ANNUAL OF NAVIGATION 19/2012/part 2

10.2478/v10367-012-0019-3

ROBERTO SABATINI, CELIA BARTEL, ANISH KAHARKAR, TESHEEN SHAID, LEOPOLDO RODRIGUEZ, DAVID ZAMMIT-MANGION, HUAMIN JIA
Cranfield University — Department of Aerospace Engineering, UK

LOW-COST NAVIGATION AND GUIDANCE SYSTEMS FOR UNMANNED AERIAL VEHICLES — PART 1: VISION-BASED AND INTEGRATED SENSORS

ABSTRACT

In this paper we present a new low-cost navigation system designed for small size Unmanned Aerial Vehicles (UAVs) based on Vision-Based Navigation (VBN) and other avionics sensors. The main objective of our research was to design a compact, light and relatively inexpensive system capable of providing the Required Navigation Performance (RNP) in all phases of flight of a small UAV, with a special focus on precision approach and landing, where Vision Based Navigation (VBN) techniques can be fully exploited in a multisensor integrated architecture. Various existing techniques for VBN were compared and the Appearance-Based Approach (ABA) was selected for implementation. Feature extraction and optical flow techniques were employed to estimate flight parameters such as roll angle, pitch angle, deviation from the runway and body rates. Additionally, we addressed the possible synergies between VBN, Global Navigation Satellite System (GNSS) and MEMS-IMU (Micro-Electromechanical System Inertial Measurement Unit) sensors, as well as the aiding from Aircraft Dynamics Models (ADMs). In particular, by employing these sensors/models, we aimed to compensate for the shortcomings of VBN and MEMS-IMU sensors in high-dynamics attitude determination tasks. An Extended Kalman Filter (EKF) was developed to fuse the information provided by the different sensors and to provide estimates of position, velocity and attitude of the UAV platform in real-time. Two different integrated navigation system architectures were implemented. The first used VBN at 20 Hz and GPS at 1 Hz to augment the MEMS-IMU running at 100 Hz. The second mode also included the ADM (computations performed at 100 Hz) to provide augmentation of the attitude channel. Simulation of these two modes was accomplished in a significant portion of the AEROSONDE UAV operational flight envelope and performing a variety of representative manoeuvres (i.e., straight climb, level turning, turning descent and climb, straight descent, etc.). Simulation of the first integrated navigation system architecture (VBN/IMU/GPS) showed that the integrated system can reach position, velocity and attitude accuracies compatible with CAT-II precision approach requirements. Simulation of the second system architecture (VBN/IMU/GPS/ADM) also showed promising results since the achieved attitude accuracy was higher using the ADM/VBS/IMU than using VBS/IMU only. However, due to rapid divergence of the ADM virtual sensor, there was a need for frequent re-initialisation of the ADM data module, which was strongly dependent on the UAV flight dynamics and the specific manoeuvring transitions performed.

Keywords:

Vision-Based Navigation, Integrated Navigation System, MEMS Inertial Measurement Unit, Unmanned Aerial Vehicle, Low-cost Navigation Sensors.

INTRODUCTION

The use of Unmanned Aerial Vehicles (UAVs) in civil and military applications has much increased as these vehicles provide cost-effective and safe alternatives to manned flights in several operational scenarios. These robotic aircraft employ a variety of sensors, as well as multi-sensor data fusion algorithms, to provide autonomy to the platform in the accomplishment of mission- and safety-critical tasks. UAVs are characterized by higher manoeuvrability, reduced cost, longer endurance and less risk to human life compared to manned systems. UAV guidance and control depend primarily upon accurate and continuous knowledge of vehicular position and attitude. These tasks can be accomplished by a remote pilot and, frequently, implementing automatic flight modes. Technical requirements for air navigation systems primarily include accuracy, physical characteristics such as weight and volume, support requirements such as electrical power, and system integrity. One of the most important concepts is to use a multi-sensor integrated system to cope with the requirements of long/medium range navigation and landing. This would reduce cost, weight/volume and support requirements and, with the appropriate sensors and integration architecture, give increased accuracy and integrity of the overall system. The best candidates for such integration are indeed satellite navigation receivers and inertial sensors. In recent years, computer vision and Vision-Based Navigation (VBN) systems have started to be applied to UAVs. VBN can provide a self-contained autonomous navigation solution and can be used as an alternative (or an addition) to the traditional sensors (GPS, INS and integrated GPS/INS).

The required information to perform autonomous navigation can be obtained from compact and lightweight cameras. Sinopoli [22] used a model-based approach to develop a system which processes image sequences from visual sensors fused with readings from GPS/INS to update an inaccurate 3D model of the surroundings that was built with Digital Elevation Maps (DEM). Occupancy Grid Mapping was used to divide the map into cells. The probability of an obstacle being present in a cell is associated with each one. Using this ‘risk map’ and the images from the visual sensors, the UAV can update its stored virtual map. Se [21] proposed a system which deals with Vision-Based SLAM using a trinocular stereo system. In this study, Scale--Invariant Feature Transform (SIFT) was used for tracking natural landmarks and to build the 3D maps. The algorithm built submaps from multiple frames which were merged together. The SIFT features detected in the current frame were matched to the pre-built database map in order to obtain the location of the vehicle. A stereo Vision-Based navigation system was implemented on the 2004 twin Mars Exploration Rovers, Spirit and Opportunity. They were equipped with three types of stereo cameras which allowed them to navigate the Martian surface for long distances. A stereo Vision-Based navigation system for lunar rovers was proposed by Cui [6]. The system performed robust motion estimation and disparity estimation using stereo images for accurate lunar navigation, obstacle avoidance and DEM reconstruction. The position of the rover while descending was found using sequence images taken by a descent camera and the reading were integrated with measurements from a laser altimeter and DEM. Image processing was carried out using feature detection, tracking and stereo matching. Levenberg-Marquardt non-linear estimation was used for motion estimation and Weighted Zero Sum of Squared Differences gave the disparity estimation. Matsumoto [12] proposed a representation of the visual route taken by robots in appearance-based navigation. This approach was called the View--Sequenced Route Representation (VSRR) in which the visual route connected the initial position and destination was used for localisation and guidance in the autonomous run. Pattern recognition was achieved by matching the features detected in the current view of the camera with the stored images. The visual route was learnt while the robot was manually guided along the required trajectory. A matching error between the previous stored image and current view was used to control the capture of the next key image. The key image with the greatest similarity to the current view represented the start of the visual route. The location of the robot depended purely on the key image used and no assumption was made of its location in 3D space. During the autonomous run, the matching error between the current view and key images was monitored in order to identify which image should be used for guidance. The robot was controlled so as to move from one image location to another and finally reach its destination. This ‘teach-and-replay’ approach was adopted by Courbon [4, 5], Chen [3] and Remazeilles [15]. In the case of [5] and [4], a single camera and natural landmarks were used to navigate a quadrotor UAV along the visual route. The key images were considered as waypoints to be followed in sensor space. Zero Normalised Cross Correlation was used for feature matching between the current view and the key images. A control system using the dynamic model of the UAV was developed. Its main task was to reduce the position error between the current view and key image to zero and to stabilize and control the UAV. Vision algorithms to measure the attitude of a UAV using the horizon and runway were presented by Xinhua [26] and Dusha [9]. The horizon is used by human pilots to control the pitch and roll of the aircraft while operating under visual flying rules. A similar concept is used by computer vision to provide an intuitive means of determining the attitude of an aircraft. This process is called Horizon-Based Attitude Estimation (HBAE). In [26] and [9], grayscale images were used for image processing. The horizon was assumed to be a straight line and appeared as an edge in the image. Texture energy method was used to detect it and this was used to compute the bank and pitch angle of the UAV. The position of the UAV with respect to the runway was found by computing the angles of the runway boundary lines. During this research, it was observed that the image processing frontend was susceptible to false detection of the horizon if any other strong edges were present in the image. Therefore, an Extended Kalman Filter (EKF) was implemented to filter out these incorrect results. The performance of the algorithms was tested via test flights with a small UAV and a Cessna 172. During this research new VBN sensor specifically tailored for approach/landing applications were design and tested. Furthermore, during approach phase runaway features extraction was employed. Additionally, various candidates were considered for integration with the VBN sensor, including Global Navigation Satellite Systems (GNSS) and Micro Electro Mechanical Systems (MEMS) based Inertial Measurement Units (IMUs). MEMS-IMUs are low-cost and low-volume/ weight sensors particularly well suited for small/medium size UAV applications. However, their integration represent a challenge, which need to be addressed either by finding improvements to the existing analytical methods or by developing novel algorithmic approaches that counterbalance the use of less accurate inertial sensors. In line with the above discussions, the main objective of our research was to develop a low-cost and low-weight/volume Navigation and Guidance System (NGS) based on VBN and other low-cost and low-weight/volume sensors, capable of providing the required level of performance in all flight phases of a small/medium size UAV, with a special focus on precision approach and landing (i.e., the most demanding and potentially safety-critical flight phase), where VBN techniques can be fully exploited in a multisensory integrated architecture. The NGS is composed by a Multisensor Integrated Navigation System (MINS) using an Extended Kalman Filter (EKF) and an existing controller that employs Fuzzy logic and Proportional-Integral-Differential (PID) technology.

