Real-Time Visual Tracking and Remote Control Application for an Aerial Surveillance Robot

Yeungkwon Woo, Seyoung Kim, and Eunjin Koh
Agency for Defense Development, Daejeon, Republic of Korea
Email: {wyk921, dontechman71, eikoda}@gmail.com

***Abstract*—This paper studied a real-time visual tracking and remote controlling application for an aerial surveillance robot. An automatic aerial surveillance has an intrinsic problem because it is hard to follow a designated target without its relative position information. In addition, the problem becomes worse in indoor situation, because the aerial robot cannot use the GPS information. However, the visual tracking technique can be a good alternative to keep the targets under the indoor environment or not. The visual tracking algorithm utilized by us uses correlation filters which can track the designated target in the complicated background. In our application, the aerial robot is quad-copter which can fly in 6 DOF (Degree Of Freedom). The aerial robot is navigated by method of the PNG (Proportional Navigation Guidance). The proportional navigation is generally used for targeting objects in aerial moving system. The image sequences captured from the aerial robot are transferred to a server and the server processes the visual tracking algorithm. Then the server sends the control command to the aerial robot in real-time. The aerial robot can route an aerial surveillance scenario. For that surveillance, we implemented a visual image processing and a control method using the AR-Drone 2.0 SDK. The application controls the quad-copter in workspace remotely through Wi-Fi. We generate ground-truths by manual surveillance. The result shows remarkable performance in real time, and the aerial robot achieves to fly for aerial surveillance scenario.**

***Index Terms*—real-time application, surveillance, aerial robot, visual tracking, remote control, proportional navigation**

I. Introduction

The research of surveillance application is growing with development of image sensing device and platform. In case of general application, the image sensing device means general cameras, which are CCD, IR, and SAR. Nowadays image sensing devices become smaller and can be mounted on small mobile robots. Hence, the mobile robot can be applied for surveillance application with using various image processing technology [1] [2]. Because of that, researchers become to be interested in SLAM [3] [4] (Simultaneous Localization And Mapping), guidance flight[5] [6], or aerial robot surveillance using image sense. For the example, in case of aerial robots which are the drone UAVs (Unmanned Aero Vehicle), the visual surveillance is regarded as the most important mission.

This paper proposes the real-time visual tracking and remote controlling application for aerial surveillance robot. In case of aerial surveillance, the problem is hard to follow the designated target because of the lack of the relative position information between the target and the robot. Furthermore, aerial robot is difficult for flying indoor without GPS information. Because of the problems, the image sensing device can make the aerial robot keep target under the surveillance using proposed application. The application satisfies to solve the problems in real-time, which mean online controlling with flying. In our application, a correlation filter based visual tracking algorithm is processed on server side. Simultaneously, aerial robot takes the remote control command as client side through Wi-Fi access. Using both sides of application, the aerial robot achieves the mission for surveillance scenario. Because of aerial moving scenario, PNG (Proportional Navigation Guidance) is used for the mission. The application is implemented by using C/C++ source code based AR-Drone 2.0 SDK. The experimental result shows performance in real time.

The paper is organized as follow. Section II shows the composition of the proposed real-time application system. Section III mainly describes the visual tracking algorithm. Section IV presents PNG control and our aerial surveillance scenario. Section V explains experimental results with tracking image and ground truth. In the last section VI, we remark conclusions about our application.

II. Real-Time System for Research

A. Aerial Suvelliance Robot

The aerial robot utilized in this research is a quad-copter, which is the Parrot's AR-Drone. This aerial robot has the technical specification related with our research as follows:

TABLE I. Aerial Robot Specification

Front camera	Field of view: 90 degree wide-angle diagonal lens
	CMOS color sensor
	Video frequency: 30 fps
	Resolution: 1280 x 720
Others	Wi-Fi /g/n
	6 DOF (pitch, yaw, and roll measurements)

Manuscript received July 1, 2014; revised December 15, 2014.

doi: 10.12720/joace.3.6.487-491

In this front camera speciation, 90 degree wide-angle view is helpful for the surveillance because the aerial robot is controlled as the strap-down type without the gimbals system. Through the 90 degree wide-angle, the aerial robot can get the 30 fps (frame per second) video stream form CMOS sensor. Thus the aerial robot can be controlled in 30 Hz. The control command is updated along with the result of the proposed tracking algorithm. In spite of the color image which has 3 channels RGB, our proposed tracking algorithm uses gray scale image (Fig. 1). Because that this tracking is based correlation filters which are related with frequency domain.

