2017 IEEE 4th International Conference Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD) pp.7-12

Visual Navigation System Adjustment

V.M. Sineglazov
Aviation Computer-Integrated Complexes Department
National Aviation University
Kyiv, Ukraine
svm@nau.edu.ua

V.S. Ischenko
Aviation Computer-Integrated Complexes Department
National Aviation University
Kyiv, Ukraine
IschenkoVitaly@gmail.com

***Abstract*—Unmanned aerial vehicle flight simulation bench for visual navigation system. Visual navigation system gives opportunity to flight in radio interference zone. Considered approach for adjustment and evaluating visual navigation system**

***Keywords*—visual navigation system; unmanned aerial vehicle; servomotors; stepmotors computer vision; primary frame; secondary frame; driver; microcontroller**

I. INTRODUCTION

Visual navigation system is an important navigation module of unmanned aerial vehicle (UAV). It is required to conduct numerous laboratory tests that are not dependent on weather conditions and lengthly flight preparations to determine the effectiveness of its work, to exclude design errors, and to select the optimal computational algorithms. This paper aims to develop UAV flight simuation test stand for the visual navigation system. Basic designs for the hardware and electronic parts of the stand are presented.

The problem of high-precision aircraft navigation without any use radio signals recently received high relevance due to the active use of drones in radio interference zones where is a possibility of the Global Positioning System (GPS) signal loss, which in turn can lead to the UAV accident.

Visual navigation system (VNS) in the case of GPS signal suppression can serve as inertial navigation system corrector using only visual cues on the ground. Despite its shortcomings, namely the inability to work in case of unfavorable weather conditions (fog, rain) or the lack of clearly defined guidelines (water surface, desert), visual navigation is a promising way to solve navigation tasks. However, nowaday there are no serially produced VNS. The reason lies not so much in the lack of technical means, but primarily in the difficulty of tuning the mathematical software that the on-board computer implements.

II. VISUAL NAVIGATION SYSTEM CONSTRUCTION PROBLEM

To build a visual navigation system, it is necessary to:

1) Develop the mathematical software of the image processing subsystem in order to detect visual reference points. Mathematical software includes the following tasks:

- preprocessing;
- segmentation;
- the definition of points belonging to the descriptor boundary;
- contour analysis;
- correlation analysis;
- comparison of the detected reference correlation function with the correlation functions corresponding from the database of references with predetermined coordinates, and determination of the current UAV coordinates.

2) Develop a mathematical software of the navigation subsystem in order to determine the current coordinates of the UAV:

- special points detection (Scale invariant feature transform / Speed up robust features);
- match determination between the feature points of the same descriptor on the different images (RANSAC):
- determination of the current UAV coordinates.

3) Optimal selection of a technical means complex:

- camera;
- onboard computer;
- controlled gyro suspenssion;
- lithium-ion battery for autonomous power supply system;
- controller of gyro suspension control.

III. VISUAL NAVIGATION SYSTEM BLOCK DIAGRAM

As follows form the block diagram, the necessary selection of the VNS components includes: an autopilot, an onboard microcomputer, a battery for their autonomous power supply and a camera (Fig. 1).

The camera connects to the onboard computer via the universal serial bus (USB) interface; single photograph period is given. If a gyro stabilized suspension is used, control signals are applied to the suspension from the autopilot. Otherwise, during the photographing onboard computer sends a request to the autopilot for the angular parameters of the UAV orientation and carries out the image correction (alignment). Next, the onboard computer detects special points, determines the offset of the special points, calculates the current UAV coordinates and sends them back to the autopilot by feedback. Feedback is conducterd via the serial interface (SPI).

The visual navigation system can be powered by the main UAV battery. It’s accomplished by installing a DC/DC step-down converter. Voltage can also be supplied from its own lithium-ion battery, especially in cases when the UAV uses a motor such as internal combustion engine.

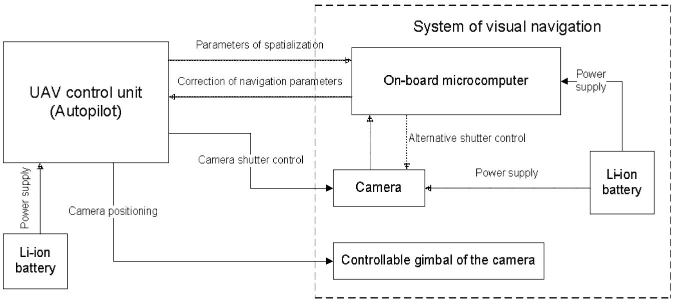

Fig. 1. Block Diagram of the Visual Navigation System.

