2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

An Image Processing Algorithm for Detection and Tracking of Aerial Vehicles

Sungwook Cho, Sungsik Huh, Hyong Sik Choi and David Hyunchul Shim, *Member*, *IEEE*

***Abstract*— This paper proposes an image processing algorithm for detection and tracking of aerial vehicles in sight. The proposed algorithm detects moving objects using the image homography calculated from a video stream taken from the onboard camera and determines whether the detected objects are approaching aerial vehicles by the Probabilistic Multi- Hypothesis Tracking (PMHT) method. This algorithm performs well especially when it is needed to detect any approaching aircraft seen with cluttered background. Further, our algorithm is suitable for real flight application as it is less sensitive to light conditions or color variations. The performance of the proposed algorithm is validated by applying it to the onboard video clips taken during actual flights using two unmanned aerial vehicles.**

I. Introduction

Federal Aviation Agency (FAA) mandates that all pilots flying in the civilian airspace should maintain constant vigilance for any incoming airplanes along their flight path[1]. Accordingly, in order for unmanned aerial vehicles (UAVs) to access the civilian airspace, they must be equipped with some type of sensing and avoidance capabilities. Often dubbed as "see-and-avoid", such technologies should allow UAVs to detect any incoming aircraft and execute an evasive maneuver following a collision-free trajectory [2][3].

There are a number of existing methods for UAVs to detect approaching aircraft. Radars and transponders are already widely adopted as standard collision avoidance methods. However, radars are expensive and heavy, rendering them not suitable for small-size UAVs. Transponder-based approaches such as TCAS only work with other cooperating vehicles equipped with compatible transponders. Therefore, for small UAVs, it has been proposed to use vision sensors such as cameras for detecting any incoming aircraft. Cameras can be inexpensive, light, and passive unlike radars or transponders. In addition, many UAVs are already equipped with cameras as their standard equipment for airborne surveillance.

So far, most vision-based detection and tracking algorithms have been developed for the cases that other aircraft is seen with clear sky or with scattered clouds in the background [4][5]. However, small UAVs oftentimes operate at lower altitudes due to the nature of given missions or limitations in service ceilings or communication range. In such cases, detecting approaching aerial vehicles becomes much more difficult due to the background cluttered with many natural or artificial features. Therefore, it is important to develop an effective algorithm to detect incoming aircraft seen with complex background. Such algorithm should be also computationally efficient since the relative speed of two approaching aircraft can be very high and there is not much time to make an evasive maneuver after an impending collision is detected.

Motivated by such needs, this paper suggests an image processing algorithm that allows for UAVs to detect and track approaching aircraft in real time using a vision sensor mounted on a UAV. This algorithm computes image homography from an image sequence to detect any moving objects in the view. Then a Probabilistic Multi-Hypothesis Tracking (PMHT) method is applied to track and manage the detected objects. Finally it is again used to determine whether an aerial vehicle is in a potential collision course or not. In the following sections, we will present the proposed algorithm as well as the results obtained by applying to the actual video clips of incoming aircraft seen from another UAV.

II. Detection of Moving Objects

A. Feature Detection, Tracking and Management

When a UAV "sees" an approaching aerial vehicle with its camera, the difference between the two images taken at different times (i.e. position) can be used to detect the aerial vehicle. Since the images are seen from two different positions, one image has to be "warped" as if it was taken from the same viewpoint of the other image. In order to find the transformation from one image to another, one has to find and track the matching feature points in the images.

Manuscript received (March 22, 2011). This work was supported by the grant No. 2009-09-sunggwa-7 from the Korea Aerospace Research Institute (KARI) and Brain Korea 21 Project, Korea Advanced Institute of Science and Technology (KAIST).

S. W. Cho is with the Department of Aerospace Engineering, KAIST, Daejeon, South Korea (phone: +82-42-350-3764; fax: +82-42-350-3710; e-mail: swcho84@kaist.ac.kr).

