International Micro Air Vehicle Conference and Flight Competition (IMAV2013)
17-20 September 2013, Toulouse, France

Vision-Based Navigation Solution for Autonomous Indoor Obstacle Avoidance Flight

Kirill E. Shilov[1], Vladimir V. Afanasyev[2] and Pavel A. Samsonov[3]

[1] Moscow Institute of Physics and Technology (MIPT), Zhukovsky, Russia
`aviaks.kirill@gmail.com`
[2] Moscow State University, Moscow, Russia
`vvafanasjev@mail.ru`
[3] Moscow State University, Moscow, Russia
`aspsnaky@gmail.com`

Abstract

This paper describes scientific and technical approaches as well as the results obtained during the research and flight tests aimed at enhancing MAV autonomy by the means of computer vision sensing. An inspiration for the research and development of video-based navigation system was given by the task to autonomously overcome the distance in a corridor containing randomly set obstacles. Being located indoor in GPS-denied space the corridor size is slightly more than a vehicle size. Medium-sized hexacopter was chosen as an airframe. Onboard electronics consists of two parts: flight control module and video processing module. Flight control module is based on author‘s custom SmartAP autopilot (presented at IMAV2012) and provides low-level services such as hexacopter stabilization and point-to-point flight routing. Video processing module is responsible for position estimation, obstacles determination and further flight trajectory calculation. Flight control module guides the MAV in order to follow the desirable path generated by video processing system. Technical details of the setup and computational solutions are presented in the paper.

1 Introduction

The typical mission to be completed consists of start zone, landing zone, two corridors and randomly set obstacle. MAV should overcome the distance without pilot control, including autonomous take off, randomly set obstacles detection and avoidance, autonomous landing.

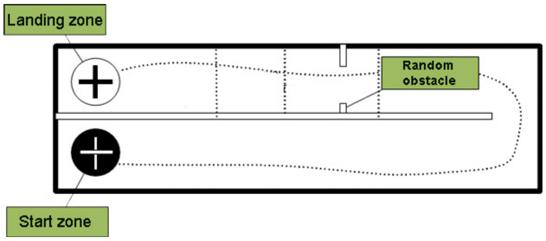

Figure 1: Scheme of the mission to be completed

Start zone and landing zone are marked with the big circles of black and white color respectively with the plus signs. The diameter of the circles is 3 meters. Corridor length is 50 meters, width is 8 meters. Corridor walls are painted in light-blue color.

Figure 2: Corridor with landing zone circle

2 General approach

In order to perform the flight described MAV must have both Flight Control System (FCS) which is responsible for MAV stabilization and control and Computer Vision System (CVS) for obstacles detection and path planning.
SmartAP autopilot [1] was chosen as FCS module. CVS module is based on laptop‘s motherboard instead of single-board computer because computational resources are critical for the task and can be provided only by high-end CPU and GPU.

3 FCS module

Flight control system hardware contains MCU, sensors and external ports for peripheral connections. This allows providing sustainable navigation solution.
The key features of the FCS are:

- Powerful microcontroller 72MHz ARM Cortex M3 MCU [2]
- 9-Degrees Of Freedom Inertial Measurement Unit [3]
    - 3-axis accelerometer
    - 3-axis gyroscope
    - 3-axis magnetometer [4]
- Pressure sensor for altitude measurement [5]
- Differential pressure sensor for air speed measurement
- GPS receiver for global position determination [6]
- Wireless data channel for two-way telemetry
- SD-card for in-flight data logging
- USB port for PC/Laptop connection for firmware uploading, debugging and testing routines
- PWM inputs/outputs to read signals from standard RC receiver and send them to motors/servos
- Several ADC channels for battery voltage/amperage monitoring
- Analog inputs for ultrasonic rangefinders connection

