2012 12th International Conference on Control, Automation and Systems
Oct. 17-21, 2012 in ICC, Jeju Island, Korea

Stereo Vision Based Collision Avoidance of Quadrotor UAV

Jongho Park[1] and Youdan Kim[2]

Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul, 151-742, Korea
[1] (Tel : +82-2-880-7392; E-mail: parkjo05@snu.ac.kr)
[2] (Tel : +82-2-880-7398; E-mail: ydkim@snu.ac.kr)

Abstract: In this paper, collision avoidance of a quadrotor unmanned aerial vehicle using stereo vision sensor is proposed. Quadrotor dynamic modeling is performed, and under-actuated problem of the quadrotor is dealt by introducing virtual inputs. Waypoint guidance and controller are designed using feedback linearization and linear quadratic tracker. Stereo vision is used to acquire depth map, and an obstacle is detected by the computed depth map. Then, collision cone approach is adopted to perform collision avoidance. Numerical simulation is performed to validate the performance of the collision avoidance algorithm.

Keywords: UAV, Quadrotor, Stereo vision, Depth map, Feedback linearization, Collision cone.

1. Introduction

Recently, Unmanned Aerial Vehicle (UAV) has been attracting a lot of attention in many fields. UAV has great advantage of autonomously performing missions such as surveillance, reconnaissance, exploration, delivery, etc. Most of these missions require close proximity with various natural and artificial structures. A collision with such structures or other UAVs can be fatal in accomplishing the mission. Therefore, it is highly required for the UAV to detect an obstacle and avoid collision [1]. Various kinds of sensors are used for obstacle detection and collision avoidance. Active sensors such as lidar give accurate information, but they are expensive and heavy, which is not suitable for small UAVs. On the other hand, vision based passive sensors such as camera are gaining popularity due to their low cost and light weight [2,3].

In this study, a quadrotor, one type of UAV systems, is modeled without any approximation or linearization. The whole system is divided into position and attitude control subsystems, and controllers are designed separately. Virtual inputs are used to deal with an under-actuated problem of the quadrotor [4]. Feedback linearization is used to transform the coordinate system, and linear quadratic tracker is applied to design a controller [5]. Stereo vision sensor system is used to detect an obstacle due to its advantage of capability of obtaining depth map over monocular camera [6]. When an obstacle is detected through the depth map, collision safety boundary and collision cone can be formed [7]. If the velocity vector of the quadrotor is contained in the collision cone, the quadrotor performs collision avoidance maneuver. Numerical simulation is performed to verify the performance of the proposed collision avoidance algorithm.

2. Quadrotor Dynamics Modeling

Quadrotor is configured in a way that front rotor and rear rotor rotate clockwise, and the others rotate counterclockwise as shown in Fig. 1. Let us define $\xi \triangleq \begin{bmatrix} x & y & z \end{bmatrix}^T$ and $\varphi \triangleq \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T$ as the position and attitude angle vectors of the quadrotor expressed in the inertial frame, respectively. And, $V \triangleq \begin{bmatrix} u & v & w \end{bmatrix}^T$ and $\Omega \triangleq \begin{bmatrix} p & q & r \end{bmatrix}^T$ are defined as the translation and rotation velocity vectors in the body fixed frame, respectively.

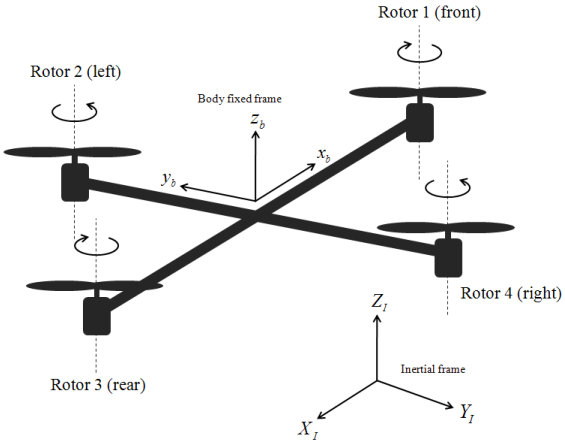

Figure 1. Quadrotor structure

Newton-Euler equations describe a rigid body dynamics including external forces and external torques expressed in the body fixed frame as follows

