

## 무인항공기용 실시간 장애물 탐지 및 회피 경로 생성 알고리즘

# Real-time Obstacle Detection and Avoidance Path Generation Algorithm for UAV

고 하 윤<sup>1</sup> · 백 중 환<sup>1\*</sup> · 최 형 식<sup>2</sup>

<sup>1</sup>한국항공대학교 항공전자정보공학부

<sup>2</sup>한국항공우주연구원 무인기체계부

Ha-Yoon Ko<sup>1</sup> · Joong-Hwan Baek<sup>1\*</sup> · Hyung-Sik Choi<sup>2</sup>

<sup>1</sup>School of Electronics and Information Engineering, Korea Aerospace University, Gyeonggi-do, 10540, Korea

<sup>2</sup>Korea Aerospace Research Institute, Daejeon, 169-84, Korea

### [요 약]

본 논문에서는 무인항공기용 실시간 장애물 탐지 및 회피 경로 생성 알고리즘을 제안한다. 2-D Lidar를 이용하여 장애물을 검출하고, 검출 정보는 지역 회피 경로 생성을 위한 실시간 히스토그램 생성과 목표 지점까지 전역 회피 경로 생성을 위해 사용되는 2-D SLAM 지도를 생성하는데 사용된다. 지역 회피 경로 생성을 위한 VFH 알고리즘은 장애물들이 벡터 방향과 거리에 따라 얼마큼 분포되어 있는지에 대한 실시간 히스토그램을 생성하고, 이 히스토그램은 근접 장애물 검출 시 지역 회피 경로를 생성하는데 사용된다. 기존의 RRT\*-Smart 알고리즘의 한계로 인해 Modified RRT\*-Smart 알고리즘을 제안한다. 이 알고리즘은 새로운 노드가 생길 때 목표 지점과의 직선 경로 여부를 판단하고, 목표 지점 방향으로 생성되도록 방향성을 부여하며, 노드의 길이를 확률적으로 나누어 일정한 단위의 길이가 아닌 랜덤 단위의 길이로 퍼뜨림으로써 보다 적은 비용으로 목표 지점까지의 효율적인 전역 회피 경로를 생성한다. 본 논문에서는 효율적인 회피 경로를 생성하여 회피 기동함을 다양한 시뮬레이션 실험 환경을 통해 검증하였다.

### [Abstract]

In this paper, we propose a real-time obstacle detection and avoidance path generation algorithm for UAV. 2-D Lidar is used to detect obstacles, and the detected obstacle data is used to generate real-time histogram for local avoidance path and a 2-D SLAM map used for global avoidance path generation to the target point. The VFH algorithm for local avoidance path generation generates a real-time histogram of how much the obstacles are distributed in the vector direction and distance, and this histogram is used to generate the local avoidance path when detecting near fixed or dynamic obstacles. We propose an algorithm, called modified RRT\*-Smart, to overcome existing limitations. That generates global avoidance path to the target point by creating lower costs because nodes are checked whether or not straight path to a target point, and given arbitrary lengths and directionality to the target points when nodes are created. In this paper, we prove the efficient avoidance maneuvering through various simulation experiment environment by creating efficient avoidance paths.

**Key word :** UAV, Obstacle avoidance, Path planning, VFH, RRT.

<https://doi.org/10.12673/jant.2018.22.6.623>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 12 November 2018; Revised 27 November 2018

Accepted (Publication) 27 December 2018 (30 December 2018)

\*Corresponding Author; Joong-Hwan Baek

Tel: +82-2-300-0125

E-mail: jhbaek@kau.ac.kr

## 1. 서론

최근 무인 항공기에 대한 군사 및 민간분야에서 다양한 연구가 진행되고 있다. 그로 인해 무인항공기 시장도 확대되고 개인 소유의 무인항공기도 상용화되고 있는 추세이다. 하지만 무인항공기의 오작동 및 조종 미숙으로 인한 충돌, 낙하로 인한 2차 파손과 손괴가 발생하기도 하며 특정 임무를 수행하기 위해 무인항공기 스스로의 자율적 비행이 필요로 해졌다.