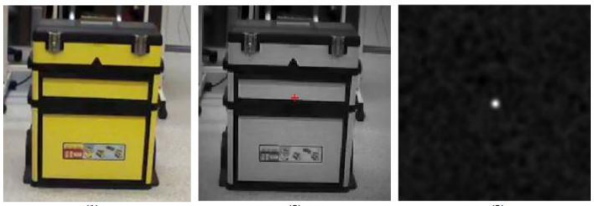

Figure 1. (1) The original color image, (2) Input image, (3) Correlation output of MOSSE filtering

As other specification, the aerial robot can connect to a sever side in Wi-Fi. Using Wi-Fi, the proposed system is composed of the client side and the server side. The client side is aerial robot which can transmit images from camera to server side, and takes control command from the server side. On the other hand, the sever side is a laptop computer which gets images from client side and sends control command. The quad-copter can fly in 6 DOF along control command. The aerial robot uses commands for Forward-Back, Left-Right, Up-Down, and Yaw (Turn).

Figure 2. The Parrot's AR-Drone.

B. *The Application for Visual Tracking and Contorl*

The application is implemented by using C/C++ source code based on AR-Drone SDK 2.0. The application deals with images for the tracking algorithm in real time. For that performance, the tracking algorithm is entirely included in the SDK code. For the tracking algorithm, various image processing methods are used for calculating image feature. For example, converting color images to gray images, cropping of images, and calculation for Fourier Trans Form are there.

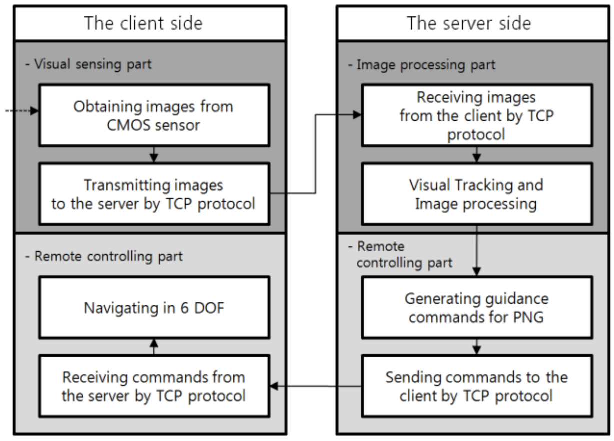

Figure 3. Application system composition.

For aerial robot control, the application calculates angle error between of tracking point and line of sight. The angular error outputs command with a tuned gain parameter. Then the commands which are acceleration in 6 DOF reduce errors. As reducing errors, aerial robot follows the target in field of view, and keeps it under surveillance.

III. VISUAL TRACKING ALGORITHM

A. *Correlation Filter by Using the FFT*

The correlation filter can track complex images through rotation, occlusion, and other distraction [7] [8]. The complex images are as same as those obtained from the aerial robot. This is the reason why application uses correlation filter based tracking algorithm for aerial surveillance. For the tracking, correlation filter is modeled initially in the first frame image. To initialize of correlation filter, we select the target as setting up a ROI (Region of Interest).

The selected the ROI computes a correlation in the Frequency domain. For transforming spatial image of the ROI to frequency domain, the tracking algorithm uses the FFT (Fast Fourier Transform). The FFT is formulated by F as the input image and H as a filter. The convolution of F and H* (*: the complex conjugate) is denoted [7]:

$$G = F \odot H^* \tag{1}$$

The convolution output indicates the tracking point and updates correlation filter. The tracking point is related with the maximum value of correlation outputs. After the IFFT (Inverse FFT) processing, the maximum value corresponds to the tracking point in the spatial image. If the variation of tracking point is above a certain amount, the correlation filter becomes to update on basis of the new tracking point.