VBN SENSOR DESIGN, DEVELOPMENT AND TEST

VBN techniques use optical sensors to extract visual features from images which are then used for localization in the surrounding environment. Cameras have evolved as attractive sensors as they help design economically viable systems with simpler hardware and software components. Computer vision has played an important role in the development of UAVs [13]. Considerable work has been made over the past decade in the area of Vision-Based techniques for navigation and control [26]. UAV Vision-Based systems have been developed for various applications ranging from autonomous landing to obstacle avoidance. Other applications looked into the possible augmentation INS and GPS/INS by using VBN measurements [17]. The vast majority of VBN sensor schemes fall into one of the following two categories [7]: Model-based Approach (MBA) and Appearance-based Approach (ABA). MBA uses feature tracking in images and create a 3D- model of the workspace in which robots or UAV operates [20]. The 3D maps are created in an offline process using a priori information of the environment. Localisation is carried out using feature matching between the current view of the camera and the stored 3D model. MBA is the most common technique currently implemented for Vision-Based navigation. However, the accuracy of this method is dependent on the features used for tracking, robustness of the feature descriptors and the algorithms used for matching and reconstruction. The reconstruction in turn relies on proper camera calibration and sensor noise. Knowledge of surroundings so as to develop the 3D models is also required prior the implementation. ABA algorithms eliminate the need for a metric model as they work directly in the sensor space. This approach utilizes the appearance of the whole scene in the image, contrary to model-based approach which uses distinct objects such as landmarks or edges [12]. The environment is represented in the form of key images taken at various locations using the visual sensors. In this approach, localisation is carried out by finding the key image with the most similarity to the current view. The vehicle of interest is controlled by either coding the action required to move from one key image to another or by a more robust approach using visual servoing [15, 16]. The ABA approach is relatively new and has gained active interest. Modelling of the surrounding using a set of key images is more straightforward to implement compared to 3D modelling. However, a major drawback of this method is its limited applicability since the key images need to be recaptured each time the vehicle moves to a new workspace. It is limited to work in the explored regions which have been visualised during the learning stage [1]. The ABA approach has a disadvantage in requiring a large amount of memory to store the images and is computationally more costly than MBA. Nevertheless, due to improvements in computer technology, this technique has become a viable solution in many application areas. We selected the ABA approach for the design of our VBN sensor system.

Learning Stage

During this stage, a video is recorded using the on-board camera while guiding the aircraft manually during the landing phase. The recorded video is composed of a series of frames which form the visual route for landing. This series of frames is essentially a set of images connecting the initial and target location images. The key frames are first sampled and the selected images are stored in the memory to be used for guidance during autonomous landing of the aircraft. During the learning stage, the UAV is flown manually meeting the Required Navigation Performance (RNP) requirements of precision approach and landing. If available, Instrument Landing System (ILS) can also be used for guidance. It should be noted that if the UAV needs to land at multiple runways according to its mission, the visual route for all the runways is required to be stored in the memory. The following two methods can be employed for image capture during the learning stage.

- **Method 1:** Frames are captured from the video input at fixed time intervals. Key frames are selected manually.
- **Method 2:** Frames are captured using a matching difference threshold [12]. This matching difference threshold is defined in number of pixels and can be obtained by tracking the features in the current view and the previously stored key image. The key images can then be selected based on the threshold and stored in the memory.

The algorithm starts by taking an image at the starting point. Let this image be M_i captured at location *i*. As the aircraft moves forward, the difference between the current view (V) and the image M_i increases. This difference keeps increasing until it reaches the set threshold value (x). At this point, a new image M_{i+1} is taken (replacing the previous image M_i) and the process is repeated until the aircraft reaches its destination.

Localisation

Localisation aims to determine the current location of the aircraft at the start of autonomous run. This process identifies the key image which is the closest match to the current view. The current view of the aircraft is compared with a certain number of images, preferably the ones at the start of the visual route. The key image with the least matching difference is considered to be the start of the visual route to be followed by the aircraft. At the start of the autonomous run, the aircraft is approximately at the starting position of the visual route. The current view, captured from the on-board camera is compared with a set of images (stored previously in the memory during the learning stage) in order to find the location of aircraft with respect to the visual route. The key image with the least matching difference with the current view is considered to be the location of the aircraft and marks the start of the visual route to be followed. In this example, the number of images to be compared (X) is taken as 20. First, the algorithm loads the current view (V) and the first key frame (M_i). Then the difference between the current view and the current key frame is computed. The algorithm then loads the next key frame M_{i+1} and again computes the difference with the current view. If this difference is less than the previous difference, Mi+1 replaces Mi, and the process is repeated again. Otherwise, Mi is considered as the current location of the aircraft.

Autonomous Run

During the autonomous run phase, the aircraft follows the visual route (previously stored in memory during the learning stage) from the image identified as the current location of the aircraft during localisation. The set of key images stored as the visual route can be considered as the target waypoints for the aircraft in sensor space. The current view is compared to the key images so as to perform visual servoing. The approach followed to identify the key image to be used for visual servoing, is describes as follows. Let M_j be the current key frame, i.e. image with the least matching difference with the current view. During the autonomous run, the current key image and the next key image (M_{j+1}) are loaded. The matching differences of the current view V with M_j and M_{j+1} (which are $D_{Mj,V}$ and $D_{Mj+1,V}$ respectively) are tracked. When the matching difference $D_{Mj,V}$ exceeds $D_{Mj+1,V}$, M_{j+1} is taken as the current key image replacing M_j and the next key image is loaded as M_{j+1}. This same process keeps repeating until the aircraft reaches its destination, that is the final key frame. The proposed vision based navigation process is depicted in fig. 1. The key frames represent the visual route the aircraft requires to follow.

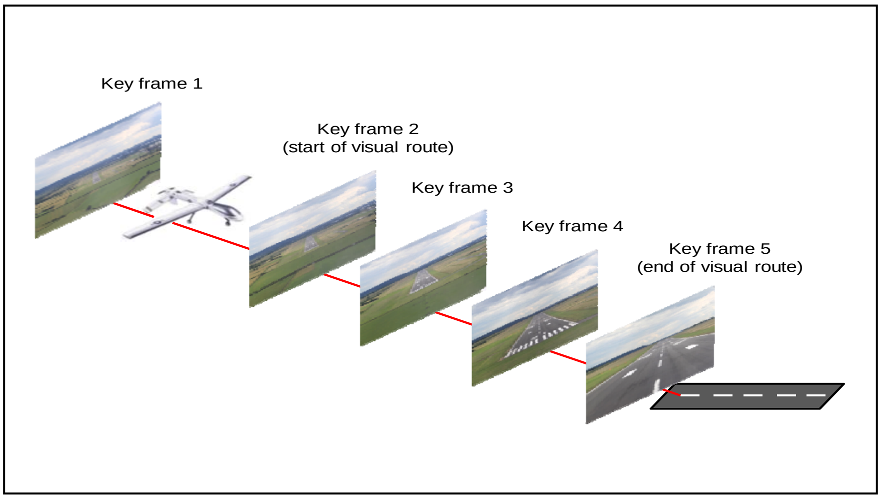


Fig. 1. VBN process [own study]

The figure shows that the key frame 2 is identified as the starting point of the visual route during the localisation process. The on-Dboard computer tracks the matching difference between current view and the second and third key frames until the difference for key frame 2 and the current view exceeds the difference of key frame 3 and the current view. At this stage, key image 3 is used to control the UAV and the matching differences between key frames 3, 4 and the current view are monitored. This process is repeated until the UAV reaches its destination. To capture the outside view, a monochrome Flea camera from Point Grey Research was used. This camera was also used in a previous study on stereo vision [14] and was selected for this project. The Flea camera is a pinhole charged coupled device (CCD) camera with a maximum image resolution of 1024 x 768 pixels. It is capable of recording videos at a maximum rate of 30 fps. An IEEE 1394 connection was used to interface the camera and computer with a data transfer speed of 400 Mbps. A fully manual lens which allows manual focus and zoom was fitted to the camera. A UV filter was employed to protect the lens and to prevent haziness due to ultraviolet light.

Image Processing Module

The Image Processing Module (IPM) of the VBN system detects horizon and runway centreline from the images and computes the aircraft attitude, body rates and deviation from the runway centreline. As a first step of IPM reprocessing, the size of the image is reduced from 1024 x 768 pixels to 512 x3 84 pixels. It was found that this size reduction speeds up the processing without significantly affecting the features detection process. The features such as the horizon and the runway centreline are extracted from the images for attitude computation. The horizon is detected in the image by using Canny edge detector while the runway centreline is identified with the help of Hough Transform. The features are extracted from both, the current view (image received from the on-board camera) and the current key frame. The roll and pitch are computed from the detected horizon while the runway centreline in used to compute the deviation of aircraft from the runway centreline. Then the roll and pitch difference are computed between the current view and the current key frame. Optical flow is determined for all the points on the detected horizon line in the images. The aircraft body rates are then computed based on the optical flow values. The image processing module provides the aircraft attitude, body rates, pitch and roll differences between current view and key frame, and deviation from the runway centreline. The attitude of the aircraft is computed based on the detected horizon and the runway. The algorithm calculates the pitch and roll of the aircraft using the horizon information while aircraft deviation from the runway centreline is computed using the location of runway centreline in the current image. Fig. 2 shows the relationship between the body (aircraft) frame (O_b, X_b,Y_b, Z_b), camera frame (O_c, X_c,Y_c, Z_c) and Earth frame coordinates (O_w, X_w, Y_w, Z_w).