이를 해결하기 위해 무인 항공기 스스로 실시간 장애물 데이터를 처리함으로써 장애물 탐지 및 회피 경로 생성 임무를 수행하는 연구가 활발히 진행 중에 있다. 무인항공기 스스로 장애물을 탐지하고 회피하기 위해서 저가형 광학 카메라를 사용한 장애물 검출 기술이 주로 연구되어 왔지만, 최근에는 스테레오 센서, 저가형 2-D Lidar (two-dimensional light detection and ranging) 기반의 탐지 및 회피 기술이 연구되었다 [1], [2].

본 논문에서는 2-D Lidar를 이용한 실시간 장애물 탐지 및 회피 경로 생성 알고리즘을 제안한다. 2-D Lidar를 통해 얻은 장애물 정보를 이용하여 VFH (vector field histogram) 알고리즘 기반의 지역 회피 경로 생성 알고리즘을 사용하며, 무인항공기와 검출된 장애물의 상대위치를 이용한 동시적 위치추정 및 2-D 지도 생성 알고리즘인 Hector SLAM (simultaneous localization and mapping)에 의해 생성된 지도를 이용하여 Modified RRT\*-Smart 전역 회피 경로 생성 알고리즘을 사용한다 [3]. VFH 기반의 지역 회피 경로 생성 알고리즘은 전역 경로에 따른 이동 또는 호버링 상태에서 근접 고정 장애물 및 동적 장애물에 대한 지역 회피 경로를 생성한다. Modified RRT\*-Smart (rapidly exploring the random tree star-smart) 기반의 전역 회피 경로 생성 알고리즘은 최초의 RRT 알고리즘을 확장시킨 RRT\*Smart 알고리즘의 단점을 보완하여 보다 적은 연산, 노드, 거리 비용으로 효율적인 전역 회피 경로를 생성한다. VFH 기반 지역 회피 경로 생성 알고리즘은 단독적 지역 회피 경로 생성뿐만 아니라, Modified RRT\*-Smart 기반의 전역 회피 기동 중 근접 장애물 검출 시에도 발동되어, 발동 시 지역 회피 기동 후 전역 회피 경로는 재생성 되어 목적 지점까지 전역 회피 기동한다. 이 때, 급격한 무인기의 비행으로 인한, Hector SLAM 지도 생성에서 발생하는 위치추정 오류 등을 고려하여 초속 1.25m/s로 비행 속도를 제한하여 비행하도록 제어하였다. 제안하는 알고리즘의 성능은 3-D 공간 시뮬레이터인 GAZEBO를 이용하여 검증하였다.

본 논문의 2장에서는 Hector SLAM 및 장애물 회피 경로 생성 알고리즘에 대해 설명한다. 3장에서는 시뮬레이션 기반의 실험 방법 및 결과를 보이며, 4장에서 결론을 맺는다.

## II. Hector SLAM 및 장애물 회피 경로 생성 알고리즘

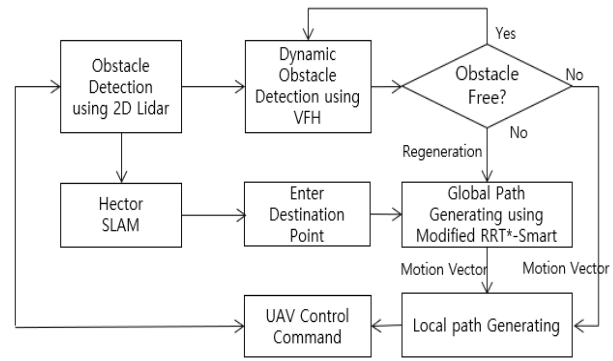


그림 1. 실시간 장애물 탐지 및 회피 경로 생성 알고리즘

Fig. 1. Real-time obstacle detection and avoidance path generation algorithm.