B. *MOSSE Filter for Tracking*

The tracking algorithm is designed by new type of correlation filter which is a MOSSE (Minimum Output Sum of Squared Error) filter. The MOSSE filter can have robustness about variations of illumination, pose, and occlusion. The application requires the robust filter in a single frame because shorter training time is better to follow a target. The robust filter means to find the optimal filter that is based on the correlation output. The MOSSE filter is derived from optimization problem. On the assumption that every i_{th} frame customizes the output G (1) for the target tracking, the MOSSE filter is expressed as [7]:

$$H^* = \frac{\sum_i G_i \odot F_i^*}{\sum_i F_i \odot F_i^*} \quad (2)$$

The MOSSE filter is applied to track the target by online updating. As tracking the target through a number of frames, the target's appearance is changed by aspect change, scale, roll, and any disturbance. The tracking algorithm is able to deal with this change by updating filter. The MOSSE filter in each i_{th} frame is updated as follow as [7]:

$$H_i^* = \frac{A_i}{B_i} \quad (3)$$

$$A_i = \alpha G_i \odot F_i^* + (1-\alpha) A_{i-1} \quad (4)$$

$$B_i = \alpha F_i \odot F_i^* + (1-\alpha) B_{i-1} \quad (5)$$

The updating algorithm is usually concern about the update rate (or learning rate). In this (4), (5), the α is the updating rate. If the α is more than 0.5, the up-to-date frame is more affected than before. In case of us, we get the α = 0.1 as experimental parameter.

IV. Remote Control via the PNG

A. Remote Control Command Functions

The control command functions are based on the functions of AR-Drone SDK 2.0. The SDK has an acceleration function which along the x, y, z translational axis and yaw-axis. (The Fig. 4 shows axis on the basis of aerial robot). The function controls the aerial robot by the inputs about each direction. The input type of function is the velocity value of m/s. For example, the aerial robot can move to forward and down direction as input of V_x and V_z.

$$\text{Accelrtation F}(V_x, V_y, V_z, V_{yaw}) \quad (6)$$

$$\text{For normalization: } if\,(|V_i| > 1)\; V_i = \frac{V_i}{|V_i|} \quad (7)$$

$$(i \in \{x, y, z, yaw\})$$

The each input is estimated by the production of coefficients. The coefficients are experimentally determined for stabilization, which has ranged between 0.0 and 1.0. For normalization, if the absolute value of velocity is bigger than 1 m/s, then it becomes the maximal value as +1 or -1.

The function composes TCP packets as following syntax: "AT*PCMD=MODE, No rmalization of (V_x,V_y,V_z,V_{yaw}) ". This TCP packet becomes commands to control the aerial robot. Through the Wi-Fi, the commands are sent to client side by 30Hz as rate of videos streaming. The 30Hz rate is stable for flying at 0.5 m/s of V_x and 0.2 m/s of V_z in our application.

B. Proportional Navigation Guidance with Mean Filter

The PNG (Proportional Navigation Guidance) is one of guidance laws, which is generally used in homing problem of air target missiles [9]. The problem is based on the collision course of two moving object's velocity vectors. The velocity vector is relative to line of sight rate (LOS-rate). As follow Fig. 4, PNG is presented below:

$a_n = N\, i\, V$,

N: Proportionality constant,
i: LOS-rate, V: Closing velocity

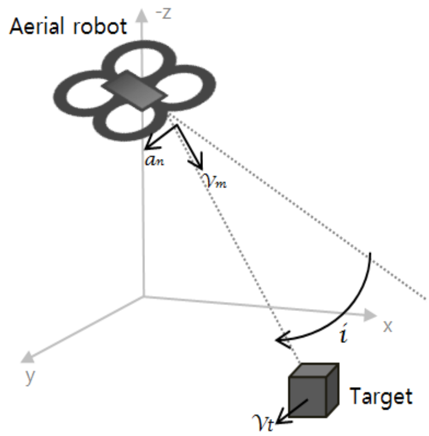

Figure 4. The coordinates of PNG.

In case of our application, the velocity of target Vt is a uniform motion as 0 or constant. The velocity of aerial robot Vm is calculated with $a_n$which has N = 2~4. In the a_n, the i V is calculated by a tracking error. The tracking error means the LOS-rate of target which is tracked from previous frames to current frame image. For stable tracking, we use a mean filter of the LOS-rates about 40 frames.

V. Experiment Result

A. Scenario of Surveillance

The scenario of our experiment is an aerial surveillance flying. First, we manually control the aerial robot and this robot watches the target objects. The aerial robot can fly in indoor and outdoor and the target objects are a general box, wastebaskets, and the pedestrian. In our work, the application can track the targets and make the aerial robot fly automatically by the tracking algorithm and the PNG controlling. As the flying scenario, the aerial robot can record the surveillance images. Thus we can compare both of the manual controlling and the application controlling. The experimental result shows that the application controlling can be better to keep the targets under the surveillance.