$$\begin{bmatrix} m\mathrm{I} & 0 \\ 0 & J \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\Omega} \end{bmatrix} + \begin{bmatrix} \Omega \times mV \\ \Omega \times (J\Omega) \end{bmatrix} = \begin{bmatrix} \sum F_{ext} \\ \sum T_{ext} \end{bmatrix} \quad (1)$$

where m and $J = diag\left[J_x, J_y, J_z \right]$ are the mass and the inertial matrix of the quadrotor, respectively, $\mathrm{I} \in \mathbb{R}^{3\times 3}$ is an identity matrix, and $\sum F_{ext}$ and $\sum T_{ext}$ include external forces and external torques, respectively.

The relationship between the body fixed frame and

978-89-93215-04-5 95560/12/$15 ⓒICROS

the inertial frame can be described as follows

$$\dot{\xi} = R_t^{B\to I} V \tag{2}$$

$$\Omega = R_r^{I\to B} \dot{\varphi} \tag{3}$$

where $R_t^{B\to I}$ is the transformation matrix from the body fixed frame to the inertial frame, and $R_r^{I\to B}$ is the rotation velocity matrix from the inertial frame to the body fixed frame. Using (1-2-3) rotation Euler angles, the transformation matrices can be represented as

$$R_t^{B\to I} = \begin{bmatrix} c\theta c\psi & -c\theta s\psi & s\theta \\ s\phi s\theta c\psi + c\phi s\psi & -s\phi s\theta s\psi + c\phi c\psi & -s\phi c\theta \\ -c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta s\psi + s\phi c\psi & c\phi c\theta \end{bmatrix} \tag{4}$$

$$R_r^{I\to B} = \begin{bmatrix} c\theta c\psi & s\psi & 0 \\ -c\theta s\psi & c\psi & 0 \\ s\theta & 0 & 1 \end{bmatrix} \tag{5}$$

where $s\phi$ and $c\phi$ denote $\sin\phi$ and $\cos\phi$, respectively. Differentiating Eq. (2) and Eq. (3) with respect to time and substituting the results into Eq. (1) yield

$$\ddot{\xi} = R_t^{B\to I} \frac{1}{m} \sum F_{ext} \tag{6}$$

$$\begin{aligned} \ddot{\varphi} = {} & \left(R_r^{I\to B}\right)^{-1} J^{-1} \left(\sum T_{ext} \right. \\ & \left. - \left(R_r^{I\to B}\dot{\varphi}\right) \times \left(J R_r^{I\to B}\dot{\varphi}\right) \right) \\ & - \left(R_r^{I\to B}\right)^{-1} \left(\frac{\partial R_r^{I\to B}}{\partial \theta}\dot{\theta} + \frac{\partial R_r^{I\to B}}{\partial \psi}\dot{\psi} \right) \dot{\varphi} \end{aligned} \tag{7}$$

In Eq. (6), the external forces $\sum F_{ext}$ include F_i produced by the rotor system of the quadrotor, an aerodynamic reaction force F_{aero}, and a gravitational reaction force F_{grav} as follows

$$\sum F_{ext} = F - F_{aero} - F_{grav} \tag{8}$$

where

$$F = \begin{bmatrix} 0 & 0 & \sum_{i=1}^{4} F_i \end{bmatrix}^T \tag{9}$$

$$F_{aero} = K_t V = K_t \left(R_t^{B\to I}\right)^{-1} \dot{\xi} \tag{10}$$

$$F_{grav} = m \left(R_t^{B\to I}\right)^{-1} G \tag{11}$$

In the above equations, F_i is a thrust force with $i \in \{1,2,3,4\}$, $K_t = diag\left[k_{tx}, k_{ty}, k_{tz}\right]$ is an aerodynamic friction matrix of the force, and G is a gravity acceleration vector. Using Eq. (6) and Eqs. (8)－(11), the translational dynamics of the quadrotor can be expressed in the inertial frame as

$$\begin{aligned} \ddot{\xi} = {} & -\frac{1}{m} R_t^{B\to I} K_t \left(R_t^{B\to I}\right)^{-1} \dot{\xi} \\ & - G + \frac{1}{m} R_t^{B\to I} F \end{aligned} \tag{12}$$