S. Huh, D. H. Shim are with the Department of Aerospace Engineering, KAIST, Daejeon, South Korea (e-mail: {hs2, hcshim}@kaist.ac.kr)

H. S. Choi is with the Korea Aerospace Research Institute (KARI), Daejeon, South Korea (e-mail: chs@kari.re.kr).

The first stage of moving object detection is to find the initial features in the first frame of the input image sequence. In this paper, the eigenvalue is calculated using a Shi-Tomasi-Kanade tracker[6][7] to measure the gray intensity change in the image. The gray intensity gradient is defined in the gray scale image as (1). The eigenvalues of x- and y-axes gradient matrix in (1) are obtained by solving (2).

$$\mathbf{C}=\begin{bmatrix}\sum\left(\frac{\partial I}{\partial x}\right)^2 & \sum\left(\frac{\partial I}{\partial x\partial y}\right)\\ \sum\left(\frac{\partial I}{\partial y\partial x}\right) & \sum\left(\frac{\partial I}{\partial y}\right)^2\end{bmatrix}=\begin{bmatrix}g_x^2 & g_{xy}\\ g_{yx} & g_y^2\end{bmatrix} \tag{1}$$

$$\mathbf{C}\cdot\mathbf{d}=\mathbf{e} \tag{2}$$

where $\mathbf{d}=\begin{bmatrix}d_x\\ d_y\end{bmatrix}$, $\mathbf{e}=\int\int_W\left[I(\mathbf{x},t)-J(\mathbf{x},t)\right]\begin{bmatrix}g_x\\ g_y\end{bmatrix}w(\mathbf{x})dxdy$, and $\mathbf{x}=\begin{bmatrix}x & y\end{bmatrix}^T$. In (2), $\mathbf{e}$ is the gray intensity change and $I(\mathbf{x})$ and $J(\mathbf{x})$ are images from the image sequence. W is a given feature window centered on the point $\mathbf{x}$ and $w(\mathbf{x})$ is a weighting function for computing gray intensity change. Note that $\mathbf{d}$ is the unknown displacement of a feature. If $\mathbf{d}_k=\mathbf{C}^{-1}\mathbf{e}$ is the displacement estimate at iteration k, and assuming a unit time interval between frames, the algorithm for minimizing (2) is

$$\begin{cases}\mathbf{d}_0=0\\ \mathbf{d}_{k+1}=\mathbf{d}_k+\mathbf{C}^{-1}\sum_W\left[\left(I(\mathbf{x},t)-I(\mathbf{x}+\mathbf{d}_k,t+1)\right)\nabla I(\mathbf{x},t)\right]\end{cases} \tag{3}$$

A feature is considered feasible if the eigenvalue satisfies certain conditions[7]. A Lucas-Kanade optical tracker[8] is used to match and track features, which will move around both in and out of the view area as the viewpoint changes during the flight. This method tracks features not only in the original image, but also in the images with their resolution decreased by exponents of 2 through an image pyramid method[8].

Additionally, in this paper, a procedure as shown in Fig. 1 is applied to manage the movement of features extracted using the algorithm explained above. Features that move beyond the limit of a prescribed bounded area around the edge of the image are discarded. The bounded area is divided into four parts, and in the case where there are no features in one of these areas, new features are added and their number is counted. This management procedure helps to maintain the adequate number of features for homography estimation.