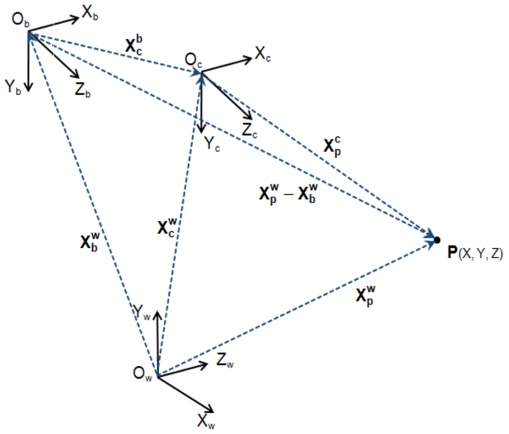


Fig. 2. Relation between the used coordinate systems [own study]

The position of a 3D space point P in Earth coordinates is represented by a vector X_p^w with components x_p, y_p and z_p in the Earth frame. The position of aircraft centre with respect to the Earth coordinates is represented by the vector X_b^w with components x_b, y_b and z_b in the Earth frame. The vector X_p^c represents the position of the point P with respect to the camera frame with components x_{cp}, y_{cp} and z_{cp} in the camera frame. The position of centre of camera lens with respect to the body frame is represented by the vector X_c^b. The vector X_c^w represents the position of lens centre with respect to the ground frame with components x_c, y_c and z_c in the ground frame. The position of point P with respect to body frame with components in the Earth frame can be computed as X_p^w-X_b^w. From [18] the optical flow equation is:

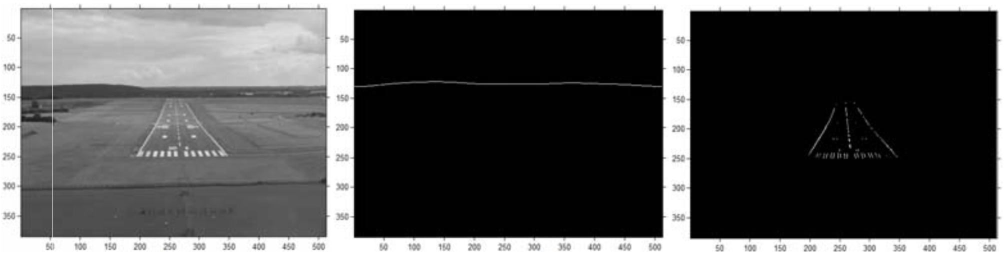
$$\begin{bmatrix}\dot{u}\\ \dot{v}\end{bmatrix} = \begin{bmatrix}\frac{uv}{f} & -(f+\frac{u^2}{f}) & v\\ (f+\frac{v^2}{f}) & -\frac{uv}{f} & -u\end{bmatrix}\begin{bmatrix}\omega_x\\ \omega_y\\ \omega_z\end{bmatrix} \tag{1}$$

where (u, v, f) are the coordinates of P in the image plane and the vector $(\omega_x, \omega_y, \omega_z)$ correspond to the rotational rates of the vehicle of interest. To minimize the effect of errors, a Kalman filter was employed. The state vector consists of the roll angle, pitch angle and body rates of the aircraft. It is assumed that the motion model of the aircraft is disturbed by uncorrelated zero-mean Gaussian noise. The measurement equations are comprised of direct observations of the pitch and the roll from the horizon and i optical flow observations on the detected horizon line. The relation between measurement vector and states is represented by following linear equations [18]:

$$\begin{bmatrix}\phi(k)\\ \theta(k)\\ \dot{u}_1(k)\\ \dot{v}_1(k)\\ \cdots\\ \dot{u}_i(k)\\ \dot{v}_i(k)\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0\\ 0 & 0 & \frac{u_1v_1}{f} & -\left(f+\frac{u_1{}^2}{f}\right) & v_1\\ 0 & 0 & \left(f+\frac{v_1{}^2}{f}\right) & \frac{-u_1v_1}{f} & -u_1\\ \cdots & \cdots & \cdots & \cdots & \cdots\\ 0 & 0 & \frac{u_iv_i}{f} & -\left(f+\frac{u_i{}^2}{f}\right) & v_i\\ 0 & 0 & \left(f+\frac{v_i{}^2}{f}\right) & \frac{-u_iv_i}{f} & -u_i\end{bmatrix}\begin{bmatrix}\phi(k)\\ \theta(k)\\ \omega_x(k)\\ \omega_y(k)\\ \omega_z(k)\end{bmatrix} \tag{2}$$

VBN Sensor Performance

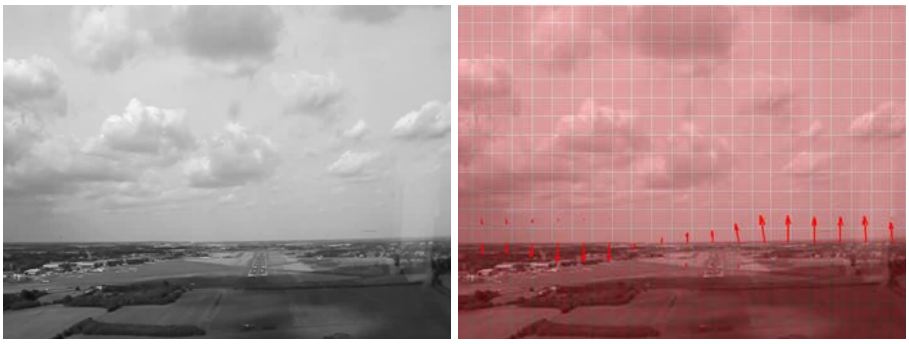
Based on various laboratory, ground and flight test activities with small aircraft and UAV platforms, the performance of the VBN sensor were evaluated. Fig. 3(a), (b) shows a sample image used for testing the VBN sensor algorithms and the results of the corresponding horizon detection process for attitude estimation purposes. The algorithm detects the horizon and the runway centreline from the images. The horizon is detected in the image by using Canny edge detector with a threshold of 0.9 and standard deviation of 50. threshold and the standard deviation values were selected by hit--and-trial method. The resulting image after applying the Canny edge detector is a binary image. The algorithm assigns value ‘1’ to the pixels detected as horizon while the rest of the pixels in the image are assigned value ‘0’. From this test image, the computed roll angle is 1.26° and the pitch angle is –10.17°. To detect the runway in the image, kernel filter and Hough Transform are employed. The runway detected from the same test image is shown in fig. 3(a), (c). For this image, the location of the runway centreline was computed in pixels as 261. The features were extracted from both the current view (image received from the on-board camera) and the current key frame. After the pitch, roll and centreline values were determined, the roll/pitch differences and the deviation from centreline are computed between the current view and the current key frame.



(a) b) (c)

Fig. 3. Original image of the runway (a), identified runway (b), and horizon (c) during the landing phase [own study]

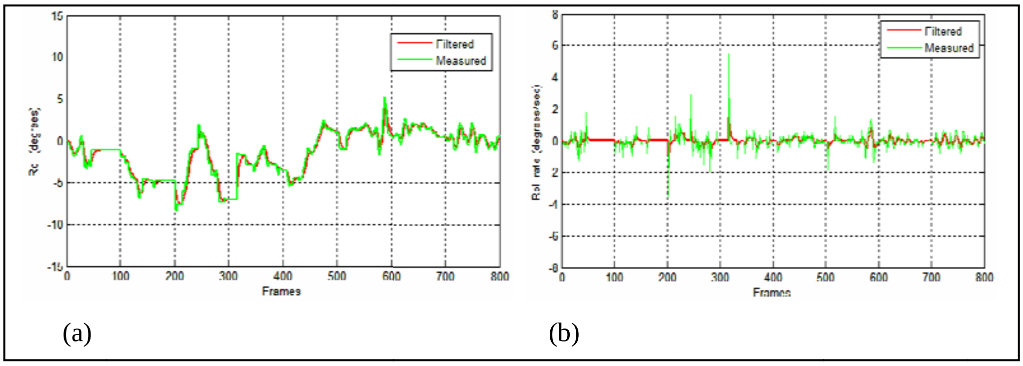
The algorithm also computes the optical flow for all points on the detected horizon line in the images. The optical flow is determined based on the displacement of points in two consecutive frames of a video. The algorithm takes two consecutive frames at a time and determines the motion for each point on the horizon. These optical flow values are used to compute the body rates of the aircraft. An example of the optical flow calculation is shown in fig. 4, where the original image (from the camera) is shown on the left and the image on the right shows the optical flow vectors (in red) computed for the detected horizon line. The vectors are magnified by a factor of 20. Since the vectors on the right half of the horizon line are pointing upwards and the vectors on the left half are pointing downwards, the aircraft is performing roll motion (clockwise direction).



