Comb Studio's Autonomous Aircraft for the IARC 2014

Xiang He, Dongze Huang, Jinpeng Yang, Yun Sang,
Qi Wu, Kun Yan, Dongyao Zhang, Qing Lin
Beihang University

ABSTRACT

This paper describes the details of an autonomous aircraft capable of exploring cluttered indoor areas and interacting with object in the environment without relying on external navigational aids. An integrated visual navigation method providing relative position, velocity, and attitude information is introduced. Multi-ground-object are detected by HOG-Based SVM. Mission planning section applies the velocity-obstacle method to program for the vehicle drive away ten ground robot to a set sideline. The vehicle is intended to be *Beihang University Comb Studio*'s entry for the International Aerial Robotics Competition in 2014.

1. INTRODUCTION

1.1 Statement of the Problem

The IARC 2014 requires the teams to apply completely autonomous aircraft. In the mission IARC 2014, a square arena is marked on the ground in an indoor GPS-free arena. This square arena will be 20 meters on each side. The boundary shall consist of wide white lines bounding the sides of the square arena, with a wide red line on one end, and a wide green line on the other end as shown in the figure. Ten iRobot Create programmable ground robots will be placed at the center of the arena. These ground robot is programmed to run in a changeless rule with small unpredictable variation on its direction. The ultimate goal of the aerial vehicle is to chasing the ground robot and drive them away to the set green line by slightly touching the surface of the robot to make it turn 45 °or obstructing in the route to make it turn around.

This will challenge teams to demonstrate two new behaviors that have never been attempted in any of the past six IARC missions. First, interaction between aerial robots and moving objects, specifically, autonomous ground robots, challenges aerial vehicle's ability of control and recognizing the moving object. Second, navigation in a sterile environment with no external navigation aids such as GPS or large stationary points of reference such as walls challenges the aerial vehicle to get good navigation using visual only. The whole task must be finished within 10 minutes, which also challenges the mission planning to be smart enough to make a strategic decision.

1.2 Conceptual Solution to Solve the Problem

The Comb Studio of Beihang University takes a quadrotor to accomplish the task. An integrated visual navigation method is used to provide relative position, velocity, and attitude information. Multi-ground-object are detected by HOG-Based SVM. Mission planning section applies the velocity-obstacle method to program for the vehicle drive away ten ground robot to a set sideline. The Overall System Architecture is shown below.