The FCS software architecture consists of several levels: from the lowest - STM microcontroller libraries to the highest flight control logic. CMSIS Library from STM makes possible to create higher software hardware-independent. The second level is STM Library which provides high-level functions for microcontroller peripheral communications. Next level includes drivers for sensors readings and actuators control. The highest level consists of functions responsible for stabilization, navigation, flight control and digital signal filtering where its necessary. Additionally, the highest level includes GCS interface functions based on MAVLink communication protocol library. The entire software is run on FreeRTOS (Real Time Operating System) developed especially for embedded applications which provides functionality for effective tasks and resources control being crucial for such devices. Every function belongs to its priority group corresponding to its importance for safety guideline.
Therefore, FCS block diagram can be presented in the following way:

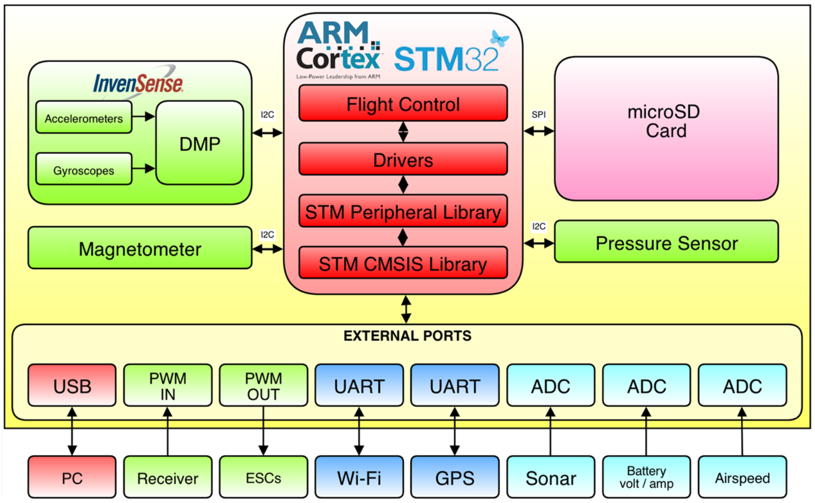

Figure 3: FCS block diagram

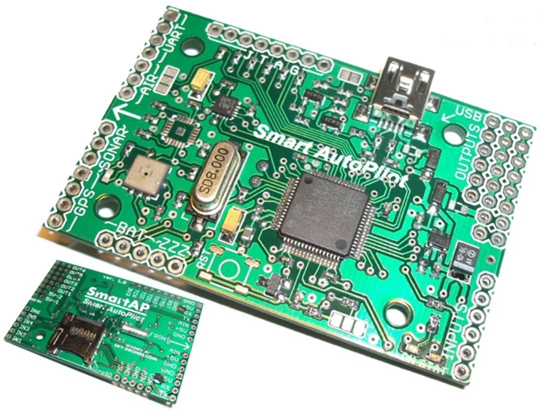

Figure 4: Assembled FCS hardware

4 CVS Module

On-board video-processing hardware contains:

- Laptop motherboard (Asus N46VZ)
- High-end mobile CPU Intel Core i7 3670QM
  Middle-end GPU NVidia GT650M
  8 GB of DDR3 system memory and 2 GB of GDDR5 video memory
- 3 industrial high-speed USB 3.0 cameras (Ximea MQ013CG-E2)
  1.3 MPixel resolution
  60 FPS

CPU and GPU computational resources are used for high-speed 3D map generation. Cameras with fixed positions form a number of stereograms. This allows scene depth calculation.

Flight control and video processing modules are connected via USB. Flight control module sends inertial data information and receives back adjusted position, velocity and desired position for the following iteration. Additionally, airframe is equipped with 4 ultrasonic rangefinders. One of them is facing downwards (for altitude estimation), the other three are facing straight forward and are responsible for collision avoidance in case of noisy video gathered data.

The video processing module receives image frames and sensor data. Sensor data is used as initial approximation for the vehicle position refinement. First, the frames are preprocessed in order to get points for effective local 3D reconstruction. Each frame gives a small number of reliable 3D object points that supplement and refine the whole 3D map. Restricted and permitted areas are marked on this map and are renewed during the flight. The desired trajectory to the destination is calculated using the map and the nearest target point is transmitted to the flight control module.

On this basis, flight control module and video processing module form sustainable electronic platform allowing MAV to fly indoor autonomously.