In Eq. (7), the external torques $\sum T_{ext}$ include T_i created by the rotor system of the quadrotor, and an aerodynamic reaction torque T_{aero} as follows

$$\sum T_{ext} = T - T_{aero} \tag{13}$$

where

$$T = \begin{bmatrix} d\left(F_2 - F_4\right) \\ d\left(F_3 - F_1\right) \\ \left(T_1 - T_2 + T_3 - T_4\right) \end{bmatrix} \tag{14}$$

$$T_{aero} = K_r \Omega = K_r R_r^{I\to B} \dot{\varphi} \tag{15}$$

In the above equations, T_i is a torque with $i \in \{1,2,3,4\}$, d is a distance from the center of mass of the quadrotor to the rotor axes, and $K_r = diag\left[k_{rx}, k_{ry}, k_{rz}\right]$ is an aerodynamic friction matrix of the torque. By substituting Eqs. (13) and (15) into Eq. (7), the rotational dynamics of the quadrotor can be represented in terms of the inertial frame as

$$\begin{aligned} \ddot{\varphi} = {} & -\left(R_r^{I\to B}\right)^{-1} J^{-1} K_r R_r^{I\to B} \dot{\varphi} \\ & - \left(R_r^{I\to B}\right)^{-1} J^{-1} \left(R_r^{I\to B}\dot{\varphi}\right) \times \left(J R_r^{I\to B}\dot{\varphi}\right) \\ & - \left(R_r^{I\to B}\right)^{-1} \left(\frac{\partial R_r^{I\to B}}{\partial \theta}\dot{\theta} + \frac{\partial R_r^{I\to B}}{\partial \psi}\dot{\psi} \right) \dot{\varphi} \\ & + \left(R_r^{I\to B}\right)^{-1} J^{-1} T \end{aligned} \tag{16}$$

3. Guidance and Control System Design

The quadrotor has six states $\begin{bmatrix} x & y & z & \phi & \theta & \psi \end{bmatrix}^T$ and four inputs (four rotors). Therefore, the quadrotor is an under-actuated system. It is not easy to design a controller for the under-actuated systems. To make the quadrotor a fully-actuated system, the whole system is divided into position and attitude control subsystems, and two outputs $\begin{bmatrix} \phi & \theta \end{bmatrix}^T$ of the attitude control subsystem are regarded as the virtual inputs of the position control subsystem as shown in Fig. 2.

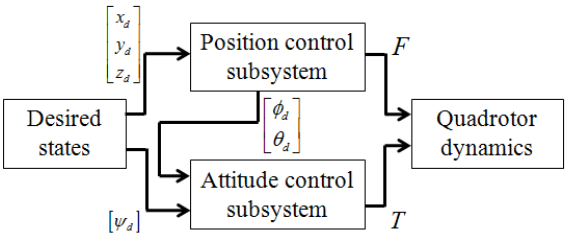

Figure 2. Control system

Feedback linearization technique is adopted to deal with the nonlinearity of the quadrotor dynamics. The method transforms an original nonlinear system into an equivalent linear system without any approximation.