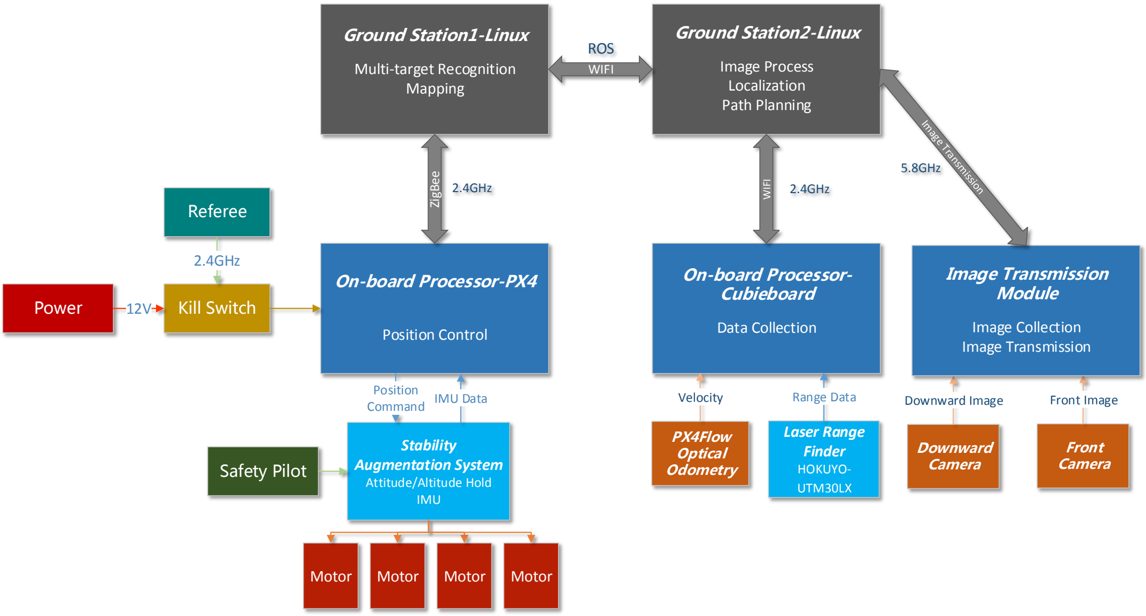


Figure 1. Overall System Architecture

1.3 Yearly Milestones

This is the first time for the Comb Studio to participate in IARC mission 7. Both our technology and experience are immature. Therefore, in this year, our goal is to achieve a fully autonomous flight based on visual navigation and drive as much as 3 or more ground robot to the green line.

2. AIR VEHICLE

We choose quadrotor helicopter as the flight platform for the mission. The quadrotor airframe is chosen from the well-designed airframes which are available for DIY players. And the PX4 board is chosen as the flight control system. The PX4 is an independent, open-source, open-hardware project aiming at providing a high-end autopilot to the academic, hobby and industrial communities at low costs and high availability. It employs 168 MHz Cortex M4F CPU and runs a very efficient real-time operating system (RTOS), which provides a POSIX-style environment. For its fast data process ability and the open-source software structure, the PX4 is an ideal hardware system for us. So we choose it as our flight control board and develop the control algorithm for the competition. To satisfy the safety request, we design and manufacture protect shell for the quadrotor.

Figure 2 The Quadrotor

2.1 Guidance, Navigation, and Control

Stability Augmentation System

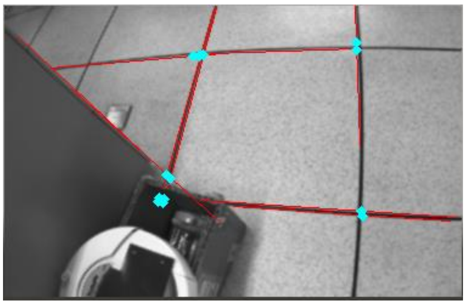
The vehicle is an unstable system, in order to make it move as expected, an attitude and heading controller is needed. There are three different levels of communication with the Pelican: sending the vehicle direct motor commands, sending the vehicle angles (pitch, roll and yaw), sending the vehicle waypoints. We chose the level 2, sending the vehicle direct angles, and we don't need to concern how the vehicle can achieve the desired angles. That is to say, the stability augmentation system is done by the Ascending Technology GmbH; we just need to develop our position controller based on it.

Navigation

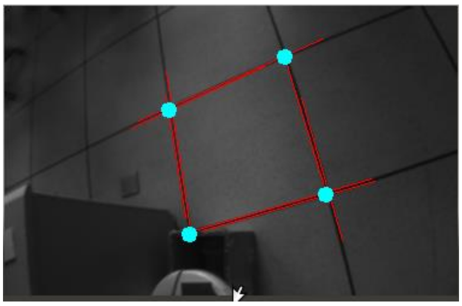
Limited by the large space of competition venue, the laser data based SLAM we used in the last mission is no longer useful. Information that can help the vehicle to localize itself is all lying on the floor. Thus, the visual based navigation system is to be established to fulfill the requirement. All of the currently existing visual localization method, however, suffers from its deficiency. Optical flow sensors[1] have shown its effectiveness on indoor navigation of mobile robot. But the fatal problem of optical flow's drifting[2] can only be solved by building a map using SLAM (Simultaneously Localization and Mapping)[3], which would drop the rate to nearly 30Hz. Navigation using artificial landmarks[4] is the earliest visual navigation method implement on mobile robot, which, also has a limitation when the number of visible artificial landmark is insufficient. We develop an integrated system that combines navigation from optical flow and artificial landmarks and these two will complement mutually.