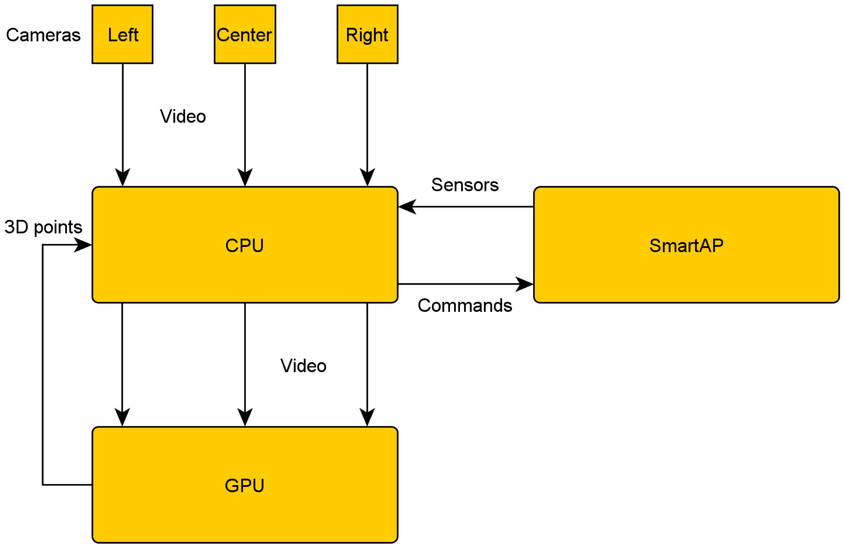

Figure 5: CVS Hardware

5 Airframe

Since the laptop motherboard is quite heavy it was decided to use hexacopter airframe instead of traditional quadcopter to allow higher carrying capacity. The motherboard is installed at the top of the hexacopter. FCS is mounted on the motherboard. Hexacopter is assembled to fly in + mode meaning that the one arm is facing forward. Three video cameras are mounted to the end of the arms. Total weight of the airframe is about 2700g.

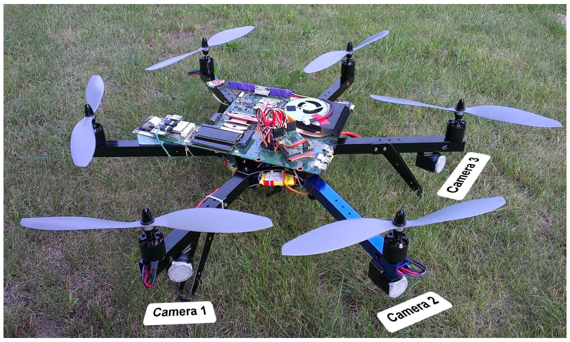

Figure 6: Hexacopter airframe

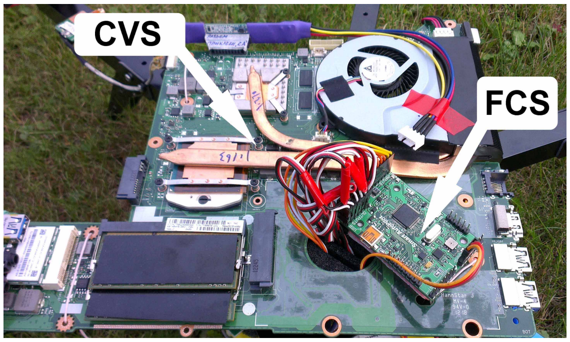

Figure 7: Close-up of CVS and FCS

6 Video processing

The emphasis is done on 3 cameras system mostly because of the laptop motherboard has only 3 USB 3.0 ports. Though for the algorithm presented even 2 cameras are sufficiently, 3 cameras will provide wider angle of view.

The video processing system is based on handling several high-fps video streams. Each of video streams has outputs 50 FPS. So, motherboard is intended to process 150 frames per second. Being uncompressed this data stream is approximately 600 MB/s. CPU is able to perform only several operations by pixel which is not enough That‘s why GPU instead of CPU is used.