Figure 1. The aerial surveillance for the yellow box

B. Experimental Result

The application is worked for our challengeable experiment. This experiment is worked in narrow workspace, so our aerial robot need fly exactly. As the result indicates, the tracking algorithm can precisely track the target object. Even experimental situation has many bottlenecks, the MOSSE correlation filter works robustly as our parameter. Fig. 5 shows stream images of aerial robot, and the target (the yellow box) can be kept under surveillance in whole frame images.

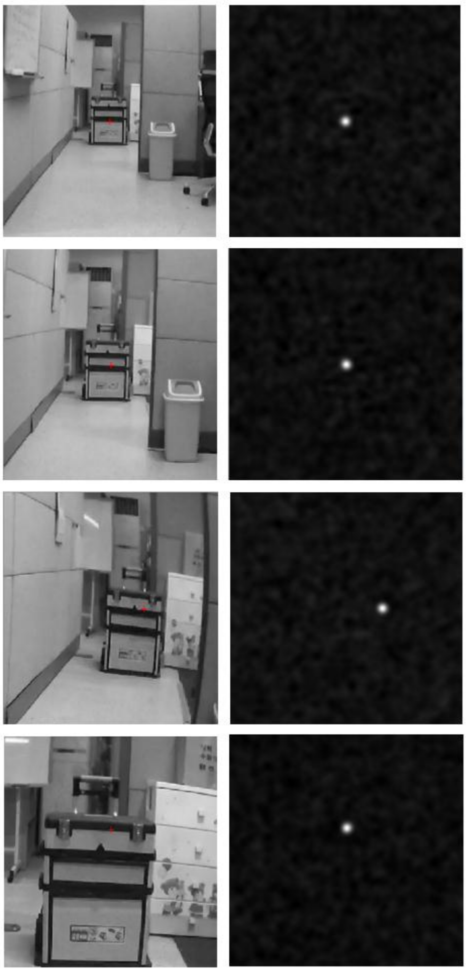

Figure 5. The visual tracking result: Tracking point(Red cross) and Correlation outputs(White circle)

In our work, we use 256 x 256 ROI and update late 0.1, PNG N = 4 to test the Fig. 6. The result in Fig. 6 shows tracking points corresponding to input images. Those images are transmitted from aerial robot. The server side receives images and inputs them to the tracking algorithm. The tracking algorithm convolutes the input image as modeling filter initially. After calculating correlation outputs, the tracking algorithm obtain new track point. The application updates filter on basis of new point and control command by the ROS rate as along PNG gain. Then, the application sends TCP packet with control command to aerial robot. The aerial robot receives the packet and controls to direction for decreasing the ROS rate.

C. Time Performance

The system of our application is composed as quad core processor (Intel i7 CPU @ 2.7GHz), Wi-Fi /g/n 100 M bps. In real time test, the application deals with 256 x 256 ROI image for tracking algorithm. Fig. 7 shows the processed time in the tracking algorithm as along the ROI sizes.

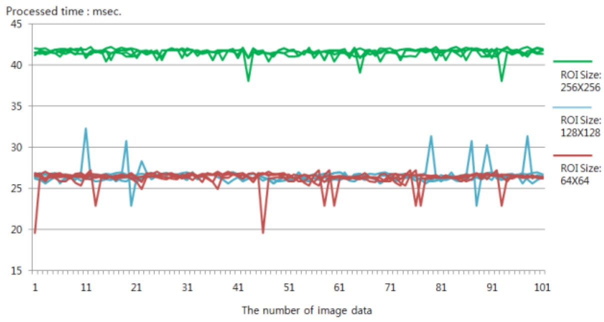

Figure 6. The processed time as along ROI size

As 256 x 256 ROI image, the application is processed under 45 msec. for tracking algorithm. This tracking contains the FFT/IFFT calculation and updating. The aerial robot can be controlled by 24 times update command on 1 second. This update rate can accurately control the aerial robot. We achieve to control the aerial robot in real time.

TABLE II. FPS (FRAME PER SECOND) FOR ROI SIZE

256 x 256	24 fps Avg.
128 x 128	37 fps Avg.
64 x 64	38 fps Avg.

We test other sizes which are 128x128, 64x64. In this case, our application performed more rapidly and the FPS is more efficient (The FPS Average is shown in TABLE II). Nevertheless more efficient, much small ROI is difficult to track a target because the image is lack of feature for correlation.

D. Robustness of Visual Tracking

The visual tracking algorithm can track various target objects. In our experiment, the targets are on the ground for aerial surveillance. For the tracking test, we use objects from small to large things. We try to track for aspect change, rotation and occlusion situation. In Fig. 8 is shown the result about the robustness of visual tracking.