To control the transformed linear system, a linear quadratic tracker is adopted. Let us consider the following linear system which is transformed by the feedback linearization.

$$\begin{aligned} \dot{\bar{x}} &= A\bar{x} + B\bar{u} \\ \bar{y} &= C\bar{x} + D\bar{u} \end{aligned} \tag{17}$$

where $\bar{x} \in \mathbb{R}^n$ is a state vector, $\bar{u} \in \mathbb{R}^m$ is an input vector, and $\bar{y} \in \mathbb{R}^l$ is an output vector. The purpose of the linear quadratic tracker is to make the output $\bar{y}$ track the desired output $\bar{y}_d$ as $t \to \infty$. The model dynamics of the desired state vector can be represented as

$$\begin{aligned} \dot{\bar{x}}_d &= A\bar{x}_d + B\bar{u}_d \equiv 0 \\ \bar{y}_d &= C\bar{x}_d + D\bar{u}_d \end{aligned} \tag{18}$$

In the position control subsystem, the desired output is the position vector of the waypoint. And, in the attitude control subsystem, the desired output is three angles where two of them are provided by the position control subsystem.

The cost function of the linear quadratic tracker is defined as

$$J = \frac{1}{2}\int_0^\infty \left[\tilde{x}^T Q\tilde{x} + \tilde{u}^T R\tilde{u}\right] dt \tag{19}$$

where $\tilde{x}$ and $\tilde{u}$ are state error and control input error vectors, respectively, and Q and R are weighting matrices of the state error and control input error vectors, respectively. The optimal solution minimizing Eq. (19) is obtained by solving the following matrix algebraic Riccati equation.

$$PA + A^T P - PBR^{-1}B^T P + Q = 0 \tag{20}$$

Now, the control input error and the real control input vectors can be obtained as follows

$$\tilde{u} = -R^{-1}B^T P\tilde{x} = -k\tilde{x} \tag{21}$$

$$\bar{u} = \left(\bar{u}_d + k\bar{x}_d\right) - k\bar{x} \tag{22}$$

where $\bar{x}_d$ and $\bar{u}_d$ can be obtained by

$$\begin{bmatrix} \bar{x}_d \\ \bar{u}_d \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \bar{y}_d \end{bmatrix} \tag{23}$$

4. Depth Map Modeling

In a monocular camera system, an object in 3-dimensional space is projected onto an image plane which is 2-dimensional space. As a result, the information of distance in the direction of projection is lost. On the other hand, a stereo vision system can compute a depth map which compensates the lost information.

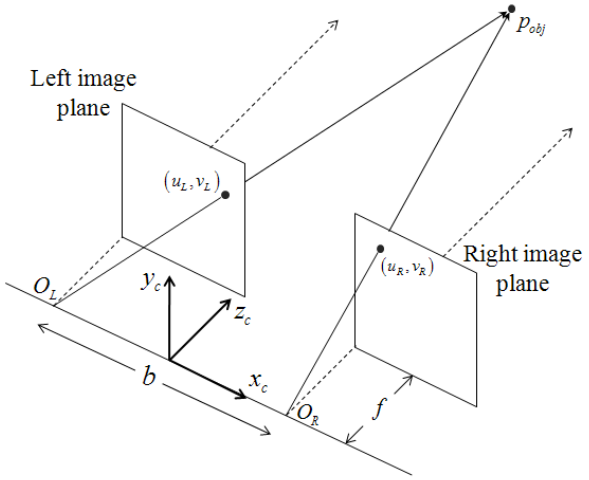

Figure 3. Structure of stereo vision system

Figure 3 shows the structure of the stereo vision system. In Fig. 3, b is a baseline, f is a focal length, and p_{obj} is the location of object. Also, (u_L, v_L) and (u_R, v_R) are points on the left and right image planes, respectively, which are obtained as follows

$$\begin{bmatrix} u_L \\ v_L \end{bmatrix} = \frac{f}{z}\begin{bmatrix} x - (-b/2) \\ y \end{bmatrix} = \frac{f}{z}\begin{bmatrix} x + b/2 \\ y \end{bmatrix} \tag{24}$$

$$\begin{bmatrix} u_R \\ v_R \end{bmatrix} = \frac{f}{z}\begin{bmatrix} x - b/2 \\ y \end{bmatrix} \tag{25}$$