1) Artificial Landmark Matching

As we described before, optical odometry need other method to correct deviation. Here we choose the basic method of using the information of artificial landmarks. According to the rules of 7th mission of IARC, the venue is divided into grids of 1m^2, and the frame is marked white, which facilitates the process of abstracting the artificial landmarks. We choose the intersection point of frame, from the yellow background of the venue. Edge detection is done by the Canny edge detection[5]. We set the threshold high to avoid the texture inside the grid. For the next step, the Hough transform is used to extract the line feature from the image after Canny edge detection. The image is 376×240 pxiel. Threshold of length in the Hough transform is set to be $\frac{1}{3}$ of the height of the image to avoid wrong line extraction. After the process above, we finally get the intersection point of ground tile, shown in Fig. 4a. Yet there is still some point dispersal due to the schismatic line extracted because of barrel distortion of camera. We filter the pose of these points and exclude the outliers. The average center is calculated to stand for those cluster of point. Fig. 4b shows the final extracted intersection point.



(a)



(b)

Figure 4a. Intersection point extraction without filter. Some point is extracted incorrectly. 4b. Filtered intersection point. Even an invisible point, which is overlapped by the box, is extracted correctly.

By the end of the image process part a serial of intersection point is extracted. Let $[u_i, v_i]$ be the pose of intersection point i in the image coordinate. The model of projection we have here is the pinhole model. Let point i present the point $\mathrm{P}_i = [x, y, z]^T$ in the camera coordinate. Base on the pinhole model projection equation, we have

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = A \cdot \begin{bmatrix} x_i'' \\ y_i'' \\ 1 \end{bmatrix} \tag{1}$$

Here $[x_i'', y_i'', 1]^T$ present the vector from the optical center to the projected point in the coordinate of camera before undistortion. A stands for the intrinsic matrix of camera get from calibration. Let the undistort vector be $[x_i', y_i', 1]^T$.

$$\begin{bmatrix} \mathrm{x_i''} \\ \mathrm{y_i''} \end{bmatrix} = \begin{bmatrix} \mathrm{x_i'}(1 + \mathrm{k_1 r^2} + \mathrm{k_2 r^4}) \\ \mathrm{y_i'}(1 + \mathrm{k_1 r^2} + \mathrm{k_2 r^4}) \end{bmatrix} \tag{2}$$

$$r^2 = x_i'^2 + y_i'^2 \tag{3}$$

$$\frac{x_i''}{y_i''} = \frac{x_i'}{y_i'} \tag{4}$$

We calculate only the radial distortion, and k_1 and k_2 represent the intrinsic distortion parameter also from camera calibration. Here we implement the Newton Iteration Method to solve the quintic equation. Let the rotation matrix from camera coordinates to world coordinate be RCW, then we have the equation

$$\begin{bmatrix} x_{iW} \\ y_{iW} \\ z_{iW} \end{bmatrix} = RCW \cdot \begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix} \tag{5}$$

The rotation matrix RCW is calculate from the IMU. $[x_{iW}, y_{iW}, z_{iW}]^T$ represents the projection vector in world coordinates. At this time we use the previous camera pose estimation as the center of current possible pose. The range of possibility circle is set related with time from previous certain localization to the current run. Let the camera position be $[x_C, y_C, z_C]^T$. The relationship between the camera position and the point P_i would be

$$\begin{bmatrix} x_C \\ y_C \\ z_C \end{bmatrix} + k \cdot \begin{bmatrix} x_{iW} \\ y_{iW} \\ z_{iW} \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{6}$$

Gain parameters is set k here. From the downward facing sonar, we assume z_C to be the ground distance, and $z = 0$, since the world frame is built on the ground. Then

$$k = -\frac{z_C}{z_{iW}} \tag{7}$$

Thus we get the possible point pose of P_i. If the camera does not travel too long without seeing any intersection point, the projection point of P_i should be not far from the artificial landmark. Then we set the position of P_i be the known pose in the world frames, bring it back into the former equation. We can get the camera position estimation of this run.

2) Optical Odometry

PX4FLOW is an integrated module board, combined with a CMOS camera, a sonar, a frame grabber processer and a CPU. Set as a downward facing equip, the module board provides data of the flow image, camera velocity, and ground distance. This board, by its own, can be a navigation sensor. The aerial vehicle could get the position feedback by performing an integration on the velocity from this optical flow odometry. Navigation result is shown below.