(a) (b)

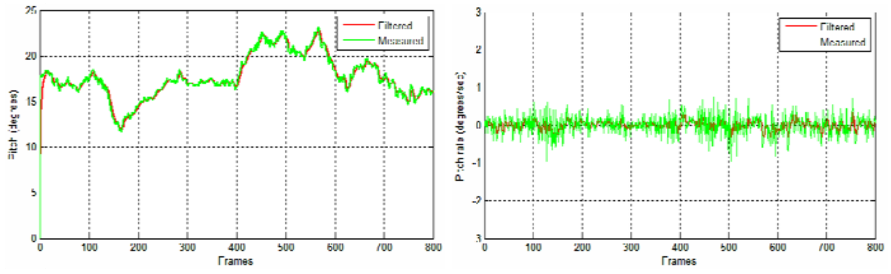
Fig. 4. Received image from camera (a) and optical flow computed for the detected horizon (b) [own study]

The real-time performance of the IPM algorithms were evaluated using a combination of experimental data (from the VBN camera) collected in flight and IPM simulation/data analysis performed on the ground using MATLAB®. The algorithm processed the video frame by frame and extracted horizon and the runway from each frame. The roll and pitch of the aircraft were computed based on the horizon detected in each frame. The algorithm also identified the location of runway centreline in each frame which was further used to calculate the deviation of the aircraft from the runway centreline. Kalman filter was employed to reduce the effect of errors in the measurements. The roll and roll-rate results obtained for 800 frames are shown in fig. 5. Similarly, fig. 6 depicts the results for pitch and pitch-rate. The computed location of centreline (pixels) and the centreline drift rate (pixels per second) are shown in fig. 7.



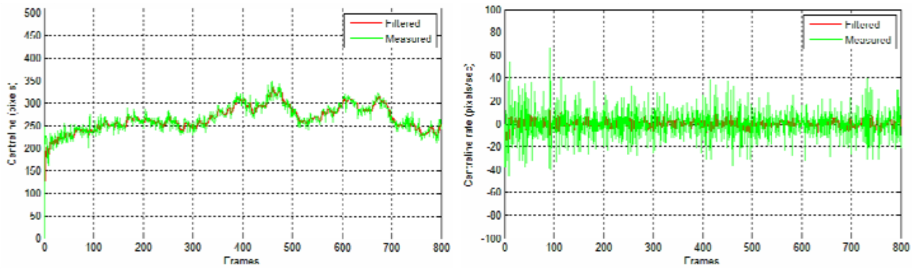
(a) (b)

Fig. 5. Roll (a) and roll-rate (b) computed from the test video [own study]



(a) (b)

Fig. 6. Pitch (a) and pitch-rate (b) computed from the test video [own study]



(a) (b)

Fig. 7. Computed location (a) and rate of change (b) of centreline location from the test video [own study]

Although the test activities were carried out in a limited portion of the aircraft/UAV operational flight envelopes, some preliminary error analysis was performed comparing the performance of the VBN sensor an INS. The mean and standard deviation of the VBN attitude and attitude-rate measurements are listed in table 1.

Table 1. VBS Attitude and Angular Rates Errors Parameters [own study]

VBN Measured Parameters	**Mean**	**Standard Deviation**
Roll angle	0.22°	0.02°
Pitch angle	–0.32°	0.06°
Yaw angle (centreline deviation)	0.64°	0.02°
Roll Rate	0.33°/s	0.78 °/s
Pitch Rate	–0.43°/s	0.75°/s
Yaw Rate	1.86°/s	2.53°/s

The performance of the VBN sensor is strongly dependent on the characteristics of the employed camera. The developed algorithms are unable to determine the attitude of the aircraft in case of absence of horizon in the image. Similarly, the deviation of the aircraft from the runway centreline cannot be computed in the absence of runway in the image. The most severe physical constrain is imposed by the FOV of the camera. The maximum vertical and horizontal FOVs of the Flea Camera are 57.6° and 77.6° respectively [16]. Due to this limitation, the VBN sensor can perform computations with a minimum pitch angle of –28.8° and a maximum of +28.8°. Additionally, environmental factors such as fog, night/low-light conditions or rain also affect the horizon/runway visibility and degrade the performance of the VBN system.

INTEGRATION CANDIDATE SENSORS

There are a number of limitations and challenges associated to the employment VBN sensors in UAV platforms. VBN is best exploited at low altitudes, where sufficient features can be extracted from the surrounding. The FOV of the camera is limited and, due to payload limitations, it is often impractical to install multiple cameras. When multiple cameras are installed, additional processing is required for data exploitation. In this case, also stereo vision techniques can be implemented. Wind and turbulence disturbances must be modelled and accounted for in the VBN processing. Additionally the performance of VBN can be very poor in low-visibility conditions (performance enhancement can be achieved employing infrared sensors as well). However, despite these limitations and challenges, VBN is a promising technology for small-to-medium size UAV navigation and guidance applications, especially when integrated with other low-cost and low-weight/volume sensors currently available. In our research, we developed an integrated NGS approach employing two state-of-the-art physical sensors: MEMS-based INS and GPS, as well as augmentation from Aircraft Dynamic Models (Virtual Sensor) in specific flight phases.

GNSS and MEMS-INS Sensors Characteristics

GNSS can provide high-accuracy position and velocity data using pseudorange, carrier phase, Doppler observables or various combinations of these measurements. Additionally, using multiple antennae suitably positioned in the aircraft, GNSS can also provide attitude data. In this research, we considered GPS Standards Positioning Service (SPS) pseudorange measurements for position and velocity computations. Additional research is currently being conducted on GPS/GNSS Carrier Phase Measurements (CFM) for attitude estimation. Tables 2 lists the position and velocity of state-of-the-art SPS GPS receivers. Position error parameters are from [23] and velocity error parameters are from [8], in which an improved time differencing carrier phase velocity estimation method was adopted. Typically, GPS position and velocity measurements are provided at a rate of 1 Hz.

Table 2. GPS position and velocity errors [own study]

Errors	Mean	Standard Deviation
North Position Error (m)	–0.4	1.79
East Position Error (m)	0.5	1.82
Down Position Error (m)	0.17	3.11
North Velocity Error (mm/s)	0	3.8
East Velocity Error (mm/s)	0	2.9
Down Velocity Error (mm/s)	2.9	6.7

An Inertial Navigation System (INS) can determine position, velocity and attitude of a UAV based on the input provided by various kinds of Inertial Measurement Units (IMUs). These units include 3-axis gyroscopes, measuring the roll, pitch and yaw rates of the aircraft around the body-axis. They also comprise 3-axis accelerometers determining the specific forces in the inertial reference frame. In our research, we considered a strap-down INS employing low-cost MEMS Inertial Measurement Units (IMUs). MEMS-based IMUs are low-cost and low-weight/volume devices that represent an attractive alternative to high-cost traditional INS sensors, especially for general aviation or small UAVs applications. Additionally, MEMS sensors do not necessitate high power and the level of maintenance required is far lower than for high-end INS sensors [24]. The main drawback of these sensors is the relatively poor level of accuracy of the measurements that they provide. In our research, INS-MEMS errors are modeled as White Noise (WN) or as Gauss-Markov (GM) processes [10, 25]. The error parameters in [18] were considered for our simulation.

ADM Virtual Sensor Characteristics

The ADM Virtual Sensor is essentially a knowledge-based module used to augment the navigation state vector by predicting the UAV flight dynamics (airtcraft trajectory and attitude motion). The ADM can employ either a 6-Degree of Freedom (6-DOF) or a 3-DOF variable mass model with suitable constraints applied in the different phases of the UAV flight. The input data required to run these models are made available from aircraft physical sensors (i.e., aircraft data network stream) and form ad-hoc databases. Additionally, for the 3-DOF case, an automatic manoeuvre recognition module is implemented to model the transitions between the different UAV flight phases. Typical ADM error parameters are listed in [10, 25]. Table 3 lists the associated error statistics obtained in a wide range of dynamics conditions for 20 seconds runtime.

Table 3. ADM error statistics [own study]

	Mean	**Standard Deviation**
North Velocity Error	4.48E-3	3.08E-2
East Velocity Error	–3.73E-2	1.58E-1
Down Velocity Error	–4.62E-2	5.03E-2
Roll Error	4.68E-5	7.33E-3
Pitch Error	3.87E-3	2.41E-3
Yaw Error	–1.59E-3	7.04E-3

MULTISENSOR SYSTEM DESIGN AND SIMULATION

The data provided by all sensors were blended using suitable data fusion algorithms. Due to the non-linearity of the sensor models, an EKF was developed to fuse the information provided by the different MINS sensors and to provide estimates of position, velocity and attitude of the platform in real-time. Two different integrated navigation system architectures were defined, including VBN/IMU/GPS (VIG) and VIG/ADM (VIGA). The VIG architecture used VBN at 20 Hz and GPS at 1 Hz to augment the MEMS-IMU running at 100 Hz. The VIGA architecture included the ADM (computations performed at 100 Hz) to provide attitude channel augmentation. The corresponding VIG and VIGA integrated navigation modes were simulated using MATLAB^{TM} covering all relevant flight phases of the AEROSONDE UAV (straight climb, straight-and-level flight, straight turning, turning descend/climb, straight descent, etc.). The navigation system outputs were fed to a hybrid Fuzzy-logic/PID controller designed at Cranfield University for the AEROSONDE UAV and capable of operating with stand-alone VBN, as well as with VIG/VIGA and other sensors data.