IV. TECHNICAL MEANS COMPLEX

Use of gyro stabilized optical devices ensures the retention of the captured target in the frame and gives independence from the UAV spatial position during the flight (Figs 2 and 3).

Fig. 2. Camera without gyro stabilization.

Fig. 3. Cameras with gyro stabilization.

On-Board Computers (Figs 4, 5 and 6):

- Quad Core 1.2GHz Broadcom BCM2837 64bit CPU;
- 1GB RAM;
- VideoCore IV 3D GPU;
- BCM43438 wireless LAN;
- Bluetooth 4,1;
- Bluetooth Low Energy (BLE) on board;
- 40-pin extended GPIO;
- 4 USB 2 ports;
- 4 Pole stereo output and composite video port;
- Full size HDMI;
- Ethernet port;
- CSI camera port for connecting a Raspberry Pi camera;
- DSI display port for connecting a Raspberry Pi touchscreen display;
- Micro SD port for loading your operating system and storing data;
- Upgraded switched Micro USB power source up to 2.5A.

Fig. 4. Raspberry Pi 3 model B.

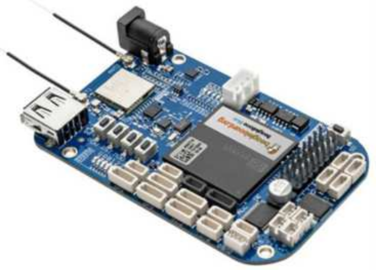

Fig. 5. BeagleBone Blue.

- CPU: Octavo Systems OSD3358 1GHz ARM® Cortex-A8;
- 512MB DDR3 RAM integrated;
- On-board 4GB 8-bit eMMC flash storage;
- 2×32-bit 200-MHz programmable real-time units (PRUs);
- ARM Cortex-M3;
- Integrated power management;
- Battery: 2-cell LiPo support with balancing, 9-18V charger input;
- Wireless: 802.11bgn, Bluetooth 4.1 and BLE;
- Motor control: 8 6V servo out, 4 DC motor out, 4 quadrature encoder in;

- Sensors: 9 axis IMU, barometer;
- Connectivity: HighSpeed USB 2.0 client and host;
- User interface: 11 user programmable LEDs, 2 user programmable buttons.

Easy connect interfaces for adding additional sensors such as: GPS, DSM2 radio, UARTs, SPI, I2C, 1.8V analog, 3.3V GPIOs

Fig. 6. ODROID XU-4.

Characteristics:

- CPU: Samsung Exynos5422 Cortex™-A15 2Ghz and 8-core Cortex™-A7;
- Mali-T628 MP6 (OpenGL ES 3.0/2.0/1.1 support and full profile OpenCL 1.1);
- 2GByte LPDDR3 RAM PoP;
- Flash drive eMMC5.0 HS400;
- 2 x USB 3.0 Host, 1 x USB 2.0 Host;
- Gigabit Ethernet port;
- HDMI 1.4a interface;
- Size: 82 x 58 x 22mm (including cooler).

Empirically, this onboard computer has the maximum performance available on the market today.

V. VISUAL NAVIGATION SYSTEM ALGORITHM STRUCTURAL SCHEME. FORMULATION OF THE PROBLEM

Matrix of **N** × **M** pixels is given. It is necessary to detect special points on this image using computer vision approaches.

- Special points and their descriptors are detected.
- By descriptors match corresponding feature points are selected.
- An image transformation model is constructed on the basis of a set of corresponding special points, using which one image can be obtained from another.

A special point is a point of the depicted object, which is likely to be found on another image of the same object.

A method for extracting special points from an image will be called a detector. The detector must ensure the invariance of detecting the same special points with respect to transformations of images. Methods for matching feature points on images.

A descriptor is the identifier of a special point that distinguishes it from the rest of the special points. In turn, descriptors should ensure the invariance of finding the correspondence between the special points relative to image transformations.

VI. CLASSIFICATION OF THE METHODS OF SEARCHING FOR FEATURE POINTS

Methods of separation of feature points in the image (detectors, descriptors):

- SIFT;
- SURF;
- ORB;
- MSER;
- AKAZE.

In this paper, the most effective methods for finding special points are considered: SURF and SIFT.

VII. COMPARATIVE ANALYSIS SEARCH METHODS FOR SPECIAL POINTS

In this part we will consider base algorithms for detecting and building descriptors of feature points of the image.