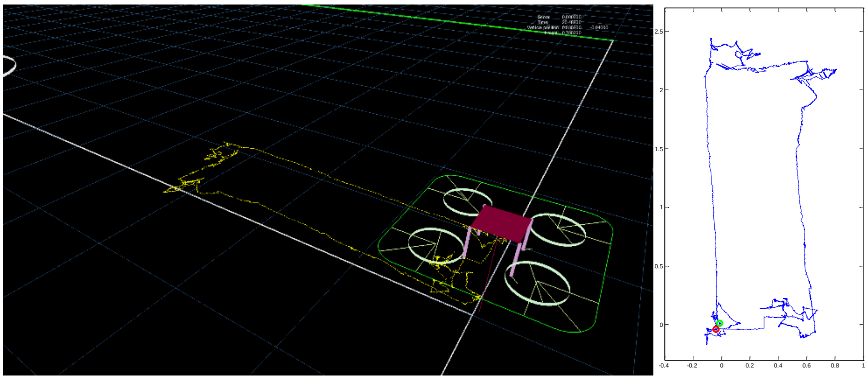


Figure 5. Navigation result.

Path Planning

The planning is based on airborne for UAV to track and intercept moving target in the presence of dynamic obstacles and considering the maneuvering characteristics of the Unmanned Aerial Vehicle (UAV). In the context of ensuring airborne sensor (camera) to keep track of the target, the approach can generate time-optimal speed commands for UAV in dynamic environment to intercept the target after discovering it.

The overall experimental framework is shown in Fig. 6. The experimental environment consists of three main parts. First, the sensor system is an information input of the planning method. Navigation module mainly provides status information of the aircraft, such as the position of the aircraft (P_A), the Velocity (V_A) and the attitude (A_A), etc. Visual tracking module is used to provide the position information of the targets (P_T) and obstacles (P_O).Second, the planner system is the implementation module of the algorithm. Finally, the aircraft model is concrete implementation part of the velocity command.

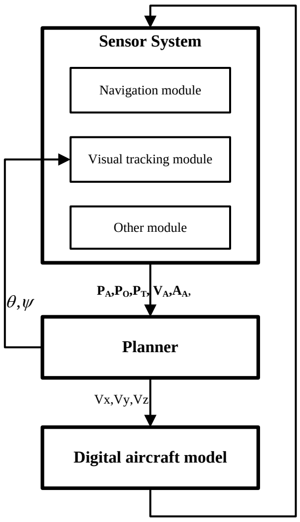


Figure 6 The overall experimental framework

When the target is found, we should start tracking intercepting and obstacle avoidance algorithm. If obstacles are in range of danger, VO should be calculated by using velocity-obstacle method as the velocity command prohibited area. Otherwise, just compute the velocity limit box (VL) and the rendezvous velocity set (RVS). Furthermore, all of the calculation results are synthesized to generate the horizontal velocity command.

The quadrotor motion characteristics in horizontal plane are similar to full-direction robot and the coupling of horizontal movement and vertical movement is not strong. Therefore, we just consider the horizontal movement of the aircraft. For the convenience of analysis, we defined the location of the aircraft as the coordinate origin. We define the relative position vector $\mathbf{r}$ connecting the aircraft to the target, the speed vector of target $\mathbf{V_T}$. We can draw a parallelogram shown in Fig. 7. The parallel-navigation rule states that the relative velocity $\dot{\mathbf{r}}$ between the aircraft and the target should remain parallel to $\mathbf{r}$ at all times. If meeting this rule, the distance between the interceptor and the target will be gradually reduced until a successful interception. Moreover, if the target is uniform motion, the guidance method will plan the time optimal results.

As shown in Fig. 7, we make a straight line parallel to vector $\mathbf{r}$ named rendezvous line (RL) represented by a dotted line. As can be seen from the figure, if the velocity vector terminal falls on the RL, the relative velocity between the aircraft and the target will be parallel to $\mathbf{r}$, which can be expressed by the following equation.

$$\mathbf{V_A} = \mathbf{V_T} + \beta\mathbf{r} \tag{8}$$

However, the velocity of the aircraft has a restricted range, i.e. aircraft velocity command has an upper limit decided by the performance of the aircraft, which can be expressed as follow:

$$\mathbf{V_{AM}} = \mathbf{V_T} + \beta_m\mathbf{r} \tag{9}$$

Easy to know, if we want to ensure that the aircraft is close to the target rather than away from it, we must ensure $\beta > 0$ (Assuming that the upper limit of the aircraft velocity command is greater than the velocity of the target). In addition, if we want to ensure that the velocity command is feasible, it is necessary to guarantee $\beta < \beta_m$.

