, the latter did

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

not seem to follow the riverbank closely. This is in fact caused by the difference between the satellite images provided by the Google™ Earth software and the actual terrain. In particular, the mapping images in the version of Google™ Earth used in this work are dated to the year 2008. The onboard video images confirmed that the UAV was indeed navigating itself along the riverbank such as those shown in Fig. 16. Apparently, a significant portion of the riverbank has been eroded over the years but it is not shown on the map. As soon as the UAV reached within 200m from waypoint 7 (pass-through criterion), the VGS disengaged automatically and the UAV started to head back to the takeoff/landing site using GPS guidance. The successful test flight verified that the proposed VGS is feasible and the consistency it has with the HIL simulation result shows that the HIL simulation system is a useful and practical research-and-development tool.

5.0 CONCLUDING REMARKS

This paper presented the development of the vision-based automatic guidance system (VGS) for a relatively small fixed-wing UAV. In essence, the system utilises a low-cost ordinary video camera and efficient image processing techniques to identify a geographical track of interest, e.g. river, coastline, road et cetera from real-time aerial images. Furthermore, a guidance algorithm is derived by using information obtained from the identified track thus allowing an independent automatic flight control system navigate the UAV along the geographical track without radio-based navigation devices. In addition, a hardware-in-the-loop simulation system was exclusively built to test the VGS in the laboratory without actually flying the UAV. By incorporating the same onboard computers used in the actual Spoonbill UAV system, the hardware-in-the-loop simulation is found to be very useful and realistic in testing and improvising the image processing algorithm. This has led to subsequent successful test flights which verified that the VGS is capable of performing up to expectation in real-world environment.

The current image processing algorithm is, however, not without its limitations. One of the most important assumptions is that the geographical track of interest must reside within the onboard camera's field-of-view at all time. If the track is completely out of the aerial images, the image processing algorithm would not be able to identify it and that would lead to the total failure of the guidance system. Another potential problem may arise due to the glare from sunlight reflecting from water surface, the current method is not capable of handling this scenario. On the other hand, since the image processing algorithm essentially uses colour to separate the object from its background, a distinctive difference in colour property between the geographical track and its background is needed to produce good result. Overall, although this system is not capable of completely replacing GPS as the primary navigation and guidance device for UAVs, it offers a supplemental navigational aid that is not susceptible to signal lost condition which is not uncommon for GPS. More importantly, this paper showed that the proposed image processing algorithm is simple and efficient enough to be effectively implemented on relatively small fixed-wing UAVs which typically have limited computing resources.

ACKNOWLEDGEMENT

This work is supported in part by the National Cheng Kung University, Taiwan, ROC under grant number, D98-3340 and D99-3340, under the Ministry of Education's 5Y50B Special Funding, and by National Science Council of Taiwan under the contract number, NSC 100-2221-E-006-099-MY3.

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press

REFERENCES

1. ETTINGER, S.M., NECHYBA, M.C., IFJU, P.G. and WASZAK, M. Vision-guided flight stability and control for micro air vehicles, 2002, IEEE International Conference on Intelligent Robots and Systems, IEEE, **3**, pp 2134–2140.
2. SCHULZ, H.W. Towards vision-based autonomous landing for small unmanned aerial vehicles:image processing hardware development, *J Aerospace Computing, Information and Communication*, 2008, **5**, (10), pp 380-395.
3. MORGAN, Q., GRIFFITHS, S. and BEARD R.W. Target acquisition, localization and surveillance using a fixed-wing mini-UAV and gimbaled camera, 2005, IEEE International Conference on Robotics and Automation, Barcelona, Spain.
4. STEPANYAN, V. and HOVAKIMYAN, N. Visual tracking of a maneuvering target, *J Guidance, Control, and Dynamics*, 2008, **31**, (1), pp 66-80.
5. 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, SC, USA.
6. KIMMETT, J., VALASEK, J. and JUNKINS, J.L. Autonomous aerial refueling utilizing a vision based navigation system, 2002, AIAA Guidance, Navigation, and Control Conference and Exhibition, Monterey, CA, USA.
7. JOHNSON, E.N., CALISE, A.J., WATANABE, Y., HA, J. and NEIDHOEFER, J.C. Real-time vision-based relative aircraft navigation, *J Aerospace Computing, Information and Communication*, 2007, **4**, (4), pp 707-738.
8. FREW, E., MCGEE, T., KIM, Z., XIAO, X., JACKSON, S., MORIMOTO, S., RATHINAM, S., PADIAL, J. and SENGUPTA, R., Vision-based road following using a small autonomous aircraft, 2004, IEEE Aerospace Conference, Big Sky, MT, USA.
9. RATHINAM, S., ALMEIDA, P., KIM, Z., JACKSON, S., TINKA, A., GROSSMAN, W. and SENGUPTA, R. Autonomous searching and tracking of a river using an UAV, 2007, 2007 American Control Conference, Marriot Marquis Hotel, New York, USA.
10. BAKER, P. and KAMGAR-PARSI, B. Using shorelines for autonomous air vehicle guidance, *Computer Vision and Image Understanding*, 2010, **114**, pp 723-729.
11. ZHANG, J., WU, Y., LIU, W. and CHEN, X. Novel approach to position and orientation estimation in vision-based UAV navigation, *IEEE Transactions on Aerospace and Electronic Systems*, April 2010, **46**, (2), pp 687-700.
12. LEE, C.S., CHAN, W.L. and HSIAO, F.B. A linear-quadratic-Gaussian approach for automatic flight control of fixed-wing unmanned air vehicles, *Aeronaut J*, January 2011, **115**, (1163), pp 29-41.
13. GONZALEZ, R.C. and WOODS, R.E. *Digital Image Processing*, 2002, Prentice-Hall International.
14. HARALICK, R.,M. and SHAPIRO, L.G. *Computer Vision and Robot Vision*, **1**, 1992, Addison-Wesley Publishing Company, USA.
15. UMBAUGH, S.E. *Computer Vision and Image Processing: A Practical Approach Using CVIPtools*, 1998, Prentice Hall PTR, Upper Saddle River, NJ. USA.
16. LEE, C.S., HSIAO, F.B. and JAN, S.S. Design and implementation of linear-quadratic-Gaussian stability augmentation autopilot for unmanned air vehicle, *Aeronaut J*, May 2009, **113**, (1143), pp 275-290.

https://doi.org/10.1017/S000192400000734X Published online by Cambridge University Press