1) SIFT algorithm has the following disadvantages:
   - not all obtained special points and their descriptors will be determined in subsequent photographs, which affects the further image comparison;
   - the algorithm does not work in the following cases:
     - the lighting conditions are different (for example, day / night);
     - the object has a reflective surface (usually cars, mirrors);
     - the objects has a pronounced 3-D structure;
     - the object has self-similar structures (as a result, "true" inconsistencies arise);
     - camera angle for different pictures is significantly different.
2) SURF algorithm has the following disadvantages:
   - Object is not distinguished from the background during image processing (special points may be inside the object, as well as on the background, in the points on object-background boundaries);
   - the algorithm does not work well for objects of simple form and without a pronounced texture (in such objects

the method is not likely to find special points. The points will be found either on the boundary of the object with the background or even only on the background).

3) MSER algorithm has the following disadvantages:
- not fully invariant to scale (scale normalization occurs depending on the size of the selected areas;
- sparseness function use, which requires the presence of contrasting areas on the image.

A. SURF (Speed Up Robust Features) algorithm

1) Form a digital image model in the form of a pixel matrix.
2) Translate this matrix into integral representation of the image for faster computation.
3) Determine Hessian matrix elements which are computed as a convolution of (sum of products).
4) Calculate pixel Hessian when the filter scale is changed.
5) Find the local maximum of the Hessian.
6) Find the orientation of a special point.
7) Calculation the descriptor of a special point.

B. SIFT Algorithm

1) Form a digital model of the image in the form of a pixel matrix.

2) Construct the Gaussian pyramid and difference of Gaussians (DOG) to search for special points.

3) Check the extremum point suitability for the role of the key.

4) Find the orientation of a special point.

5) Construct the descriptor for a special point.

C. RANSAC algorithm

The input is:

1) Set of initial data *X*.
2) M function for calculating the parameters of \ theta model *P* over the set of n-points.
3) Evaluation function *E* of the points correspondence to the obtained model.
4) The threshold *t* for the evaluation function.
5) The number of iterations of the method *k*.

This algorithm consists of one cycle, each iteration of which can be logically divided into two stages.

First stage:

- points selection and model calculation;
- random selection of n different points of the original set;
- calculation \ theta model *P* parameters based on selected points using the *M* function. Constructed model is called a hypothesis.

Second stage:

- for each point its conformity to this hypothesis is verified by evaluation function *E* and the threshold *t*;
- each dot is marked as inlier or outlier;
- after verifying all points, it is checked whether the hypothesis is the best at the moment, and if it is, then it replaces the previous best hypothesis.

The result of the work is:

1) Parameters or \ theta model *P*.
2) Original data points are marked as inliers or outliers.

Coordinates determination algorithm:

To calculate the coordinates of significant points on the reference image in the take-off point, it is necessary to determine the image pixel distance from the center, the coordinates of which are known to us, to the filtered points RANSAC method.

Pixel distance (offset) d between points $A(x_1, y_1)$ and $B(x_2, y_2)$ in the plane determined by the formula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}.$$

The obtained distance is equal to the number of pixels which must be multiplied by the current scale in meters camera matrix at a known height and focal length.

The tangent of the angle between the segment and the positive direction of the axis *Ox* is determined by the formula (this angle is measured from the axis *Ox* counterclock-wise).

$$\mathrm{tg}\,\varphi = \frac{y_2 - y_1}{x_2 - x_1}.$$

Determined by this formula tangent is the slope of the straight line. Additionally, it is necessary to attach to each shot a course on the compass read from the autopilot.

Having determined the distance to the points and the angle, it is necessary to recalculate the latitude and longitude coordinates, counting them as the destination points relative to the central point of the image, the distance from which we know from the following formula:

$$\varphi_2 = \mathrm{a}\sin\left(\sin\varphi_1 \cos\delta + \cos\varphi_1 \sin\delta \cos\theta\right),$$

$$\lambda_2 = \lambda_1 + \mathrm{a}\tan 2\left(\sin\theta \sin\delta \cos\varphi_1, \cos\delta - \sin\varphi_1 \sin\varphi_2\right),$$

where φ_1 is the current longitude; φ_2 is the longitude of the target point of the mission is loaded in memory UAV autopilot; λ_1 is the current latitude; λ_2 is the target latitude; R is the Earth radius (average radius of 6.371 km);

As the starting point of the origin, the center of the first image at the takeoff point is assumed. Thereafter found landmarks in the image will be associated with this point and its GPS coordinates. Extrapolating the received coordinates from the pictures, and reading from the memory of the autopilot the recorded coordinates of the points of the flight plan auto mission, we obtain the coordinates of the target point and the current one. Using haversine formula consider the distance to the target point through the great circle arc approximating the Earth sphere in meters.