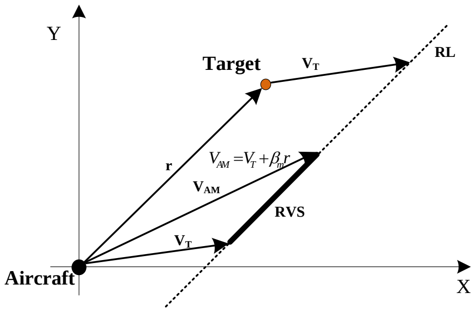


Figure 7 Construction of RVS

This method is applied to the quadrotor. The quadrotor has axial symmetry and the coupling in the x-axis direction and the y-axis direction is weak. Thus, we can only consider the kinetic limitations in the uniaxial direction. Due to the acceleration limit of the aircraft, the amount of change of the velocity command cannot be too large; otherwise it is difficult to track the velocity command. The limitation can be represented as follow:

$$|V_C - V'_C| < \Delta V_{m1} \tag{10}$$

In the previous our analysis is horizontal planning in the absence of obstacles. If the dynamic obstacle avoidance is required, the obstacle velocity can be calculate by the location deviation in a certain period and then calculate the area of velocity-obstacle (VO). The VO area represents the aircraft velocity region that the collision will happen at some time in the future. Therefore, if we calculate the VO as one of the velocity command limitations, we can achieve a dynamic obstacle avoidance capability. Here, we will analyze how to build VO region. For simplicity, we assume that the aircraft and obstacles are circular. To facilitate the analysis, we simplify the aircraft model into a point and enlarge the obstacle by the radius of the aircraft model, which is reasonable simplification while analyzing the problem of collision of two objects. As shown in Fig. 8, draw two rays (b'_r, b'_f) treating the aircraft as starting point and tangent to the obstacle circle. If the obstacle is stationary, it is easy to know that the light-colored triangle region (VO') is collision velocity area. When the obstacle is not stationary, if the velocity of the aircraft relative to the obstacle falls into VO' area, collision will occur. Then, VO region (the dark triangle region in Fig. 8) can be obtained and is expressed as follow:

$$VO = \{\mathbf{V_A} \mid (\mathbf{V_A} - \mathbf{V_O}) \in VO'\} \tag{11}$$

Where $\mathbf{V_O}$ denotes the velocity of the obstacle and $\mathbf{V_A}$ denotes the velocity of the aircraft. From the method above, we can get VO. If we limit velocity command using the VO, we can achieve the interception and obstacle avoidance capabilities at the same time.

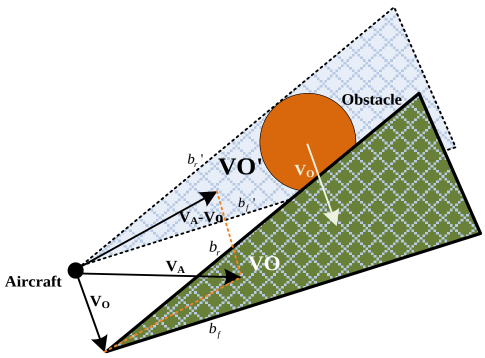


Figure 8 Construction of VO

Multi-object Detection and Tracking

1) Detection by HOG-Based SVM

Tracking a vary number of moving objects from an unsteady aerial platform has three major challenges. There are fast camera motion, changing scene and lighting, and targets entering and leaving the field of view at arbitrary position. To overcome these difficulties, a vision system is developed that is capable of learning, detecting and tracking multiple objects of interest. Our approach combines two algorithms: particle filter and SVM. The particle filter using color histogram as observation feature is a powerful technique for tacking multi-target. The color-based particle filtering process enable us to efficiently and reliably track the individual objects. However, the design of proposal distribution and the treatment of objects entering and leaving the scene are two crucial issues. In this paper, we incorporate information from HOG-based SVM detector. The offline learned SVM detector has two major functions in the tracking system. One is to initialize the starting states of particle filter automatically and add new objects entering the scene quickly. The other is to improve the proposal distribution for the particle filter.