본 논문의 장애물 회피 경로 생성 알고리즘은 무인항공기의 호버링 상태에서는 근접 장애물 검출 시 지역 회피 경로 기동을 하며, 목표 지점 이동 명령 시에는 Hector SLAM 지도 기반으로 장애물 회피 기동이 가능한 전역 회피 경로를 생성한다. 전역 경로를 따라 회피 기동 중 경로 상의 새로운 근접 고정 장애물 및 동적 장애물 검출 시 지역 회피 경로 생성을 이용한 회피 기동 후 전역 회피 경로를 재 생성한다. 전체 임무 수행을 위한 알고리즘을 위 그림 1에 보인다.

### 2-1 2-D Lidar를 이용한 장애물 검출

본 논문에서는 360° 측정이 가능한 2-D Lidar를 사용하였다. RPLidar를 통하여 얻은 장애물의 각도  $\theta$ 와 거리  $r$ 을 이용하여 장애물의 상대 위치  $p_x, p_y$ 는 아래 식 (1)과 같이 계산한다.

$$[p_x, p_y] = [r \cos \theta, r \sin \theta] \tag{1}$$

획득한 장애물 정보는 VFH 기반의 지역 회피 경로 생성에 필요한 실시간 히스토그램 생성과 Modified RRT\*-Smart 전역 회피 경로 생성에 필요한 Hector SLAM 지도 생성에 사용된다.

### 2-2 Hector SLAM

본 논문에서는 전역 회피 경로 생성을 위해 Hector SLAM을 이용하여 지속적인 지도 생성을 수행한다.

Hector SLAM은 무인항공기와 검출된 장애물의 상대위치를 이용한 동시적 위치추정 및 2-D 지도 생성 알고리즘이다. 생성된 SLAM 지도는 전역 회피 경로 생성 시 무인항공기가 통과할 수 없는 장애물 지역의 지도 data 값을 0으로 만든다. 그림 2는 시뮬레이션 공간 상의 2-D Lidar가 탑재된 무인항공기와 장애물을 보이며, 그림 3은 ROS (robot operating system) 3-D 시각화 툴인 Rviz를 이용하여 생성된 지도를 보인다 [4].

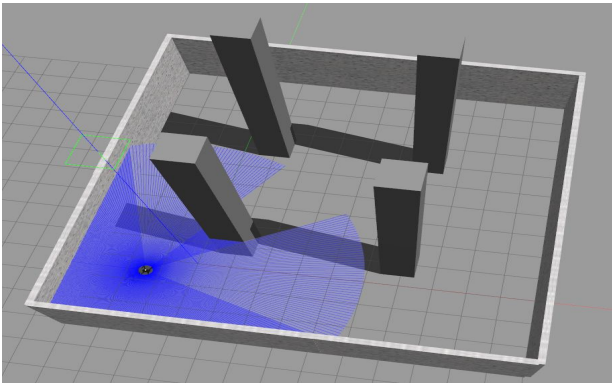


그림 2. GAZEBO 시뮬레이션  
Fig. 2. GAZEBO simulation.

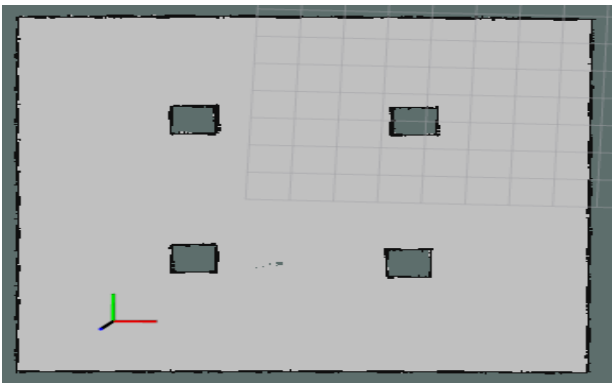


그림 3. Rviz 3-D 시각화 툴  
Fig. 3. Rviz 3-D visualization tool.

**2-3 VFH(Vector Field Histogram) 알고리즘**

본 논문에서 VFH 기반의 지역 회피 경로 알고리즘은 두 가지 경우에 사용된다. 먼저 무인항공기가 호버링 중 근접 동적 장애물이 접근하는 상황에서 지역 회피 경로 생성에 사용되며, 다음은 Modified-RRT\*-Smart 알고리즘 기반 무인항공기의 전역 회피 경로 기동 중 근접 고정 장애물 및 동적 장애물 검출 시 지역 회피 경로 생성에 사용된다.