Using Eq. (24) and Eq. (25), we have

$$u_L - u_R = \frac{f}{z}b \tag{26}$$

$$z = \frac{fb}{u_L - u_R} \tag{27}$$

where $u_L - u_R$ and z are called disparity and depth, respectively. Equation (27) refers that the depth can be obtained by using the focal length of the camera, the baseline of the camera, and the disparity of the object. Table 1 shows the depths according to the disparities where focal length is $0.01\,m$, baseline $0.2\,m$, horizontal field of view 48°, vertical field of view 36.9°, and pixel size $2.75\times10^{-5}\,m$.

Table 1. Relationship between disparity and depth

Disparity		1	2	3	4
Depth (m)		71.87	35.94	23.96	17.97
5	6	7	8	9	10
14.37	11.98	10.27	8.98	7.99	7.19
…	15	16	17	18	19
…	4.79	4.49	4.23	3.99	3.78
20	21	22	23	24	25
3.59	3.42	3.27	3.12	3.00	2.87
26	27	28	29	30	31
2.76	2.66	2.57	2.48	2.40	2.32

5. Collision Avoidance Using Stereo Vision

Figure 4 shows the concept of collision avoidance using the depth map of the stereo vision. When the quadrotor is moving towards the goal point, it constantly acquires the depth map. Note that the depth map and the actual region are not same because of the discreteness of the depth map. Disparity is composed of the multiplication of actual size of the pixel and pixel difference between the left image and the right image of the object. The latter is a positive integer, which means that the depth map does not give the entire information of true region.

Also, the blind spot, which is the opposite side of the obstacle, cannot be detected by the cameras mounted on the quadrotor. In this study, this region is also considered as obstacle as shown in Fig. 4.

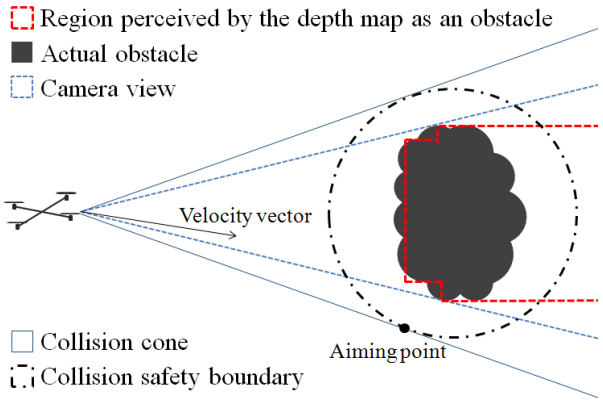

Figure 4. Generating aiming point to avoid obstacle

If the depth value of a pixel is lower than the predefined threshold, then the pixel is regarded as an obstacle. For example, if the threshold is set to $4\,m$, then the pixels whose disparity is higher than 18 (up to 31) are regarded as dangerous region as shown in Table 1.

Let us consider a situation that an obstacle is detected in the image plane during the flight. Then, the sphere called collision safety boundary is formed around the obstacle, and the collision cone is constructed. The collision cone is defined by a set of tangent lines from the quadrotor to the collision safety boundary as shown in Fig. 4. If the velocity vector of the quadrotor is contained within the collision cone, then it means that the obstacle will be an obvious threat to the quadrotor unless it changes its trajectory. Therefore, in this case, the obstacle is said to be critical, and the quadrotor immediately performs collision avoidance maneuver. The primary task of the collision avoidance maneuver is to guide the quadrotor to a new waypoint, instead of the goal point, to avoid the threat of the obstacle. The new waypoint is called aiming point as shown in Fig. 4. The point having the nearest distance to the velocity vector is selected as the aiming point as follows

$$aiming\ \ point = \arg\min_{x} \left\{ \left\| x - \dot{\xi} \right\|_2 \mid x \in C \right\} \qquad (28)$$

where C is a set of aiming point candidates. Figure 4 shows only two aiming point candidates. However, in 3-dimensional space, C contains all the intersection points of the collision safety boundary and the collision cone. The collision avoidance algorithm is summarized in Fig. 5.