To provide high MAV flight speed high-speed megapixel cameras were chosen. The higher frame rate allows less lookup size during each next frame processing in case of inaccurate position prediction or its absence.

Some technical parameters of the camera are presented below:

- Physical frame rate: 60 FPS, 50 FPS actual
- Resolution: 1280x1024
- 8 bit RGB color
- CMOS sensor with global shutter
- Lens: 6 mm CS, 90 deg. horizontal, 60 deg. vertical angle of view
- USB 3.0 interface

Camera specifications impose restrictions on the use of algorithms:

1. The high frame rate leads to low exposure time. If the frame rate is 50 FPS, exposure time for each frame must be less than 20 ms. On a sunny day 0.5 ms exposure time is required, on a cloudy day exposure should be 5-10 ms. The indoor artificial lighting is the limit for such cameras at a high frame rates. The images become dark, though, image processing is still able to handle camera data.

2. Asynchronous mode for higher frame rates must be applied. There is a possibility to synchronize the cameras in hardware, but this leads to a situation when one camera waits for another each frame, so overall frame rate degrades. It was decided that the higher frame rate is more important for the task rather than simultaneity of the frames.

6.1 Preparation part

The algorithm initializes with camera positions and rotation calibration. By now, this part is performed manually. Recalibration should been done is case of cameras were moved. After that, distortion parameters acquisition is performed. The automatic fitting of Broun model using a checkerboard is performed for every lens type. Normally, it is made only once for each camera.

Finally, the exposure adjustment algorithm starts. The exposure value is selected to provide normal average brightness in the frame. Firstly, optimal exposure is measured for each camera. After that the medium exposure is calculated and set for all cameras to have the same brightness for all images. At the moment, only static exposure calculation is implemented. Having been set once, it doesn‘t vary a lot during the flight. However, dynamic exposure adjustment is also planned to be done.

6.2 Image processing pipeline

Images are acquired from each camera in separate threads. On the maximum FPS each camera thread uses almost 100% load of a single CPU core. The most of processor time is utilized in camera driver. So, we have only one core left for all remaining tasks. Then the image is passed from CPU to GPU through PCI-E 16x 2.0 interface. The CPU task is to get the image from camera driver and pass it to GPU. This is a necessary minimum that should be done with such huge data. In this manner the load from CPU put off as much as it is possible.

The distortion in the image is corrected with pre-built correction map that contains new coordinates of each pixel. The white balance is corrected right after distortion. Color correction parameters are set only once for each flight.

After that the image is adaptively binarized to find the cross in the image. All image is processed by Gaussian blur and subtracted from initial image. The difference image is then binarized using a static threshold.

The feature points are allocated in the binary image. A fast feature detection algorithm is used. Then the feature points 3D positions are reconstructed using the correspondence lookup on camera pairs forming stereograms. Features are also matched on sequential images from the same camera to get 3D point correspondence and calculate the new MAV position. Due to the fact that images are shot asynchronously by cameras and MAV has non-zero speed we get a variable addition to a constant stereogram configuration. Variations in stereogram parameters may lead to significant errors in stereo reconstruction. To minimize these errors stereogram camera positions and orientation are corrected using extrapolation of the previous MAV dynamics as well as inertial sensor readings.

7 MAV position estimation

The key idea of position estimation algorithm is to combine data from all sensors. All sensors provide different data with different accuracy and different types of errors.

Sensor	Accuracy factors	Accuracy
Video	Camera calibration accuracy Distortion correction quality Processing time restrictions	Up to 1 cm at 5 m distance (depth map)
Gyroscopes	Vibration	0.001 deg/s
Accelerometers	Airframe vibrations Update frequency Filtration technique	0.01-0.1g
Ultrasonic rangefinders	Sound interference Motors air flow Surface type	Up to 1 cm at 1-2 m distance

Every type of sensor has its own failure conditions and reliability. Normally, all sensors give correct data, but one of the sensors can give data with errors. Every new piece of data from some sensor is checked for consistency. If current sensor does not form a consensus with other sensors, it is marked as unreliable and is not used to form the final position estimation.