Histograms of oriented gradients (HOG)[6] is a vector form feature descriptors which has been successfully used in object detection. We apply the HOG descriptor to encode the shape information of the targets. Since the HOG descriptor operates on localized cells, HOG descriptor is robust under viewpoint and lighting changes, and can be computed efficiently. HOG descriptor is shown in Fig. 9.

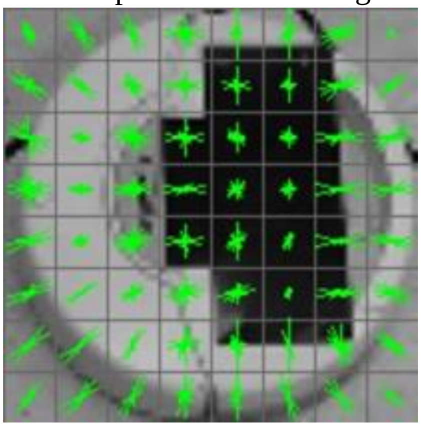


Figure 9. HOG descriptor.

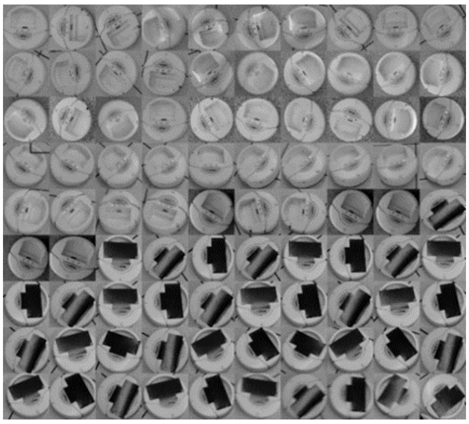


Figure 10. Training positive set of images for iRobot.

We adopt HOG-based linear SVM method for ground object detection. HOG+SVM is originally developed by Navneet Dalal and Bill Triggs[7], which demonstrates good performance in the field of pedestrian recognition. SVM is a statistical learning method based on the structure risk minimization principle. The objective of SVM is to find a best separating hyper plane with a maximum margin in the binary classification case.

In order to train the classifier, we choose a total of 200 figures of iRobot as positive samples, as shown in Fig. 10. Meanwhile, 1000 images are chosen to contain background as negative samples. These figures are scaled to have a resolution of 64×64 pixels.

One very important issue in the classifier training for one object class is how to select effective negative training samples. Since our tracker is usually implemented in a particular scene, there is no need to include training images outside the scene. To detect all the objects appearing at different scales in the test image, we build an image pyramid by sampling the test image at a factor of 1.1. At last, all the objects in the test image are scaled to the image window size (64×64) at some layer in the pyramid. Fig. 11 shows the result of our trained linear SVM detector. As we can see, HOG+SVM performs well at detection all the ground robots.

Figure 11. Results of iRobot detection.

2) Color-based Particle Filter for Tracking

Particle filter has been widely used in the specific context of visual tracking, in which the posterior density $p(\boldsymbol{x}_{0:t}|\boldsymbol{z}_{0:t})$ and the observation density $p(\boldsymbol{z}_{0:t}|\boldsymbol{x}_{0:t})$ are often non-linear and non-Gaussian. To achieve robustness against rotation and partial occlusion, we consider the color histogram (described in [8][8]) as target representation. As described in [9][9], different estimates of the "best" state at time t can be devised based on the discrete approximation of $p(\boldsymbol{x}_t|\boldsymbol{z}_{0:t})$. We can use the mean state of particles as tracker output at time t by

$$\widehat{\boldsymbol{x}}_t = \sum_{i=0}^{N} w_t^i \widetilde{\boldsymbol{x}}_t^i \quad (12)$$

If $\widehat{\boldsymbol{x}}_t$ is computed, we can get the target position relative to the camera from projection transformation.