VIG and VIGA Architectures

The VIG architecture is illustrated in fig. 8. The INS position and velocity provided by the navigation processor are compared to the GPS position and velocity to form the measurement input of the data fusion block containing the EKF. A similar process is also applied to the INS and VBN attitude angles, whose differences are incorporated in the EKF measurement vector. The EKF provides estimates of the Position, Velocity and Attitude (PVA) errors, which are then removed from the sensor measurements to obtain the corrected PVA states. The corrected PVA and estimates of accelerometer and gyroscope biases are also used to update the INS raw measurements.

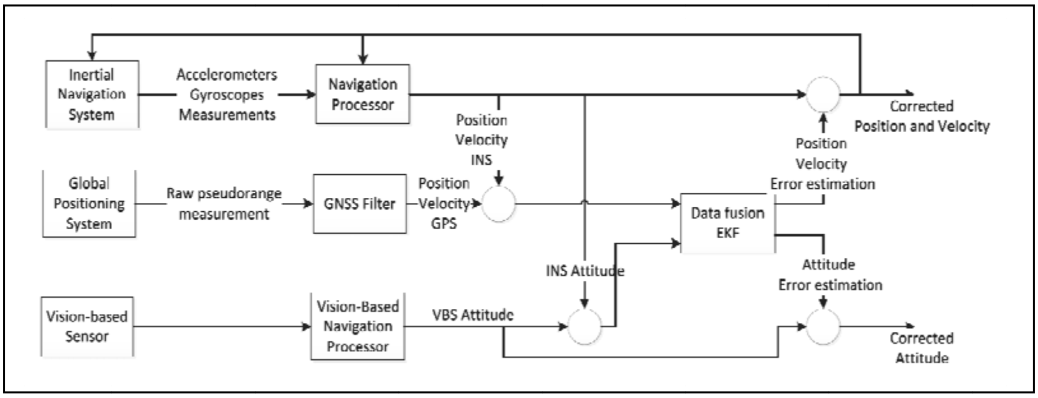

Inertial Navigation System
Accelerometers Gyroscopes Measurements
Navigation Processor
Position Velocity INS
Corrected Position and Velocity
Position Velocity Error estimation
Global Positioning System
Raw pseudorange measurement
GNSS Filter
Position Velocity GPS
Data fusion EKF
INS Attitude
Attitude Error estimation
Vision-based Sensor
Vision-Based Navigation Processor
VBS Attitude
Corrected Attitude

Fig. 8. VIG Architecture [own study]

The VIGA architecture is illustrated in fig. 10. As before, the INS position and velocity provided by the navigation processor are compared to the GPS data to form the measurement input of EKF. Additionally, in this case, the attitude data provided by the ADM and the INS are compared to feed the EKF at 100 Hz, and the attitude data provided by the VBS and INS are compared at 20 Hz and input to the EKF. Like before, the EKF provides estimations of PVA errors, which are removed from the INS measurements to obtain the corrected PVA states. Again, the corrected PVA and estimates of accelerometer and gyroscope biases are used to update INS raw measurements.

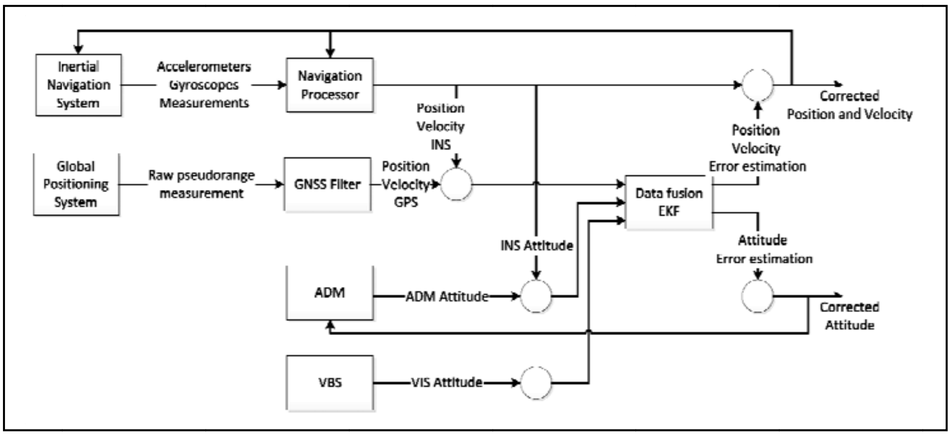
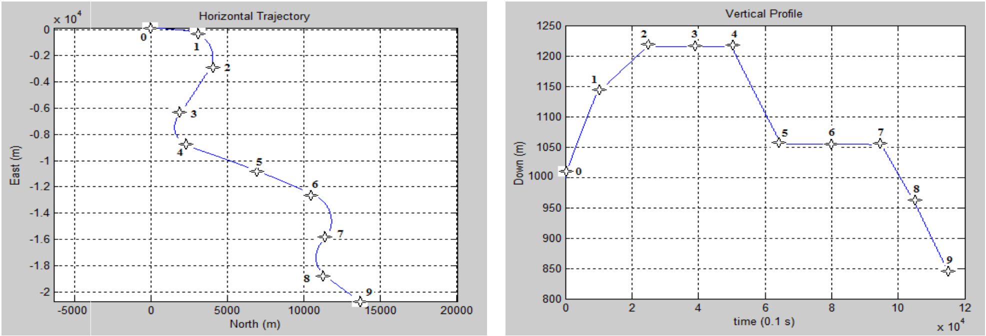

Inertial Navigation System
Accelerometers Gyroscopes Measurements
Navigation Processor
Position Velocity INS
Corrected Position and Velocity
Position Velocity Error estimation
Global Positioning System
Raw pseudorange measurement
GNSS Filter
Position Velocity GPS
Data fusion EKF
INS Attitude
Attitude Error estimation
ADM
ADM Attitude
Corrected Attitude
VBS
VIS Attitude

Fig. 9. VIGA Architecture [own study]

VIG AND VIGA SIMULATION

Both the VIG and VIGA multisensor architectures were tested by simulation in an appropriate sequence of flight manoeuvres representative of the AEROSONDE UAV operational flight envelope. An FLC/PID controller was used for simulation. The duration of the simulation is 1150 seconds (approximately 19 minutes). The horizontal and vertical flight profiles are shown in fig. 10.



(a) (b)

Fig. 10. Horizontal and vertical flight profiles [own study]

The list of the different simulated flight manoeuvres and associated control inputs is provided in table 4. The numbered waypoints are the same shown in fig. 10.

Table 4. Flight manoeuvres and control inputs [own study]

Flight Maneuver	**Required Roll (°)**	**Required Pitch (°)**	**Time (s)**	**Legs (Waypoints)**
Straight Climb (Take off)	0	10.00	100	[0,1]
Right Turning Climb	–2	4.00	150	[1,2]
Straight and Level	0	2.25	150	[2,3]
Level Left Turn	3	2.25	100	[3,4]
Straight Descent	0	–0.7	150	[4,5]
Straight and Level	0	2	150	[5,6]
Level Right Turn	–2	2	150	[6,7]
Left Turning Descent	3	–0.5	100	[7,8]
Straight Descent	0	–1	100	[8,9]

The VIG position error time histories (east, north and down) are shown in fig. 11. For comparison, also the GPS position errors (unfiltered) are shown. Table 5 presents the position error statistics associated to each flight phase.