The specific properties of every sensor are also taken into account to use its data more efficiently. Video as the most precise and reliable sensor is used to make check points in MAV trajectory.

8 Control

FCS is responsible for low-level functions (stabilization, point-to-point flight) while CVS is responsible for obstacle detection and avoidance and path-planning.

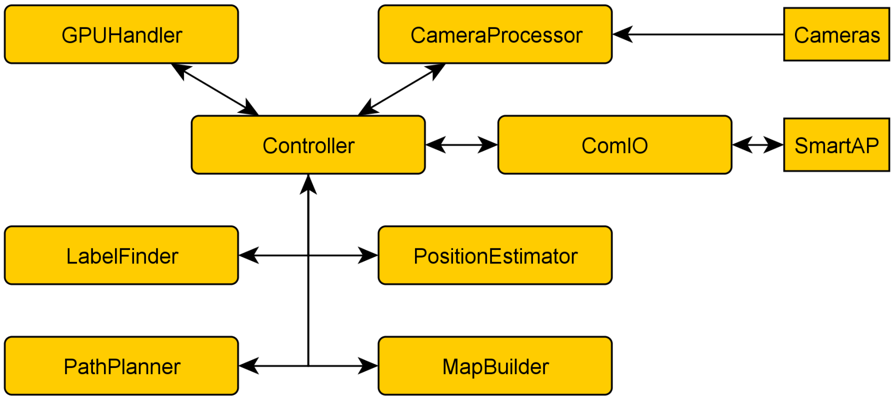

Figure 8: CVS Software

Basically, it can be subdivided into three groups:

- Targeting module
- Path planning module
- Interface with FCS

FCS path-following is done using PI-PI controller which diagram can be presented as follows:

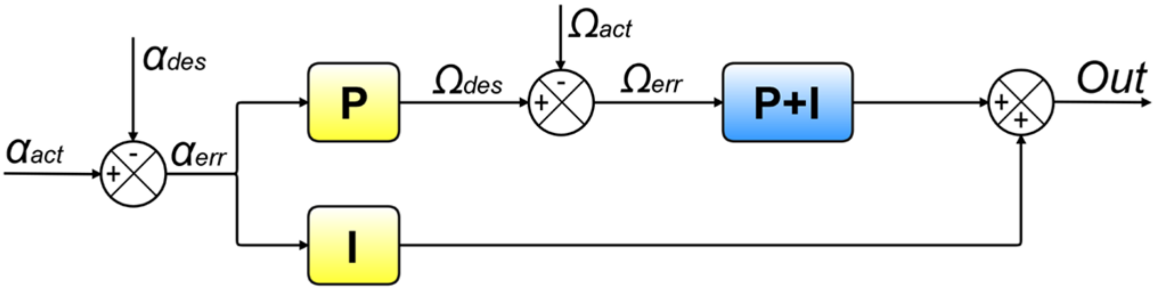

Figure 9: PI-PI Control loop for FCS

9 Conclusion

By the moment airframe had been already assembled and performed several flight tests. Computer vision system algorithms have been validated in simulation and confirmed the ideas described in the paper. Hexacopter is able to demonstrate autonomous take off and landing with position hold based on the information received from computer vision module. Fully autonomous flight in GPS-denied environments and on the fly 3D map generation are the subjects of further research in this project.

10 Reference

[1] MAV Autopilot for Commercial and Research Purposes, Kirill Shilov, Grigory Lazurin, IMAV 2012, Braunschweig, Germany, 3-6 July 2012

[2] High-density performance line ARM-based 32-bit MCU, STMicroelectronics, http://st.com/

[3] MPU-6000 and MPU-6050 Product Specification Revision 3.2, Invensense, http://invensense.com/

[4] 3-Axis Digital Compass IC HMC5883, Honeywell, http://honeywell.com/

[5] BMP085, Digital, barometric pressure sensor, BOSCH, http://www.bosch-sensortec.com

[6] UBlox LEA-6, u-blox 6 GPS Modules Data Sheet, http://www.u-blox.com/