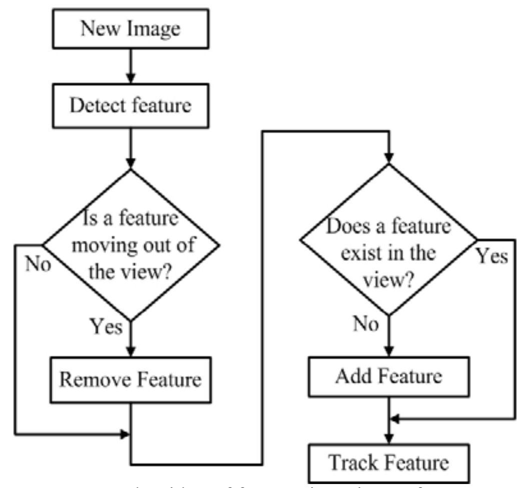


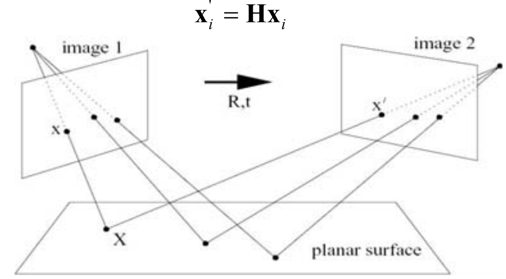
Fig. 1. Management algorithm of features in an image frame

B. 2D Homography Estimation

When a camera moves in a three dimensional space, a sequence of images with objects seen from different viewpoints is obtained as illustrated in Fig. 2.

Then there exists a projective transformation between two frames, and this is called image homography $\mathbf{H}$, which relates the same feature point $\mathbf{x}_i$ of one image to $\mathbf{x}_i^{'}$ in another image:

$$\mathbf{x}_i^{'}=\mathbf{H}\mathbf{x}_i \tag{4}$$



In (4), the two vectors $\mathbf{x}^{'}$ and $\mathbf{Hx}$ have the same direction, but they may differ in magnitude by a non-zero scale factor. Thus, the general form of the homography estimation equation is defined as (5).

$$\mathbf{x}_i^{'}\times\mathbf{H}\mathbf{x}_i=\mathbf{0} \tag{5}$$

In order to find the image homography in (5), a direct linear transformation (DLT) algorithm is used [9].

C. Detection of Moving Objects in a Successive Image Sequence

When carrying out image warping to fit a previous frame into the current frame, the image homography matrix gained from Section II.B. is used. Using (6) at a given time t, the difference image between the current and warped images can be obtained.

$$\Delta I_{sub} = I_{k-1}(x,y) - I_k(x,y) \tag{6}$$

To extract moving object in the difference image, the image is binarized by using a threshold value. By performing the morphological filtering such as erosion and dilation, the shape of the object is accentuated while the noise is eliminated.

In general, the threshold value is very sensitive to external influences such as lighting conditions, which may cause difficulties in finding and adjusting for a threshold value from actual outdoor tests. However, the moving object detection algorithm based on homography estimation described in this paper will decrease the effects of complex backgrounds during flight tests at a low altitude thus effectively reducing the sensitivity of the threshold.

III. Tracking of Aerial Vehicles

A. Multiple-Target Information Management Algorithm

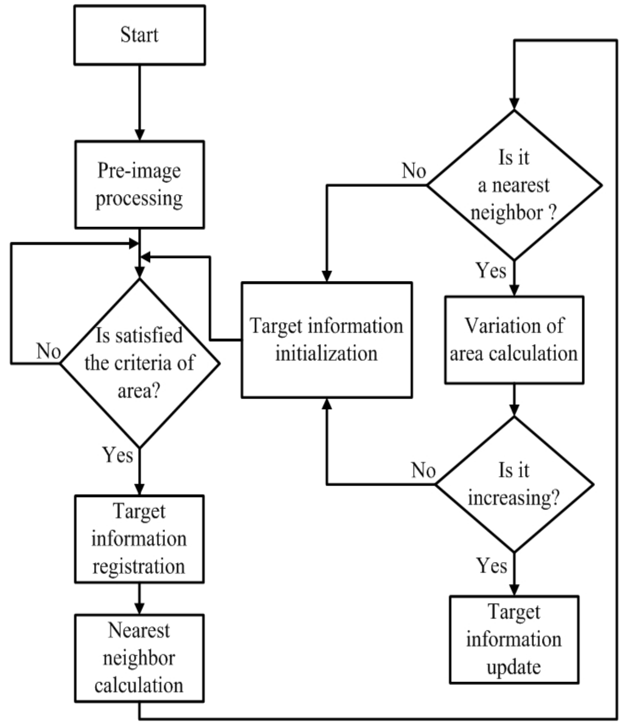
When a moving object is detected in an image, there must be a way to track it over a successive image sequence. These algorithms are called Multiple-Target Information Management (MTIM) algorithms. In this paper, we exchangeably use the term "target" as "object".