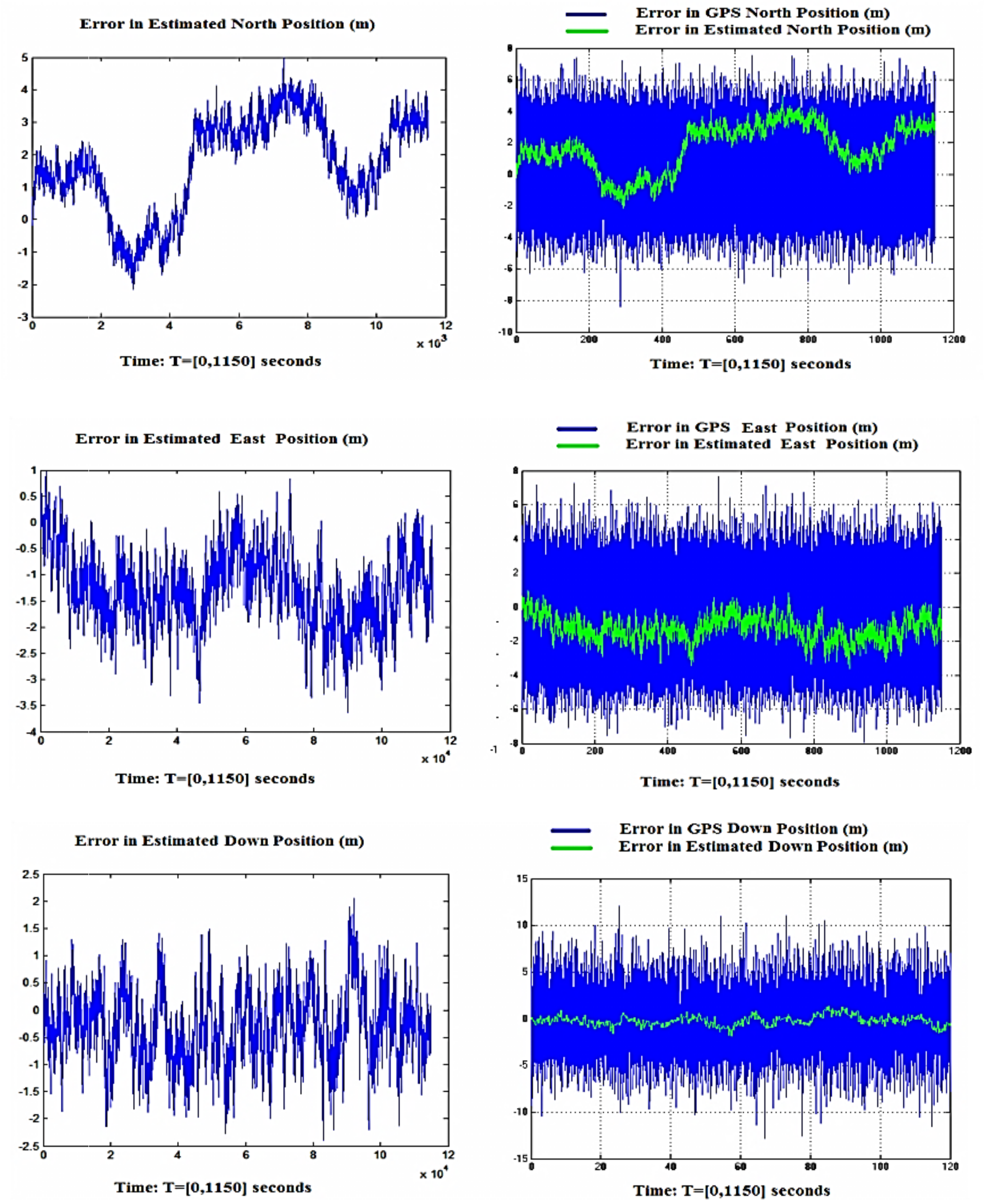


Fig. 11. VIG position error time histories [own study]

Table 5. VIG position error stastics [own study]

Phase of Flight	North Position		East Position		Down Position	
	Mean (m)	σ (m)	Mean (m)	σ (m)	Mean (m)	σ (m)
Straight Climb	1.22	4.38E-01	–4.34E-01	6.20E-01	–1.80E-01	5.30E-01
Right Turning Climb	9.88E-01	8.68E-01	–1.36	4.86E-01	–2.40E-01	6.02E-01
Straight and Level	–8.01E-01	4.71E-01	–1.51	5.02E-01	–4.38E-01	6.82E-01
Level Left Turn	1.16	1.32	–1.66	5.68E-01	–5.28E-01	6.57E-01
Straight Descent	2.77	3.60E-01	–7.20E-01	4.48E-01	–4.92E-01	6.36E-01
Level Right Turn	2.06	1.07	–1.86	5.92E-01	–1.29E-01	8.37E-01
Left Turning Descent	1.65	7.29E-01	–1.63	5.35E-01	–2.66E-01	6.23E-01

The VIG velocity error time histories are shown in fig. 12. For comparison, the GPS velocity error time histories are also shown. GPS is the dominating sensor for velocity computations but a significant improvement is with the VIG system on the accuracy of the vertical data. Table 6 shows the velocity error statistics associated to each flight phase.

Table 6. VIG velocity error statistics [own study]

Phase of Flight	North Velocity		East Velocity		Down Velocity	
	Mean (m/s)	σ (m/s)	Mean (m/s)	σ (m/s)	Mean (m/s)	σ (m/s)
Straight Climb	–6.40E-03	1.73E-02	–4.14E-03	2.14E-02	1.30E-02	1.89E-01
Right Turning Climb	–7.97E-03	1.11E-02	–7.59E-03	7.93E-03	–2.90E-04	6.79E-03
Straight and Level	–7.19E-03	1.00E-02	3.63E-03	1.08E-02	–4.20E-04	6.78E-03
Level Left Turn	1.44E-02	1.51E-02	3.61E-03	6.99E-03	–1.80E-04	6.90E-03
Straight Descent	–4.50E-03	3.64E-02	–3.71E-03	3.32E-02	–3.80E-04	9.55E-03
Level Right Turn	–3.11E-02	1.08E-01	1.21E-02	4.91E-02	7.64E-04	9.82E-03
Left Turning Descent	–3.28E-02	1.13E-01	–6.33E-03	3.71E-02	–1.32E-03	1.54E-02

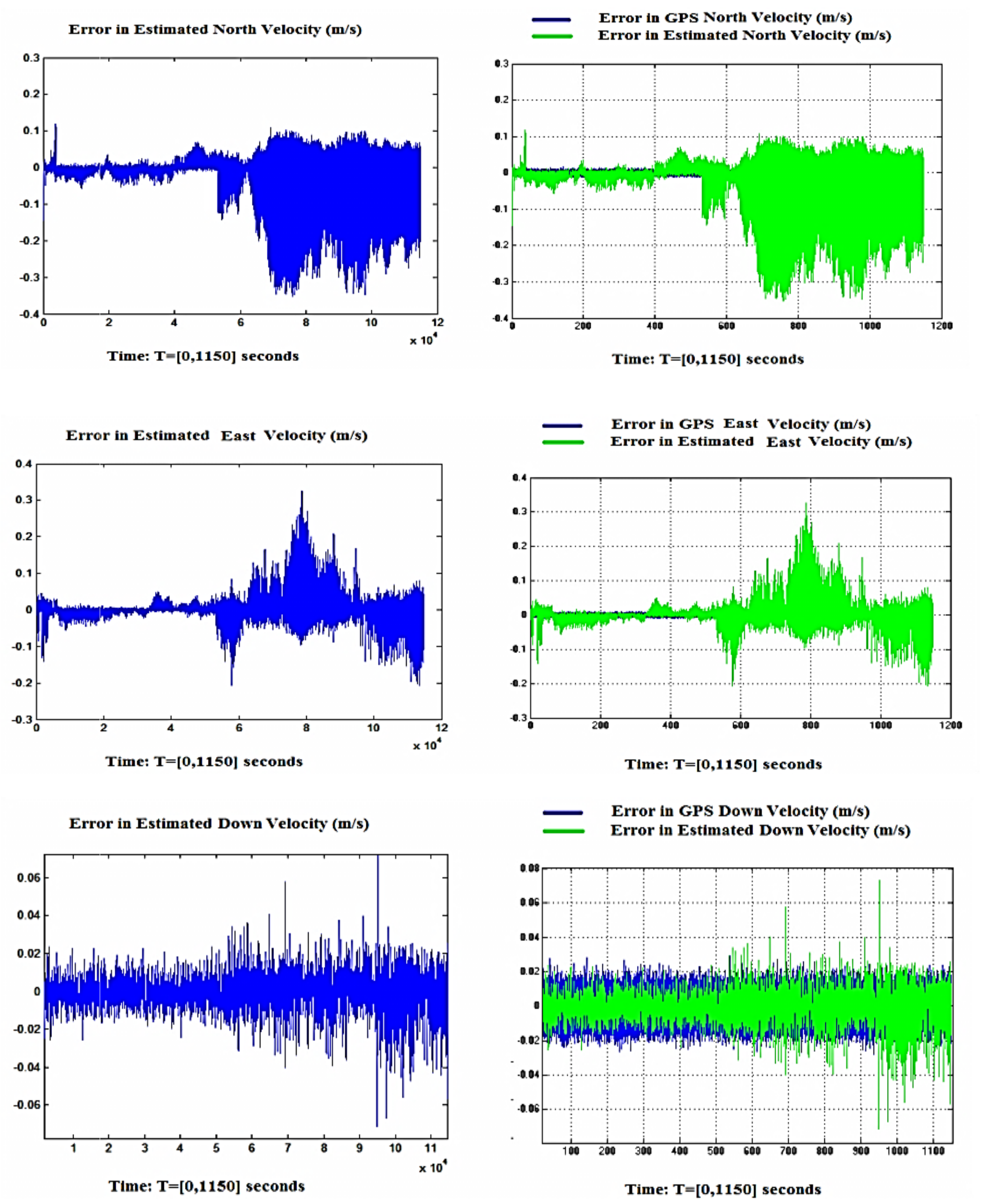
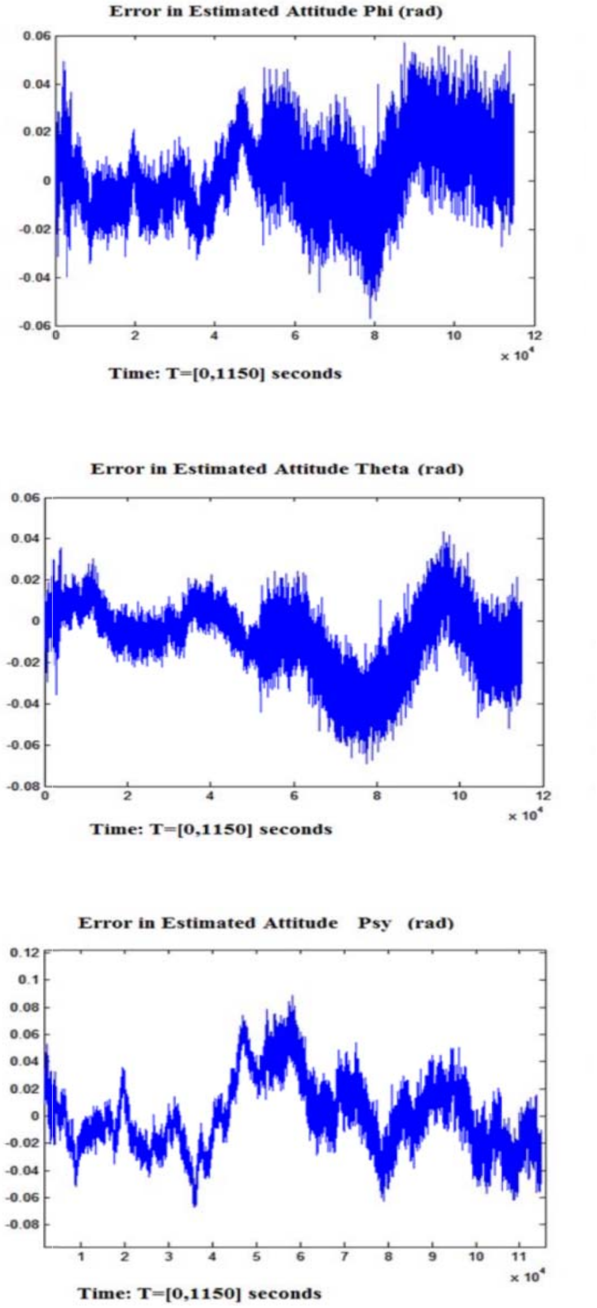