Algorithm

(i) Obtain depth map of stereo vision.

(ii) *if* depth value of a pixel $<$ predefined threshold
Go to (iii).
else
Navigation maneuver: Waypoint guidance to the goal point.

(iii) Form the collision safety boundary and the collision cone.

(iv) *if* collision cone contains the velocity vector
Collision avoidance maneuver: Go to (v).
else
Navigation maneuver: Waypoint guidance to the goal point.

(v) Find the aiming point.

(vi) Guide the quadrotor to the aiming point.

Figure 5. Collision avoidance algorithm

6. Numerical Simulation Results

6.1 Simulation Settings

The parameters of the camera used in the numerical simulation are summarized in Table 1. And those of the quadrotor are summarized in Table 2.

Table 2. Parameters of the quadrotor

Quadrotor			
m	$0.65kg$	k_{tx}	$2kg/s$
J_x	$7.5e^{-3}kgm^2$	k_{ty}	$2kg/s$
J_y	$7.5e^{-3}kgm^2$	k_{tz}	$4kg/s$
J_z	$1.3e^{-2}kgm^2$	k_{rx}	$1.5e^{-1}kgm^2/s$
d	$0.25m$	k_{ry}	$1.5e^{-1}kgm^2/s$
G	$\begin{bmatrix}0 & 0 & 9.81m/s^2\end{bmatrix}^T$	k_{rz}	$1.5e^{-1}kgm^2/s$

The weighting matrices of the LQ tracker and the designed optimal gain matrix of the attitude control subsystem are

$$Q_a = diag\begin{bmatrix}10, & 6, & 10, & 6, & 10, & 6\end{bmatrix} \quad (29)$$

$$R_a = diag\begin{bmatrix}0.1, & 0.1, & 0.1\end{bmatrix} \quad (30)$$

$$k_a = \begin{bmatrix}10.0 & 8.94 & 0 & 0 & 0 & 0\\ 0 & 0 & 10.0 & 8.94 & 0 & 0\\ 0 & 0 & 0 & 0 & 10.0 & 8.94\end{bmatrix} \quad (31)$$

And, those of the position control subsystem are

$$Q_p = diag\begin{bmatrix}2, & 9, & 2, & 9, & 2, & 9\end{bmatrix} \quad (32)$$

$$R_p = diag\begin{bmatrix}1, & 1, & 1\end{bmatrix} \quad (33)$$

$$k_p = \begin{bmatrix}1.4 & 3.44 & 0 & 0 & 0 & 0\\ 0 & 0 & 1.4 & 3.44 & 0 & 0\\ 0 & 0 & 0 & 0 & 1.4 & 3.44\end{bmatrix} \quad (34)$$

The threshold determining the obstacle in the depth map is set to $4m$. Initial position of the quadrotor is $\begin{bmatrix}20 & 0 & 7\end{bmatrix}^T$, and goal point is $\begin{bmatrix}0 & 0 & 7\end{bmatrix}^T$. There exists an obstacle centered at $\begin{bmatrix}10 & 0 & 7\end{bmatrix}^T$ with a radius $1.5m$. The quadrotor initially moves to the goal point using a waypoint guidance. To avoid excess maneuvers and gain stability of the quadrotor, the desired roll and pitch angles should be in the range of $\pm 15^\circ$.

6.2 Results

Figure 6 shows the trajectory of the quadrotor. As soon as the distance between the quadrotor and the obstacle reaches the predefined threshold, the obstacle appears at the depth map and the quadrotor starts to perform collision avoidance maneuver as shown in Fig. 6. When the obstacle is totally outside of the camera's field of view or the collision cone does not contain the velocity vector, then the quadrotor returns to the original navigation maneuver. Figures 7—9 show the time responses of the attitude of the quadrotor, velocity, and thrust input, respectively. Dotted lines in Fig. 7 are commands from the control input, and solid lines are the actual responses. The control responses are within the saturation boundaries, which avoids excess maneuvers of the quadrotor. Figure 10 shows the distance between the quadrotor and the obstacle during the flight, and Fig. 11 shows collision avoidance flag, which is 1 when the quadrotor is performing collision avoidance maneuver and 0 when it is not.