In this paper, a MTIM algorithm is implemented by dividing it into two parts. The first stage is an information collection stage using the objects acquired through methods explained in Section II.C. These objects are then labeled and its motion history is recorded. Certain objects that are detected in both previous and current frames are given with the specifications Δd_i for the distance between objects and for ΔA_i change in area.

The second stage is the exception processing and classification. Objects (now expressed as labeled contours) are assumed to have three properties and those that do not meet these three assumptions are processed as exceptions and their tracks are reset. Objects that have satisfied these three assumptions have their tracks maintained and detected as aerial vehicles.

This paper applies an in-house developed track reset algorithm for its MTIM algorithm, which uses a Probabilistic Multi-Hypothesis Tracking method[10]. The three assumptions are as follows:

1. A labeled contour must be detected at time t_{k-1} and t_k consecutively.



2. The closest labeled contour is the labeled contour that detected at time t_{k-1}.
3. For a labeled contour to be approaching, the area must increase by a set amount.

In other words, calculating the nearest neighbor finding an object simultaneously that proceeds in a direction where it increases in area are used in successive image sequence so that the object is recognized as the identical one. If an object detected in any frame does not satisfy the assumed conditions, its saved information track is reset. On the other hand, if an object does satisfy the conditions, the track is kept so that the object, which appears in continuous frames, is recognized as the same object and maintained. The flowchart of this algorithm is shown in Fig. 3.

B. Multi-hypothesis generation and management

After classifying the objects estimated as aerial vehicles by MTIM algorithm, proper algorithm must be used to assign probability for quantitative analysis. If it exceeds a certain threshold, the object would be considered as an aerial vehicle.

To calculate this probability, a hypothesis that uses specifications of the objects explained in Section III.A is defined as (7) and substituted into (8), where the log-likelihood is calculated. Involved signals of (7) are obtained by target information of MTIM algorithm.

$$p_{con} = w_{\Delta a} p_{\Delta a} + w_p p_p \tag{7}$$

$$l_i = l_{i-1} + \frac{\sum_{j=1}^{n} \left\{ \log(p_{con_j}) + \log(p_{obj_j}) \right\}}{n} \tag{8}$$

In (7), p_{con} is the connection probability of the previous frame being linked with the current frame. Note that p_p is the probability depending on the distance between continuously tracked objects, and $p_{\Delta a}$ is the probability depending on the change in area. $w_{\Delta a}$ and w_p are both the weighting factors of the probabilities. In addition, in (8), p_{obj} is the given probability when an object is consecutively detected in an image sequence and l_i is likelihood of current frame. All probabilities are calculated by normalization function. Through each specific normalization function, various values such area, distance and frame consistency are converted into a number ranging in 0.0~1.0.

Finally, the calculated log-likelihood is converted into a probability through the sigmoid function. Then if the converted probability increases over a certain criterion, the object will be classified as an approaching aerial vehicle.

This algorithm classifies the object as an approaching aircraft which fulfills the following criteria: 1) an object moving in the area continuously increasing and 2) two objects that continuously remain nearest neighbors between the previous and current frames. An object that meets both conditions in successive image sequence is given a probability of one. If an object is initially detected using the assumed criteria in Section III.A, a probability of 0.5 is given.

This hypothesis generation and management algorithm, combined with the MTIM algorithm, renders managing objects in single images more effective.