Fig. 12. VIG velocity error time histories [own study]

The attitude error time histories of the VIG system are shown in fig. 13. Table 7 presents the associated attitude error statistics.



Error in Estimated Attitude Phi (rad)

Error in Estimated Attitude Theta (rad)

Error in Estimated Attitude Psy (rad)

Fig. 13. Attitude error time histories [own study]

Table 7. VIG attitude error statistics [own study]

Phase of Flight	Roll (Phi)		Pitch (Theta)		Yaw (Psy)	
	Mean (m/s)	σ (m/s)	Mean (m/s)	σ (m/s)	Mean (m/s)	σ (m/s)
Straight Climb	–6.07E-02	5.54E-01	3.16E-01	2.92E-01	–2.17E-01	1.06
Right Turning Climb	–4.25E-01	3.09E-01	–5.10E-02	4.65E-01	–5.39E-01	6.99E-01
Straight and Level	–4.22E-01	3.44E-01	–2.24E-02	3.80E-01	–1.38	8.02E-01
Level Left Turn	6.13E-01	4.96E-01	–1.39E-01	4.27E-01	1.39	1.36
Straight Descent	3.89E-01	4.60E-01	–3.68E-01	3.50E-01	2.03	1.08
Level Right Turn	7.58E-01	7.69E-01	–5.46E-01	8.54E-01	3.91E-01	8.59E-01
Left Turning Descent	1.22	7.06E-01	–5.37E-01	7.84E-01	–3.36E-01	8.86E-01

As discussed above, the ADM data were used in the VIGA architecture to update the attitude channel (the position and velocity channels are derived from the VIG system). Therefore, only the attitude error statistics of the VIGA system are presented here. The time histories of the VIGA attitude errors are shown in fig. 14 and compared with the corresponding VIG attitude errors. Table 8 presents the VIGA attitude error statistics.

Table 8. VIGA attitude error statistics [own study]

Phase of Flight	Roll (Phi)		Pitch (Theta)		Yaw (Psy)	
	Mean (deg)	σ (deg)	Mean (deg)	σ (deg)	Mean (deg)	σ (deg)
Straight Climb	–6.76E-02	5.19E-01	3.58E-01	2.08E-01	–1.19E-01	1.01
Right Turning Climb	–4.42E-01	2.64E-01	–8.37E-02	4.07E-01	–6.07E-01	6.97E-01
Straight and Level	–4.38E-01	3.06E-01	–3.61E-02	3.27E-01	–1.44	7.92E-01
Level Left Turn	6.16E-01	4.77E-01	–1.60E-01	3.70E-01	1.45	1.31
Straight Descent	3.92E-01	3.58E-01	–4.22E-01	2.37E-01	2.07	1.08
Level Right Turn	7.79E-01	6.99E-01	–6.49E-01	7.39E-01	4.73E-01	8.30E-01
Left Turning Descent	9.00E-02	1.44E-01	3.74E-01	4.50E-01	–1.78E-01	8.59E-01

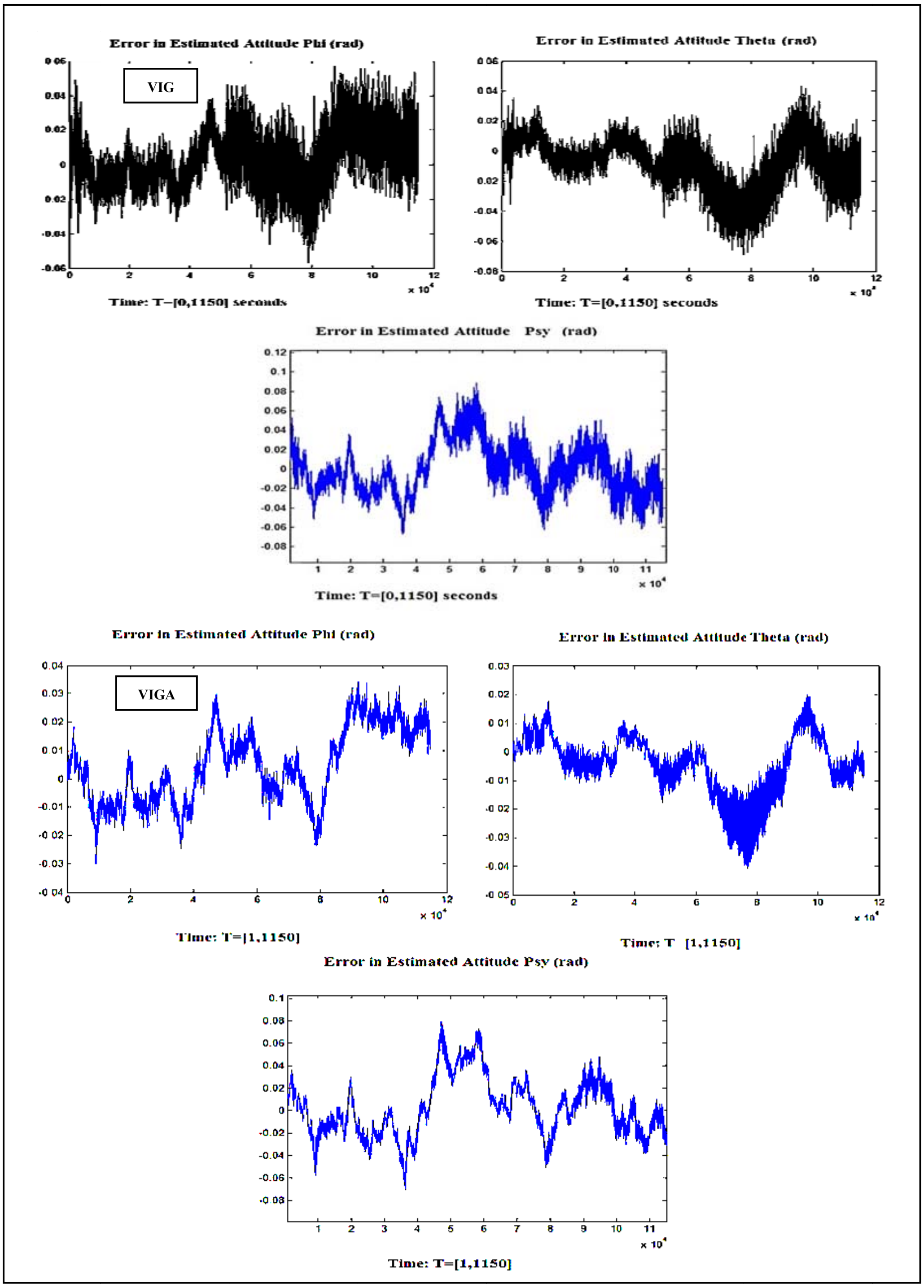


Fig. 14. Comparison of VIGA and VIG attitude errors [own study]

During the initial VIGA simulation runs it was evidenced that the ADM data cannot be used without being reinitialised regularly. For the AEROSONDE UAV manoeuvres listed in table 10, it was found that the optimal period between ADM reinitialisation was in the order of 20 seconds. Converting the data in table 6 and 6 to the corresponding RMS (95%) values, we obtain the error data in table 9. Comparing the two tables, it is evident that the ADM virtual sensor contributes to a moderate reduction of the overall attitude error budget in all relevant flight phases.

Table 9. VIG and VIGA attitude RMS-95% errors in degrees [own study]

Phase of Flight	RMS-95% Phi VIG	RMS-95% Phi VIGA	RMS-95% Theta VIG	RMS-95% Theta VIGA	RMS-95% Psy VIG	RMS-95% Psy VIGA
Straight Climb	1.11	1.05	9.24E-01	7.56E-01	2.13	1.97
Right Turning Climb	1.05	1.03	9.44E-01	8.20E-01	1.85	1.76
Straight and Level	1.09	1.07	7.63E-01	6.55E-01	3.30	3.18
Level Left Turn	1.58	1.56	9.13E-01	7.90E-01	3.97	3.82
Straight Descent	1.20	1.06	1.10	8.76E-01	4.67	4.60
Level Right Turn	2.16	2.09	2.14	1.84	1.96	1.84
Left Turning Descent	2.46	1.42	1.90	1.17	1.89	1.76

To conclude the simulation data analysis, table 10 shows a comparison of the VIG/VIGA horizontal and vertical accuracy (RMS-95%) with the required accuracy levels for precision approach [2, 11]. The VIG/VIGA performances are in line with CAT II precision approach requirements. Future research will address the possible synergies of the VIG/VIGA architectures with GPS/GNSS space, ground and aircraft-based augmentation systems.