For some autopilots, you also need to calculate the heading for the target point for navigating the formula:

$$\theta = a\tan 2\left(\sin\Delta\lambda\cos\varphi_2,\ \cos\varphi_1\sin\varphi_2 - \sin\varphi_1\cos\varphi_2\cos\Delta\lambda\right),$$

where φ_1, λ_1 is the longitude / latitude of the current point; φ_2, λ_2 is the latitude / longitude of the calculated point (destination) of the mission loaded from the memory of the UAV autopilot.

VIII. CHOICE OF VNS PARAMETERS DEFINITION

A. *SIFT*

1) The number of required special points (nFeatures).
2) The number of octave layers (nOctavesLayers = 3 default).
3) Threshold level (contrastThreshold = 0.04 by default).
4) EdgeThreshold = 10 by default.
5) Sigma = 1.6 by default.
6) The descriptor normalization factor = 0.2 by default.

B. *SURF*

1) Hessian threshold value.
2) Number of octaves (nOctaves = 4 by default).
3) The number of octave layers (nOctavesLayers = 2 by default).
4) The magnitude of the height scale (sigma).
5) Size and type of the filter kernel.

C. *RANSAC*

The threshold value for the hitting of the feature points (sigma).

IX. TEST STAND BLOCK DIAGRAM

Additional problems resolved by stand (Fig. 7):

1) UAV auto missions flight simulation with an existing or template mission flight plan.
2) Various UAV type simulation (multirotor, airplane, flying wing, hybrid).
3) Simulation and correction of the weather conditions (haziness, cloudiness, stronger light, darkness).
4) Analysis and correction of image distortions (affine, scale, distortion, etc.).
5) Search of the strategic military objects in a predetermined pattern.
6) Search for local features and objects on satellite images.
7) Map digitizing and crosslinking from UAV obtained images (optimum overlap).
8) Current UAV coordinates determination via VNS.
9) Accuracy evaluation and verification of existing software algorithms for the local features search on a relief map (creating own detectors, descriptors).
10) Creating a big data set for deep learning, and training artificial neural networks.
11) Manage payload in the template mode of the auto mission, or with gyro stabilization.
12) Development and testing of own software algorithms of the UAV visual flight navigation system.
13) VSN assessment by existing quality assessment metrics and development of own fault-taking QA metric (~taking into account existing metrics faults).

Description of the stand tuning structural scheme (Fig. 8).

This scheme assumes the availability of three servo motors, which provide three axial positioning of the platform with payload in three planes. To determine the angular position of the platform with the camera an inertial navigation system is needed which comprises of: 3-axis gyro, accelerometer and 3-axis axial compass. PIXHAWK autopilot INS is used as Inertial Navigation Systems as PIXHAWK has active unit in its structure and is located above the camera on the back side of the carriage. In this autopilot typical UAV control commands are used for the most realistic simulation.

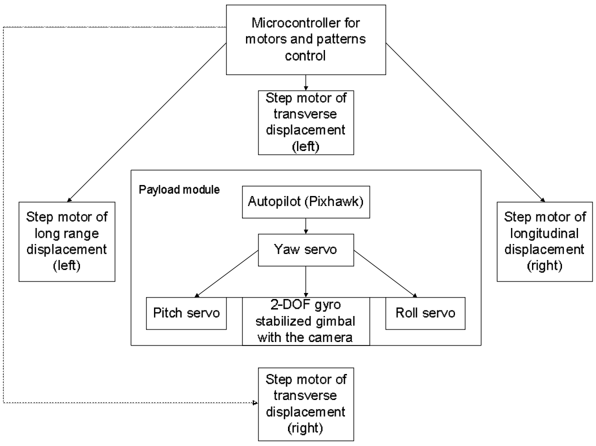

Fig. 7. Block diagram of the test stand.

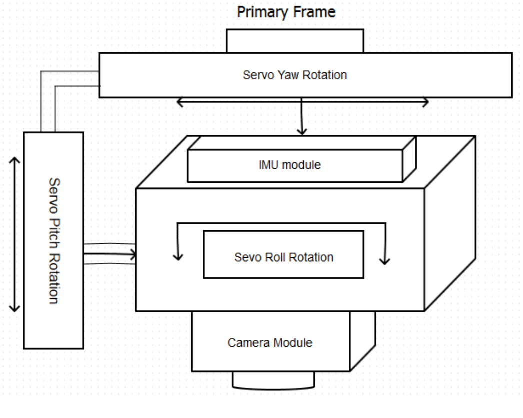

Fig. 8. Primary frame block diagram.