C. *Design of Kalman Filter for Multiple-Target Tracking*

Objects that are determined to be the same object in a successive image sequence are managed by having their hypotheses renewed – either through resetting or keeping their tracks. Along with this management algorithm, this paper uses the Kalman filter to track multiple targets detected to be aerial vehicles and even objects in the image.

In this paper, the state variables using Kalman filter for multiple target tracking are shown in (9) for the tracking of a rectangle contour.

$$\mathbf{X}_k^N = \left[x_k^N, \dot{x}_k^N, y_k^N, \dot{y}_k^N, w_k^N, \dot{w}_k^N, h_k^N, \dot{h}_k^N \right] \tag{9}$$

Note that the rectangle contour`s width and height are expressed with w and h, while N stands for the number of objects in the image. Since there are multiple moving objects to be tracked, the algorithm is designed to run internal and external loops concurrently.

When tracking an object in an image, setting the initial conditions properly is vital since the stability of the Kalman filter is greatly affected by such initial conditions. Hence the conditions are set as in (10),

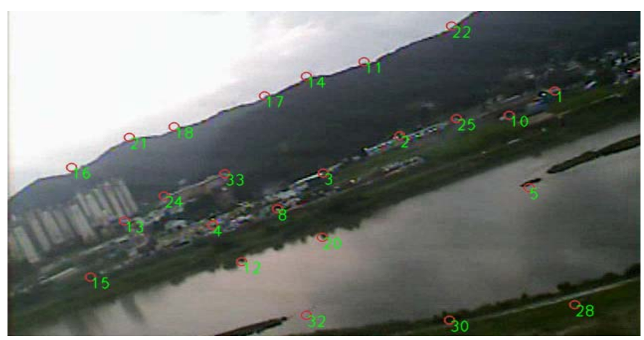
$$\dot{\eta}_k^N = \eta_k^N - \eta_{k-1}^N \tag{10}$$

which uses the relationship between the previous and current frame acquired by MTIM algorithm.

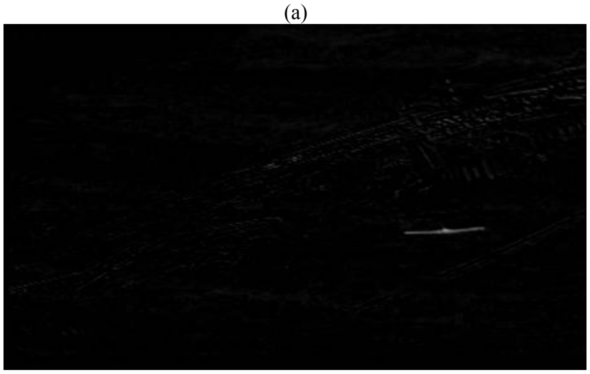
IV. Image Processing Result

The image processing algorithm designed in this paper was applied to an onboard video taken in a UAV. The onboard video was taken by two UAVs simulating a collision situation, flying along a circle of diameter 80m in opposite direction. The onboard video was taken from the UAV turning in the clockwise direction.[12]

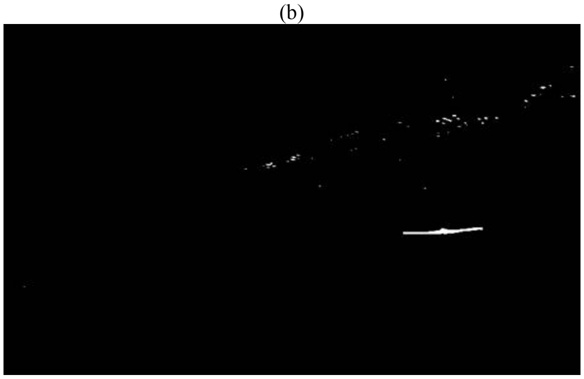
The first phase of the image processing algorithm for detecting and tracking of aerial vehicle in short range is to detect and track features from images of the ground. In Fig. 4. (a) and (b), one can see the management results of features described in Section II.A. The green circles represent newly added features while the red circles represent features exempt from calculation. Fig. 5. (a). is a result of moving



(a)



(b)



(c)

Fig. 5. (a) Source image. (b) The result of moving object being detected. (c) The result of binarization and morphological filtering.

objects being detected in the image sequence through calculating the difference image after having image warping using estimated homography.