기존의 VFF (vector field force) 알고리즘은 로봇 주위에 흩어진 장애물로부터 발생하는 반발력을 모두 합하여 하나의 반발력 벡터로 표시한다. 이는 장애물이 로봇 주변에 어떻게 분포하고 있는지에 대한 정보가 사라지는 단점이 있다. VFH 알고리즘은 이런 단점을 극복하기 위하여 2-D Lidar 주변의 360° 전 방향을 작은 섹터로 나누고 각 섹터에 장애물에 대한 정보를 저장한다 [5], [6], [7]. 따라서 주변에 장애물들이 어느 방향에 분포해있는지 알 수 있고 무인항공기에 근접한 장애물에 대해 실시간 히스토그램을 형성할 수 있다. 이 히스토그램은 근접 장애물 등장 시 회피 방향 결정 및 전역 회피 경로에 따른 지역 회피 경로 생성에 사용된다.

무인항공기의 360°방향에 대하여 Lidar 스텝 각으로 섹터를

나누고 장애물이 있는 섹터에 거리에 따른 가중치를 부여한다. 장애물이 극 히스토그램의 어느 섹터에 속하는지 다음 식 (2)와 같이 계산한다.

$$\phi = \text{INT}\left(\frac{\theta}{s}\right) \tag{2}$$

$\phi$  는 섹터의 인덱스 값이며  $s$ 는 한 섹터의 해상도이다. 섹터가 정해지면 섹터별 장애물 가중치  $m_\phi$ 은 다음 식 (3)과 같이 계산한다.

$$m_\phi = \beta(\alpha - r_{\text{MIN}}) \quad (\alpha > 0, \beta > 0) \tag{3}$$

식 (3)에서  $\alpha$ 는 2-D Lidar의 최대 측정 범위이며,  $\beta$ 는 가중치 값이다.  $\alpha$  와  $\beta$  는  $r$ 값이 최대 일 때  $m_\phi$ 가 0이 되도록 설정한다.  $r_{\text{MIN}}$ 은 섹터 구간 내의 최소 장애물 거리이다.

무인항공기가 갈 수 없는 근접 장애물 회피 범위 임계치를 설정하고 각 섹터 값이 임계치보다 큰지 작은지를 판별한다. 섹터 값이 임계치보다 크다면 이동 불가능한 방향으로 지정한다.

아래 그림 4에서 블록 A, B, C는 임의의 장애물을 나타내며, 블록 C는 근접 장애물 회피 범위 내로 접근한 상황이다. 극 히스토그램에서 C 장애물 섹터의 크기가 임계치를 초과하여 이동 불가능 영역으로 지정한 것을 볼 수 있다. 장애물 A, B 부분은 임계치 범위를 초과하지 않아 이동할 수 있는 부분으로 분류된다.

VFH 기반 지역 회피 경로 생성 알고리즘에서 호버링 시에는 전역 회피 경로가 존재하지 않기에 근접 장애물 회피 경로에 따라서 이동하며, 전역 경로를 따라 이동 중에는 지역 경로는 회피 경로 벡터와 전역 회피 경로 벡터의 벡터 합으로 설정된다. 전역 회피 경로 벡터는 이동 가능 영역 중 목표지점과 가장 가까운 방향으로 가도록 임계치 각도 내 범위의 모든 경로를 탐색하면서 설정된다. 아래 그림 5에서는 전역 회피 경로 방향으로 이동 중 300° 부근에서 장애물이 발생하여 이동 가능한 영역 중 전역 회피 경로 방향과 가까운 190° 근처로 회피 경로를 설정하였다.

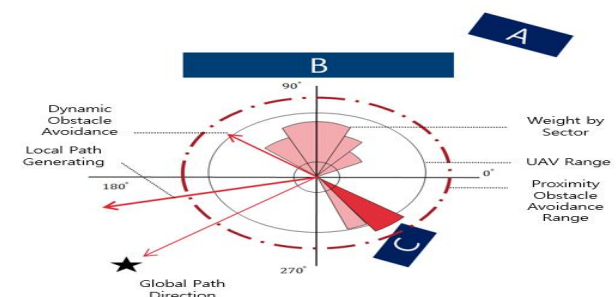


그림 4. VFH 기반 지역 회피 경로 생성  
Fig. 4. Local avoidance path generation based VFH.