The angular position of the camera module is controlled through the pulse width modulation channel of the autopilot, whose outputs are connected to servo motors and a template script for changing flight angles is started.

As the inertial measurement unit (IMU) a real autopilot has been selected, having the module in its composition, as well as a typical UAV control commands for the most realistic simulation.

Interface of camera data exchange of simulation stand's visual navigation system with an external stationary personal computer (PC) is USB. Data from the onboard computer or the laboratory PC to the autopilot is transferred via the normal USART/SPI serial interface, for the feedback and adjustment of the UAV's position in space. To simplify the design of the frame and get rid of rotating contacts, a radio module is installed on the primary frame next to the autopilot. In this case, the data is transmitted via conventional radio channel.

The angular position of the camera module is controlled through this automatic radiocommunication channel of the autopilot, whose outputs are connected to servo motors and a template script for changing flight angles is started.

This is the basic frame bearing structure the base of which consists of beams, as well as four troughs which moves wheels secondary body frame in the planes of the O_x and O_y. Drive scheme creates two axial displacement comprises 4 servo motors pairwise connected to two axes. This allows a 2-fold increase in torque and consequently the payload of the simulation bench.

Control is provided by using four DRV 8834 drivers and STMf32103C8 central microcontroller.

The electrical wiring diagram looks like this (Fig. 9).

Linear traversal of the primary frame over the relief map is carried out at the choice of one of the pre-loaded into the microcontroller templates of geometric figures: passage through a square, a triangle, a rectangle and more complex ones. Thus, the maximum similarity to the actual loadable flight plan of the auto mission in the UAV autopilot is achieved.

As a result of joining the primary and secondary frames, we obtain an imitation stand with 5 degrees of freedom, which have the following form (Fig. 10).

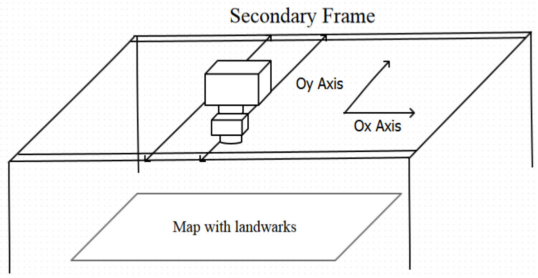

Fig. 9. Secondary frame block diagram.

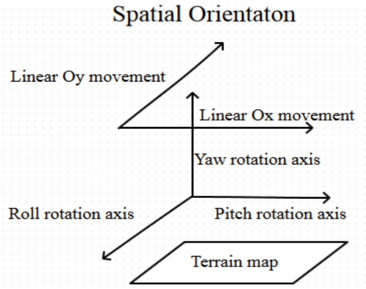

Fig. 10. Spatial dispalcement of the payload.

X. STAND CONFIGURATION ALGORITHM

Visual navigation system configuration on the test stand problem statement.

To determine the optimum values of configurable parameters of visual navigation system r_i, $i = 1, \ldots, 12$ as a criterion.

$$I=\sum_{i=1}^{N}\left[\left(x_i-x_{ir}\right)^2+\left(y_i-y_{ir}\right)^2\right],$$

where x_i, y_i are current UAV coordinates calculated using the visual navigation system; x_{ir}, y_{ir} are reference coordinates determined by the absolute position of the carriage.

The algorithm for determining optimal parameters of the visual navigation system

The algorithm for determining the optimal parameters of the visual navigation system is based on the use of a genetic algorithm.

XI. CONCLUSION

Developed mathematical software of the visual navigation system, analysis of which made possible to define many configurable parameters that are impossible to set in the field.

Developed and assembled test stand for the visual navigation system configuration which comprises: step motors, step motors drivers, microcontroller for step motors control, inertial navigation system, camera, 3-axial gyro stabilized suspension servos, onboard computer, radio data transfer channel; which provides UAV flight simulation with the location photo shooting and current position determination.

Developed the stand mathematical software to solve the optimization problem of the optimal visual navigation system parameters determination based on the genetic algorithm utilization, which will greatly reduce the time and material costs for VNS configuration.

REFERENCES

[1] V.A. Rogoshyn, V.M. Sineglazov and N.K. Filyashkin, Flight Navigation Complexes of Aircrafts. Kyiv, NAU, 2004, p. 238. (in Ukrainian).

[2] Flight controller Pixhawk http://pixhawk.org

[3] Paul D. Groves, Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems. Second Edition (Artech House Remote Sensing Library) 2nd Edition, 2008, p. 503.