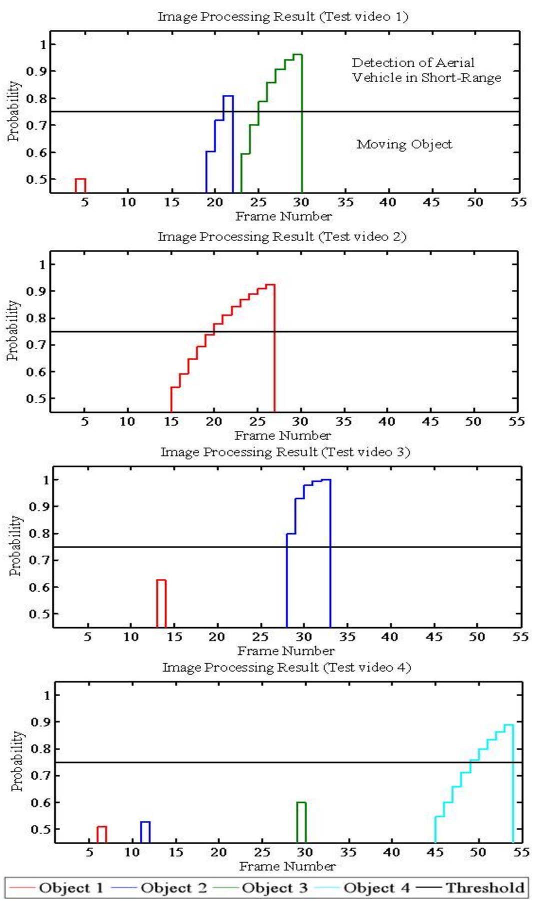
It is more difficult to detect a blended wingbody (BWB)-based UAV, which we used in our experiment[11], than a general fixed wing UAV. Nevertheless, the proposed algorithm can detect BWB-type aircrafts as shown here. The object detected by using the image difference as shown Fig. 5. (b). has a blurred silhouette in addition to the evenly distributed noise caused by errors from estimated homography. To remove the noise and accentuate the shape of object, a binarization by using a threshold value, and morphological filtering are performed consecutively. After going through such image processing procedures, the shape of the object is greatly enhanced as shown in Fig. 5. (c).

In Fig. 5. (c), a small amount of noise spots can be seen, but these can be effectively eliminated by assigning a size boundary when carrying out labeling during the MTIM algorithm. The MTIM algorithm, using the three conditions stated above, creates a data track for newly detected objects and either resets or maintains this data track. This algorithm allows for moving objects detected in successive image sequence to be maintained as the same object, as well as helping in determining whether an aerial vehicle is approaching or not. This can be confirmed through Fig. 6.

Fig. 6. The result of detecting an approaching aerial vehicle

Furthermore, the detection of aerial vehicles determined to be on approaching is represented as a warning signal displayed if they exceed certain threshold as seen in Fig. 8. The yellow rectangle in Fig. 6 signifies an object that existed in the previous frame. The red rectangle corresponds to the object in the current frame, the green rectangle the object being tracked through the Kalman filter. Expanding of the results in Fig. 6, the results of detecting and tracking an approaching aerial vehicle in successive image sequence can be confirmed in Fig. 7.

Four in-flight videos simulating a collision situation are applied to the proposed algorithms, whose results are given in Fig. 8. The vertical axis is the probability of the approaching aircraft and the horizontal axis is the number of frames that indicate the degree of consistency. By iteratively applying the entire algorithm, the threshold was assigned to a value higher than 0.75, concluding that the moving object is an approaching aircraft. Among the objects detected in Fig. 8, those that cannot be tracked continuously in image sequence have their tracks reset, while those that are determined to be the same object in image sequence have their tracks maintained. At the same time, it can be confirmed that when the track is kept, the probability of that object being an aerial vehicle increases. When this probability exceeds the preset threshold, we conclude that there is an object approaching to the host vehicle with a risk of impending collision.