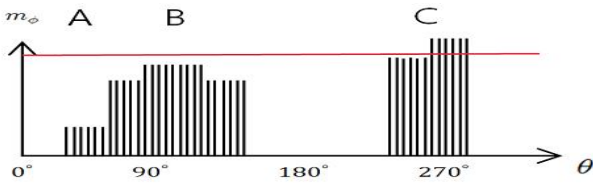


그림 5. 극 히스토그램 및 임계치  
Fig. 5. Polar histogram and threshold.

VFH 알고리즘을 활용하여 위와 같이 전역 회피 경로 기동 중 근접 고정 장애물 및 동적 장애물 등장 시 임시적인 지역 회피 경로로 이동한 뒤 다시 전역 회피 경로 생성을 수행하게 된다.

2-4 Modified RRT\*-Smart 알고리즘

RRT 알고리즘은 상태 공간 내에서 무작위로 노드를 생성하여 상태공간을 탐색하여 목적지까지의 연속적인 경로를 찾는 방법이다. RRT 알고리즘의 수도코드는 다음과 같다.

```

Algorithm RRT(
Initial configuration  $x_{init}$ ,
number of vertices in RRT  $K$ ,
incremental distance  $\Delta t$ )
 $T.Init(x_{init})$ 
for  $k = 1$  to  $K$ 
     $x_{rand} = Rand\_conf()$ ;
     $x_{near} = Nearest\_vertex(x_{rand}, T)$ ;
     $u = Select\_input(x_{rand}, x_{near})$ ;
     $x_{new} = New\_conf(x_{near}, u, \Delta t)$ ;
     $T.Add\_node(x_{new})$ ;
     $T.Add\_edge(x_{near}, x_{new}, u)$ ;
return  $T$ 
    
```

상태 공간 안에서 무작위 정점  $x_{rand}$ 를 추출한 뒤 출발점과 목적지가 포함된 트리  $T$ 를 구성한다. 트리  $T$ 에 있는 정점 중  $x_{rand}$ 와 가장 가까운  $x_{near}$ 을 찾는다.  $x_{near}$ 에서  $x_{rand}$  방향으로  $\Delta t$ 만큼 지났을 때 생성되는  $x_{new}$ 를 트리  $T$ 에 저장하고  $x_{near}$ 에서  $x_{new}$ 까지 경로를 확장해나가며 트리  $T$ 를 구성한다.

기존의 RRT 알고리즘을 확장시킨 RRT\*-Smart 알고리즘은 새로운 노드가 생성되었을 때, 주변 노드의 거리 비용 값의 비교를 수행한다. 다음으로 낮은 비용의 노드로 연결을 수정하여 목표 지점까지 경로를 생성한다. 따라서 기존 RRT 알고리즘에 비해 상대적으로 보다 적은 연산량으로 빠른 시간 안에 도착점까지의 경로를 생성할 수 있다 [8]. 하지만, RRT\*-Smart는 생성된 노드 간의 장애물 유무 판단 부분과 목표지점 방향과 다른 방향의 노드 생성 부분에 상대적으로 높은 연산량을 필요로 하고, 목표 지점까지의 직선 경로 판단을 하지 않는 단점이 존재한다. 이를 보완하기 위해 i)노드가 생길 때마다 목표 지점과