3) SVM Embedded Particle Filter

There are two major problems that need to be solved in the standard particle filter. One is the initialization of the particle filter, the other is proper design of the importance proposal distribution. Followed by the Boosted Particle Filter (BPF), which is first introduced by Okumak[10], we properly solve the two problems by incorporating the result of HOG-based SVM detection algorithm.

We consider a video sequence of 500 frames to demonstrate the efficiency of our approach. Our particle filter using N=50 samples. Fig. 12 shows the result of our multi-target tracking system where the ground objects are coming in and out. It is important to note that the SVM embedded particle tracking system can successfully handle targets randomly coming in and out of the scene. In (a) of the figure, HOG-based SVM detector automatically detect the objects of interest. The detector also quickly detects a new object with a short time sequence in (b) figure. Then particles are generated to the new object and starting to track it in (c). In (d), one iRobot has disappeared from the scene to the down left corner of the image. The experiment demonstrates that our SVM embedded particle filter is an effective and fully automatic method which can be used in multi-object tracking system.

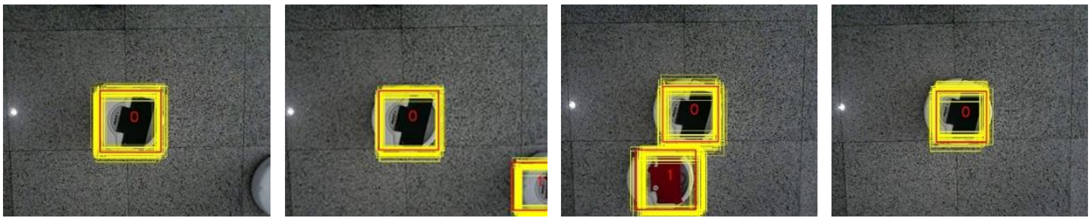


Figure 12. Ground targets appearing and disappearing from the scene.

2.2 Flight Termination System

There are three ways to achieve flight termination. The vehicle can be changed from autonomous to manual control by a switch on the RC transmitter, then the safety pilot take manual control of the vehicle. There is also a flight termination button in our ground station, if any emergency happens, someone can click the button to control the vehicle in a descent. The third way is that the judge can kill the device by the kill switch.

3. PAYLOAD

3.1 Sensor Suite

Our MAV is equipped with a laser rangefinder; one downward facing camera from IDS, an optical odometry sensor of PX4Flow and an airborne processor PX4. The laser rangefinder we select is HOKUYO UTM-30LX to getting the localization of obstacle. It is lightweight and provides a 270◦ field-of-view at 40Hz, up to an effective range of 30m. We deflect some of the laser beams downwards to estimate height above the ground plane.

3.2 Communications

A wireless ZigBee module is attached to the Atom Processor Board as well as the ground station, to provide the data link between the aerial vehicle and the ground station. Image from the aerial vehicle is transported to the ground station through a 5.8GHz wireless image transmission unit.

3.2 Power Management System

A set of 3S LiPo battery applies the power to the vehicle, it’s capacity is 6000mAh. The brushless motor controllers and the laser range finder are directly connected to the battery, other electrical equipments on the vehicle are powered by a voltage regulator module which gives 5V voltage. A loud tone will appear if the battery voltage drops under 9.8V. The interval is getting faster the lower the battery voltage gets. The battery voltage will be send to the ground station to be monitored.

4. OPERATIONS

4.1 Flight Preparations

In order to keep the vehicle safe and to complete the competition task, we developed a series of flight preparations.

Check List

1) Check the whole vehicle hardware, make sure the attachments are fixed firmly and the propellers are well.
2) Check the vehicle battery and plug into the vehicle, and check the RC transmitter battery.
3) Power on the ground station and open the control software.
4) Power on the vehicle.
5) Check the communication links between the ground stations and the vehicle, make sure it is well.
6) Check the sensors, and confirm they are working normally.
7) Turn the motors on, and then click on the take-off button.
8) Monitor the status of the vehicle, if there is any unusual circumstance, sent landing instructions immediately.
9) If any emergency happens, the safety pilot changes the vehicle from autonomous to manual control by a switch on the RC transmitter, and control it landing.