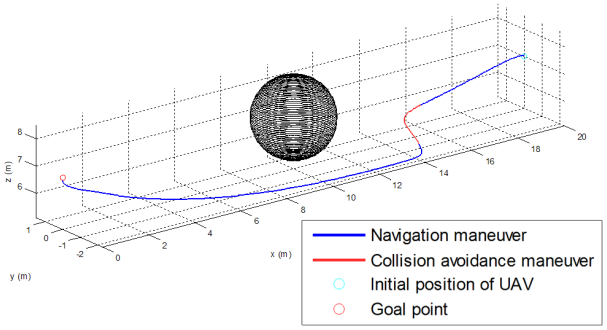

Figure 6. Trajectory of the quadrotor

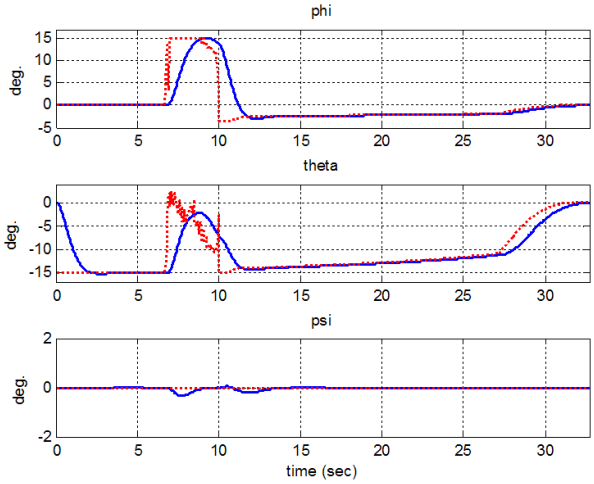

Figure 7. Time histories of roll, pitch, and yaw angles

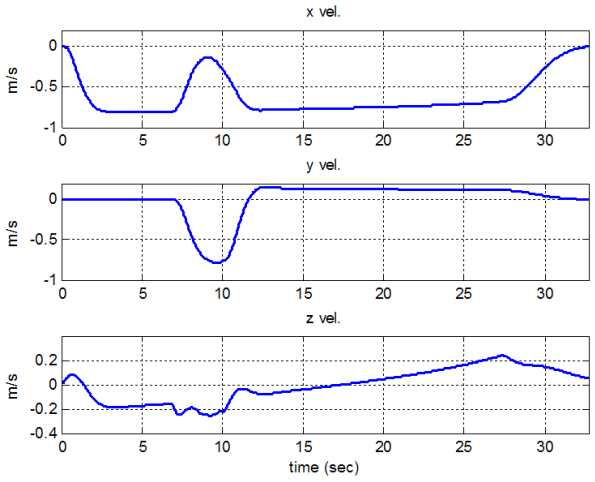

Figure 8. Time histories of velocity

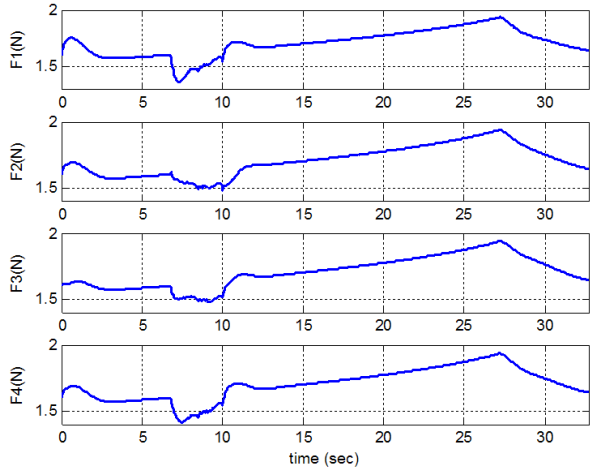

Figure 9. Rotor thrust input histories

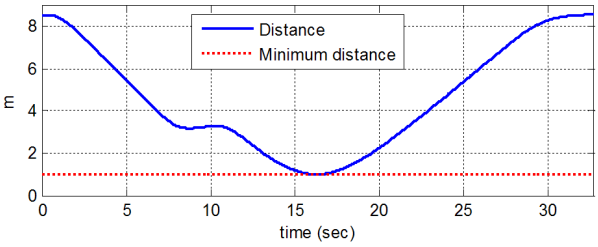

Figure 10. Distance from obstacle

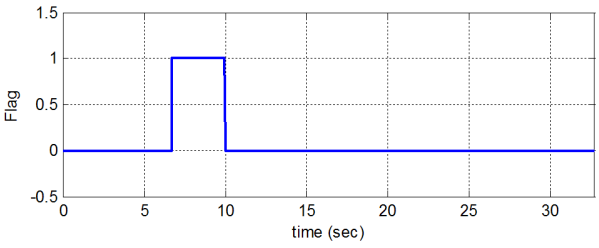

Figure 11. Collision avoidance flag

7. Conclusion

In this study, collision avoidance of a quadrotor based on stereo vision is proposed. The quadrotor system is divided into position and attitude control subsystems, and controllers are designed separately. To deal with an under-actuated problem of the quadrotor, two outputs, roll and pitch angles, of the attitude control subsystem are regarded as the virtual inputs of the position control subsystem. Linear quadratic tracker is adopted to design the controller with the help of feedback linearization. Obstacle is detected through the depth map of the stereo vision. Then, collision safety boundary and collision cone are constructed. When the velocity vector of the quadrotor is inside the collision cone, the quadrotor performs collision avoidance maneuver which guides it to the aiming point. The performance of the collision avoidance is verified through numerical simulation. For future work, the proposed algorithm will be applied to a problem with a lot of randomly shaped obstacles. Furthermore, collision avoidance for moving obstacles will be considered.