의 직선경로 여부를 판단하여 장애물이 없는 경우 직선 경로를 생성하고, ii)새로운 노드가 목표 지점 방향으로 생성되도록 방향성을 부여해주며, iii)노드 간의 길이를 확률적으로 나누어 일정한 단위의 길이가 아닌 랜덤 단위의 길이로 노드를 퍼뜨리는 Modified-RRT\*-Smart 알고리즘을 제안한다. 본 논문에서는 RRT\*-Smart 알고리즘 적용에 앞서 좀 더 빠른 경로 결정의 필요성이 있었기에 Modified-RRT\*-Smart 알고리즘을 사용하였다. Hector SLAM 지도를 기반으로 Modified-RRT\*-Smart를 이용해 생성된 전역 회피 경로는 무인항공기 회피 경로 이동 중 안전거리도 확보하며, 기존보다 검색 노드를 줄여 속도 향상이 가능하다 [9], [10]. 최종적으로 제안하는 Modified-RRT\*-Smart 알고리즘은 아래와 같다.

```

Algorithm Modified RRT*-Smart(
Initial configuration  $x_{init}$ ,
number of vertices in RRT  $K$ ,
incremental distance  $\Delta t$ )
 $T.Init(x_{init})$ 
for  $k = 1$  to  $K$ 
     $x_{rand} = Rand\_conf()$ ;
     $x_{nearest} = Nearest\_vertex(x_{rand}, T)$ ;
     $u = Select\_input(x_{rand}, x_{nearest})$ ;
    if  $x_{init,goal} = x_{nearest,rand}$ 
         $x_{new} = New\_conf(x_{nearest}, u, probabilistic \Delta t)$ ;
        if Obstacle_free( $x_{new}$ ) then
             $x_{near} = Near(T, x_{new}, V)$ ;
             $x_{min} = Chooseparent(x_{near}, x_{nearest}, x_{new})$ ;
             $T = Insertnode(x_{min}, x_{new}, T)$ ;
             $T = Rewire(T, x_{near}, x_{min}, x_{new})$ ;
            if collisions not found between  $x_{new}$  and  $x_{goal}$ 
                ( $T.Directcost$ ) = Pathoptimization( $T; x_{init}, x_{goal}$ );
return  $T$ 
    
```

III. 실험 방법 및 결과

그림 6은 2-D Lidar가 탑재된 호버링 상태의 무인항공기와 무인항공기를 향해 다가오는 동적 장애물인 원형 공으로 구성된 가상 시뮬레이션 공간을 보인다.

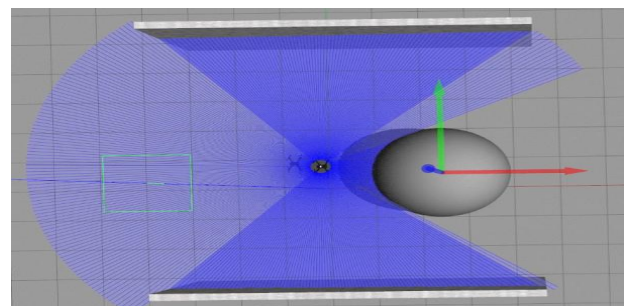


그림 6. 동적 장애물 시뮬레이션 공간  
Fig. 6. Dynamic obstacle simulation space.

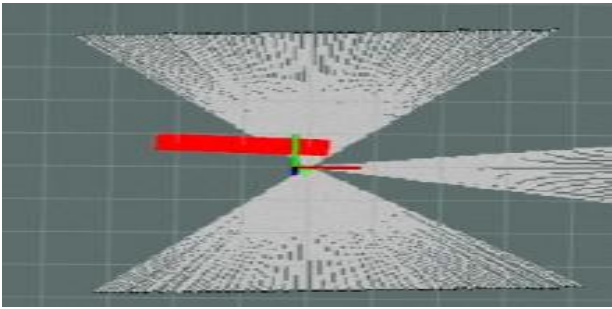


그림 7. VFH 기반 지역 회피 경로 생성  
 Fig. 7. Local avoidance path generation based VFH.

그림 7은 그림 6에 보인 시뮬레이션 공간에서 동적 장애물이 VFH 알고리즘이 발동되는 무인항공기 범위 내에 근접했을 때 생성된 지역 회피 경로 방향 및 위치를 3-D 시각 툴인 Rviz의 GUI를 통해 보인다.