V. CONCLUSION

This paper proposed an image processing algorithm for detecting and tracking aerial vehicles in short range and applied this algorithm to onboard videos from flight tests. This algorithm detects and tracks features by measuring the gray intensity change to compute image homography in the image sequence. Since the moving objects in the images are detected after warping the image by using image homography and calculating difference image, the sensitivity of the threshold value due to weather, lighting, and other external effects are attenuated. Furthermore, whether a detected moving object is an approaching aerial vehicle or not can be determined through a probabilistic multi-hypothesis tracking method and multi-hypothesis generation and management, aerial vehicles in short range are effectively detected even with backgrounds with cluttered ground objects. The proposed image processing algorithm will be applied for in-flight experiment for a validation in a real situation.

ACKNOWLEDGMENT

This work was supported by “KARI-University Partnership Program (grant No. 2009-09-sunggwa-7)” from the Korea Aerospace Research Institute (KARI) and Brain Korea 21 Project, Korea Advanced Institute of Science and Technology (KAIST).

REFERENCES

[1] *Airplane Flying Handbook*, U.S. Department of Transportation Federal Aviation Administration, FAA-H-8083-3A, 2004.

[2] A. D. Zeitlin, “Technology Milstones - Detect, Sense & Avoid for Unmanned Aircraft Systems”, *AIAA Infotech@Aerospace Conference and Exhibit,* AIAA-2007-2765, 2007

[3] Kie-Jeong Seong, Eung-Tai Kim, Seong-Pil Kim, “Development Trend of the Autonomous Flight Control Technology”, *Current Industrial and Technological Trends in Aerospace*, Vol.6.2, 2008, pp. 143~153

[4] Eric N. Johnson, Anthony J. Calise, Yoko Watanabe, Jincheol Ha, and James C. Neidhoefer, “Real-Time Vision-Based Relative Aircraft Navigation,” *Journal of Aerospace Computing, Information, and Communication*, Vol.4, 2007, pp.707-738.

[5] John Lai, Luis Mejias, and Jason J. Ford, “Airborne Vision-Based Collision-Detection System,” *Journal of Field Robotics*, Vol.28, Issue 2, 2011, pp.137-157.

[6] Jianbo Shi, and Carlo Tomasi, “Good Features to Track,” *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1994, pp.593~600.

[7] A. Fusiello, E. Trucco, T. Tommasini, V. Roberto, “Improving Feature Tracking with Robust Statistics”, *Pattern Analysis & Applications*, Vol.2, Number 4, 1999, pp. 312-320.

[8] Jean-Yves Bouguet, “Pyramidal Implement ation of the Lucas Kanade Feature Tracker Description of the algorithm” Intel Corporation, Microprocessor Research Labs, 2000, http://www.intel. com/ research/mrl/research/opencv/

[9] Richard Hartley, and Andrew Zisserman, *Multiple View Geometry in computer vision 2nd edition*, UK: Cambridge University Press, 2003.

[10] Roy L. Streit, and Tod E. Luginbuhl, "Probabilistic Multi-Hypothesis Tracking," *NUWC-NPT Technical Report* 10.428, 15, 1995

[11] S.Huh, and D.H.Shim, "A Vision-Based Landing System for Small Unmanned Aerial Vehicles using an Airbag," *Control Engineering Practice*, Vol. 18, Issue 7, 2010, pp. 812-823.

[12] Dong-Il You, and Hyun-Chul Shim, “Leader-Follower based Formation Guidance Law and Autonomous Formation Flight Test of Multiple MAVs,” *Domestic Journal of KSAS*, 2011, Vol.1.