4.2 Man/Machine Interface

Ground monitoring station is the essential part of the system. It provides the monitoring of the vehicle state and enable us to modify the action of the vehicle.

Main Functions

1) Sending the taking-off and abrupt stop instructions

In the beginning of the competition we can send the taking-off instruction; once the aircraft lose control, we can send abrupt stop instruction to avoid some serious consequences through the man-machine interface.

2) Monitoring the flight state

Through the WIFI interface GCS can receive, analyze and show state information such as the attitude, speed, height, position etc., so that we can monitoring the state of the vehicle.

3) Displaying the image

We display the image on the screen of the GCS to facilitate the monitoring of the vehicle.

4) Localizing the aerial vehicle

We call the visual navigation process to achieve localizing the vehicle, and display the map we regenerate on the GCS to facilitate the monitoring of the vehicle.

5. RISK REDUCTION

5.1 Safety

In order to make sure the safety of aircraft and personnel around, we have taken some effective measures. First, we installed a protective frame around the vehicle to prevent the propeller to hit the ground or other objects. Every sensor is installed with appropriate damping measures to ensure the sensor is working normally. The control system of the vehicle will detect the battery voltage and the status of the data link in real time. If any emergency happen, the vehicle will perform a controlled descent to the ground. We have a safety operator who has the manual override capabilities to ensure safety. A safety switch is programmed onboard to ensure that the referee has the absolute right to shut down the whole system.

5.2 Obstacle avoidance

Obstacle avoidance is programmed in the mission planning section. Since the rules of IARC7 requires that the aerial vehicle should not touch any of the obstacle robot. Our vehicle would avoid getting close to the moving obstacle. The velocity of the obstacle would add a force to the vehicle and preventing it from collapse with the obstacle robot.

6. Testing

Gradual work has been done to test the whole system. The VICON camera based object tracking system is being used to precisely navigate to adjust the parameters of the controller. The visual navigation system is tested by a hand hold localizing test. After all the simulation and unit testing, flight test are done to verify the system's reliability.

7. CONCLUSION

Based on the research of quad-rotor platform, the aircraft was made to have the ability of autonomous navigation and control in the unknown environment to accomplish some specified tasks such as obstacle avoidance, online path programming and target identification. We expect our vehicle to be able to navigate in the competition arena and drive away more than 3 of the target ground robots.

8. REFERENCES

[1]Honegger, D., Meier, L., Tanskanen, P., and Pollefeys, M. "An open source and open hardware embedded metric optical flow cmos camera for indoor and outdoor applications," *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1736-1741.

[2]Beauchemin, S. S., and Barron, J. L. "The computation of optical flow," *ACM Computing Surveys (CSUR)* Vol. 27, No. 3, 1995, pp. 433-466.

[3]Davison, A. J., Reid, I. D., Molton, N. D., and Stasse, O. "MonoSLAM: Real-time single camera SLAM," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* Vol. 29, No. 6, 2007, pp. 1052-1067.

[4]Lin, C. C., and Tummala, R. L. "Mobile robot navigation using artificial landmarks," *Journal of Robotic Systems* Vol. 14, No. 2, 1997, pp. 93-106

[5]Canny, J. "A computational approach to edge detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, No. 6, 1986, pp. 679-698.

[6]Wren, C. R., Azarbayejani, A., Darrell, T., and Pentland, A. P. "Pfinder: Real-time tracking of the human body," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* Vol. 19, No. 7, 1997, pp. 780-785.

[7]Dalal, N., and Triggs, B. "Histograms of oriented gradients for human detection," *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 1, IEEE, 2005, pp. 886-893.

[8]Pérez, P., Hue, C., Vermaak, J., and Gangnet, M. "Color-based probabilistic tracking," *Computer vision—ECCV 2002*. Springer, 2002, pp. 661-675.

[9]Doucet, A., Godsill, S., and Andrieu, C. "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and computing* Vol. 10, No. 3, 2000, pp. 197-208.

[10]Okuma, K., Taleghani, A., De Freitas, N., Little, J. J., and Lowe, D. G. "A boosted particle filter: Multitarget detection and tracking," *Computer Vision-ECCV 2004*. Springer, 2004, pp. 28-39.