Acknowledgments

The Authors gratefully acknowledge the support from UTRC (Unmanned Technology Research Center) at KAIST (Korea Advanced Institute of Science and Technology), originally funded by DAPA, ADD.

References

[1] A. Mujumdar, and R. Padhi, "Evolving Philosophies on Autonomous Obstacle/Collision Avoidance of Unmanned Aerial Vehicles," *Journal of Aerospace Computing, Information, and Communication*, Vol. 8, No. 2, 2011, pp. 17-41.

[2] Y. Watanabe, A. J. Calise, and E. N. Johnson, "Vision-Based Obstacle Avoidance for UAVs," *AIAA Guidance, Navigation, and Control Conference*, Hilton Head, SC, August 2007.

[3] H. Choi, and Y. Kim, "Vision-based Reactive Collision Avoidance Algorithm for Unmanned Aerial Vehicle," *AIAA Guidance, Navigation, and Control Conference*, Portland, OR, August 2011.

[4] S. H. Lee, "Design of Waypoint Guidance Law with Flight Envelope Protection for Quadrotor Unmanned Aerial Vehicle," MS Thesis, Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul, Korea, August 2011.

[5] H. K. Khalil, *Nonlinear Systems*, Prentice Hall, Upper Saddle River, NJ, 2002, pp. 505-544.

[6] D. A. Forsyth, and J. Ponce, *Computer Vision A Modern Approach*, Prentice Hall, Upper Saddle River, NJ, 2003, pp. 169-250.

[7] A. Chakravarthy, and D. Ghose, "Obstacle Avoidance in a Dynamic Environment: A Collision Cone Approach," *IEEE Transactions on Systems, Man, and Cybernetics–Part A: Systems and Humans*, Vol. 28, No. 5, 1998, pp. 562-574.