Figure 7. The visual tracking result

VI. CONCULSION

The proposed application presents a performance for visual tracking and remote control in the real-time. Our approach shows that MOSSE correlation filter is fast for image processing by the FFT and IFFT. As the image processing in server side, the aerial robot can be controlled by receiving commands. Thus using visual tracking, aerial robot can fly and keep the target under the surveillance and the PNG can be used for flying. The experimental result show the application achieves the surveillance scenario on AR-drone robot. In the real time, visual tracking algorithm is worked robustly for aerial flying.

REFERENCES

[1] Y. Watanabe, A. J. Calise, and E. N. Johnson, "Vision-based obstacle avoidance for UAVs," in *Proc. AIAA Guidance, Navigation and Control Conference and Exhibit*, Hilton Head, 2007, pp. 20-23.

[2] S. H. Tang, G. Hamami, B. Karasfi, and D. Nakhaeinia, "Unsupervised and online place recognition for mobile robot based on local features description," *Journal of Automation and Control Engineering*, vol. 1, pp 60-64, March 2013.

[3] S. Chaves, S. Cohen, P. O'Keefe, and P. Ozog, "Real-time 6-DOF slam for a quadrotor helicopter," University of Michigan.

[4] A. J. Davison, I. D. Reid, and N. D. Molton, "MonoSLAM: Real-time single camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007, pp 1052-1067.

[5] K. Jonathan, S. Saripalli, and G. S. Sukhatme, "Combined visual and inertial navigation for an unmanned aerial vehicle," *Field and Service Robotics*. Springer Berlin Heidelberg, 2008, pp. 255-264.

[6] S. Srikanth, F. J. Montgomer, and S. Gaurav, "Vision-based autonomous landing of an unmanned aerial vehicle," in *Proc. IEEE International Conference on Robotics and Automation*, 2002, pp. 2799-2804.

[7] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, 2010, pp. 2544-2550.

[8] D. S. Bolme, Y. M. Lui, B. A. Draper, and J. R. Beveridge, "Simple real-time human detection using a single correlation filter," *Winter-PETS*, December 2009.

[9] J. Waldmann, "Line-of-sight rate estimation and linearizing control of an imaging seeker in a tactical missile guided by proportional navigation," *IEEE Transactions on Control Systems Technology*, vol. 10, pp 556-567, July 2002.

[10] S. Kim and J. Lee, "EKF-based localization of a mobile robot using pattern matching," *Journal of Automation and Control Engineering*, vol. 1, pp. 40-44, March 2013.

[11] J. Kim and J. Lee, "Predicting maximum traction to improve maneuverability for autonomous mobile robots on rough terrain," *Journal of Automation and Control Engineering*, vol. 1, pp. 1-6, March 2013.

[12] S. C. Han and H. Bang, "Proportional navigation-based optimal collision avoidance for UAVs," in *Proc. 2nd International Conference on Autonomous Robots and Agents*, Palmerston North, New Zealand, December 2004, pp. 76-81.

[13] L. Hyon, *et al*. "Real-time image-based 6-dof localization in large-scale environments," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[14] T. Yamasaki, H. Sakaida, and K. Enomoto, "Robust trajectory-tracking method for UAV guidance using proportional navigation," in *Proc. International IEEE Conference on Control, Automation and Systems*, 2007, pp. 1404-1409.

Yeungkwon Woo received his BS degree in Electronic Engineering and Computer Science from Kyungpook National University, Daegu, South Korea, in 2009 and MS degree in Computer Science from GIST, Gwangju, South Korea, in 2011. Since 2011, he has been a member of Agency for Defense Development, Daejeon, South Korea. His main research interests include: pattern recognition, machine learning, computer vision, and image processing.

Seyoung Kim received his BS and MS in Computer Engineering from Kyungpook National University, Daegu, South Korea, in 1994 and 1996. Since 1996, he has been a senior member of Agency for Defense Development, Daejeon, South Korea. His main research interests include: signal processing, FPGA programming, Image processing, and object detection.

Eunjin Koh received his BS and MS in computer science from Inha University, South Korea, in 2005 and 2007, respectively and received his PhD from University of Science and Technology, Daejeon, South Korea, in 2011. Since 2011, he has been a member of Agency for Defense Development, Daejeon, South Korea. His main research interests include: image processing, human-computer interaction, object tracking, and pattern classification