Table 10. VIGA attitude RMS-95% errors [own study]

Category of approach	Horizontal Accuracy (m) 2D RMS-95%		Vertical Accuracy (m) RMS-95% Down	
	Required	VIG/VIGA	Required	VIG/VIGA
CAT I	16	5,8	4	1.61
CAT II	6.9		2	
CAT III	4.1		2	

CONCLUSIONS

In this paper we have described the research activities performed to design a low-cost and low-weight/volume integrated NGS suitable for small/medium size UAV applications. As a first step, we designed and tested a VBN sensor employing appearance-based techniques and specifically tailored for UAV low-level flight, including precision approach and landing operations. In addition to horizon detection and image-flow, the VBN sensor also employed runway features extraction during the approach phase. Various candidates were considered for integration with the VBN sensor and, as a result, GPS and MEMS-IMUs (with possible augmentation from ADM) were finally selected. The multisensory integration was accomplished with an EKF. The attitude/attitude-rate accuracies obtained with the VBN sensor were evaluated by a combination of laboratory, ground and flight test activities. The results were satisfactory in low-level flight and during the approach and landing phase of the UAV flight. However, the VBN sensor performance was strongly dependent on the characteristics of the employed camera. The algorithms developed are unable to determine the attitude of the aircraft in case of absence of horizon in the image. Similarly, the deviation of the aircraft from the runway centreline cannot be computed in the absence of runway in the image. The most severe physical constrain is imposed by the angular FOV of the camera. The maximum vertical and horizontal FOVs of the employed camera are 57.6° and 77.6° respectively. Due to this limitation, the VBN sensor can compute a minimum pitch angle of –28.8° and a maximum of +28.8°. Current research shows that for wind speeds greater than 20m/s, the VBN/IPM algorithms are marginally usable for navigation purposes as well as for guidance. Environmental factors such as fog, night/low-light conditions or rain also affect the horizon/runway visibility and degrade the performance of the VBN system. To cope with these limitations, current research is investigating the potential synergies obtained by integrating daylight camera vision sensors with Infrared and Night Vision Imaging Sensors (IR/NVIS). Simulation of the VIG integrated navigation mode showed that this integration scheme can achieve horizontal/vertical position accuracies in line with CAT-II precision approach requirements, with a significant improvement compared to stand-alone SPS GPS. An improvement was also observed in the accuracy of the vertical velocity data. Additionally, simulation of the VIGA navigation mode showed promising results since, in most cases, the attitude accuracy is higher using the ADM/VBS/IMU rather than using VBS/IMU only. However, due to rapid divergence of the ADM virtual sensor, there is a need for a frequent re-initialisation of the ADM data module, which is strongly dependent on the UAV flight dynamics and the specific manoeuvres/flight-phase transitions performed. In the considered portion of the UAV operational flight envelope, the required re-initialisation interval was approximately 20 seconds. To cope with this issue, the original ADM is being modified to take into account specific manoeuvre constraints and the transition states between various manoeuvres are being carefully modelled. Additionally, an automatic manoeuvre recognition algorithm is being developed for updating the ADM in real-time and providing direct feedback to the VBN sensor. It is expected that, adopting this approach, the performances of the multisensory integrated NGS will be significantly enhanced both in terms of data accuracy and data continuity.

REFERENCES

[1] Blanc G., Mezouar Y., Martinet P., Indoor navigation of a wheeled mobile robot along visual routes, Proceeding of International Conference of Robotics & Automation, 2005, pp. 3354–3359.

[2] CAA Safety Regulation Group Paper 2003/09, GPS Integrity and Potential Impact on Aviation Safety, 2003.

[3] Chen Z., Birchfield S. T., Qualitative Vision-Based path following, IEEE Trans. on Robotics, June 2009, Vol. 25, issue 3, pp. 749–754.

[4] Courbon J., Mezouar Y., Guenard N., Martinet P., Vision-Based navigation of unmanned aerial vehicles, Control Engineering Practice, July 2010, Vol. 18, issue 7, pp. 789–799.

[5] Courbon J., Mezouar Y., Guenard N., Martinet P., Visual navigation of a quadrotor aerial vehicle, Proceedings of the 2009 IEEE/RSJ Conference on Intelligent Robots and Systems, Oct. 2009, pp. 5315–5320.

[6] Cui P., Yue F., Stereo Vision-Based autonomous navigation for lunar rovers, Aircraft Engineering and Aerospace Technology: An International Journal, 2007, Vol. 79, No. 4, pp. 398–405.

[7] Desouza G. N., Kak A. C., Vision for mobile robot navigation: a survey, IEEE Trans. Pattern Analysis and Machine Intelligence, Feb. 2002, Vol. 24, issue 2, pp. 237–267.

[8] Ding W., Wang J., Precise Velocity Estimation with a Stand-Alone GPS receiver, University of New South Wales, Journal of the Institute of Navigation, USA, 2011.

[9] Dusha D., Mejias L., Walker R., Fixed-wing attitude estimation using temporal tracking of the horizon and optical flow, Journal of Field Robotics, 2011, Vol. 28, No. 3, pp. 355–372.

[10] Godha S., Performance Evaluation of Low Cost MEMS-Based IMU Integrated with GPS for Land Vehicle Navigation Application, UCGE Report, 2006, No. 20239, University of Calgary, Department of Geomatics Engineering, Alberta, Canada.

[11] ICAO — Annex 10 to the Convention on International Civil Aviation, Aeronautical Telecommunications, Volume 1: Radio Navigation Aids., ed. 6, July 2006.

[12] Matsumoto Y., Sakai K., Inaba, M., Inoue H., View-based approach to robot navigation, Proceedings of the 2000 IEEE/RSJ Conference on Intelligent Robots and Systems, Nov. 2000, Vol. 3, pp. 1702–1708.

[13] Olivares-Mendez M. A., Mondragon I. F., Campoy P., Martinez C., Fuzzy controller for UAV-landing task using 3D position visual estimation, Proceedings of IEEE International Conference on Fuzzy Systems, 2010.

[14] Rangasamy G., Image sensor fusion algorithms for obstacle detection, location and avoidance for autonomous navigation of UAVs, M.Sc. Thesis, School of Engineering, Cranfield University, 2010.

[15] Remazeilles A., Chaumette F., Image-based robot navigation from an image memory, Journal of Robotics and Autonomous Systems, 2007, Vol. 55, issue 4.

[16] Rives P., Azinheira J. R., Visual auto-landing of an autonomous aircraft, INRIA, 2002, No. 4606.

[17] Roumeliotis S. I., Johnson A. E., Montgomery J. F., Augmenting Inertial Navigation with image-based estimation, Proceeding of International Conference of Robotics & Automation, 2002, pp. 4326–4333.

[18] Sabatini R., Kaharkar A., Shaid T., Bartel C., Jia H., Zammit-Mangion D., Vision-Based Sensors and Integrated Systems for Unmanned Aerial Vehicles Navigation and Guidance, Proceedings of the SPIE Conference Photonics Europe 2012, Brussels (Beligium), April 16–19, 2012.

[19] Sabatini R., Leopoldo Rodríguez Kaharkar A., Bartel C., Shaid T., GNSS Data Processing for Attitude Determination and Control of Unmanned Aerial and Space Vehicles, Proceedings of the European Navigation Conference 2012, Gdansk (Poland), April 25–27, 2012.

[20] Santosh D., Achar S., Jawahar C. V., Autonomous image-based exploration for mobile robot navigation, Proceeding of International Conference of Robotics & Automation, 2008, pp. 2717–2722.

[21] Se S., Lowe D. G., Little J. J., Vision based global localisation and mapping, IEEE Trans. on Robotics, June 2005, Vol. 21, No. 3, pp. 364–375.

[22] Sinopoli B., Micheli M., Donato G., Koo T. J., Vision based navigation for unmanned aerial vehicles, Proceeding of International Conference of Robotics & Automation, 2001, Vol. 2, pp. 1757–1764.

[23] Tiberius C. (2003), Standard positioning service, Handheld GPS receiver acuracy, GPS World.

[24] Titterton D., Weston J., Strapdown Inertial Navigation Technology, 2nd ed., The Institution of Electrical Engineers, 2004.

[25] Troy S., Bruggemann Eng (HONS) B., Eng M., Investigation of MEMS Inertial Sensors and Aircraft Dynamic Models in Global Positioning System Integrity Monitoring for Approaches with Vertical Guidance, Unpublished PhD thesis, Queensland University of Technology, School of Engineering, 2009.

[26] Xinhua L., Cao Y., Research on the application of the Vision-Based autonomous navigation to the landing of the UAV, Proceedings of 5th International Symposium on Instrumentation and Control Technology, 2003, Vol. 5253, pp. 385–388.

Received May 2012
Reviewed November 2012