그림 8은 2-D Lidar가 탑재된 호버링 상태의 무인항공기와 다수의 장애물로 구성된 가상 시뮬레이션 공간을 보인다. 그림 8에 보인 시뮬레이션 공간 상의 장애물 정보로 만들어진 Hector SLAM 지도를 기반으로 RRT\*-Smart 알고리즘을 사용하여 생성된 목표지점까지의 전역 회피 경로를 그림 9의 왼쪽, Modified RRT\*-Smart 알고리즘을 사용하여 생성된 목표지점까지의 전역 회피 경로를 그림 9의 오른쪽에 3-D 시각 툴인 Rviz의 GUI를 통해 보인다.

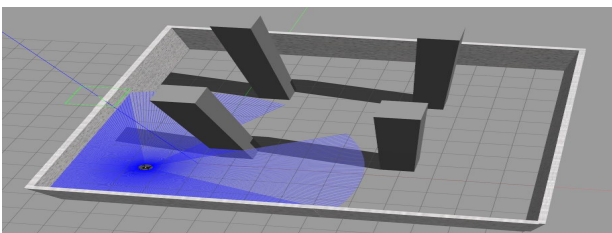


그림 8. 무인항공기 및 다수 장애물 시뮬레이션 공간  
 Fig. 8. UAV and multiple obstacle simulation space.

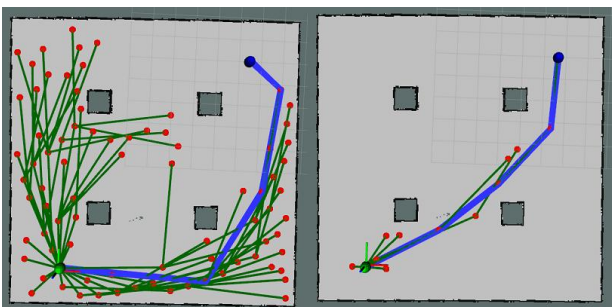


그림 9. 기존 RRT\*-Smart 알고리즘 (왼쪽), 제한하는 Modified RRT\*-Smart 알고리즘 (오른쪽)  
 Fig. 9. RRT\*-Smart algorithm (left), Modified RRT\*-Smart algorithm (right).

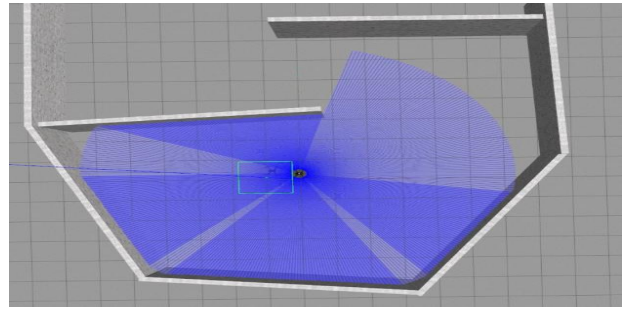


그림 10. 미탐사 지역 존재 시뮬레이션 공간  
 Fig. 10. Unknown area exists in simulation space.

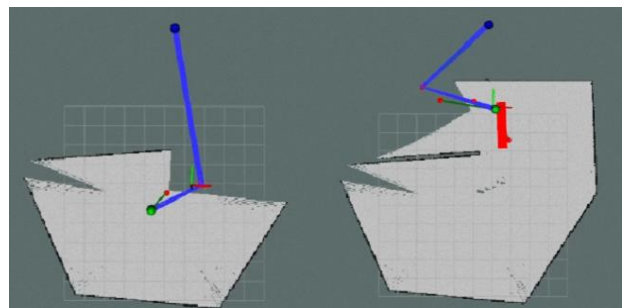


그림 11. 전역 회피 경로 생성 (왼쪽), 근접 장애물에 의한 지역 회피 경로 생성 및 전역 회피 경로 재생성 (오른쪽)  
 Fig. 11. Global avoidance path generation (left), Local avoidance path generation and global avoidance path regeneration by proximity obstacle (right).

그림 10은 그림 8과 달리 공간상의 장애물 정보를 다 취득하지 못한 가상 시뮬레이션 공간을 보인다. 그림 11은 그림 10에 보인 시뮬레이션 공간 상 Modified RRT\*-Smart 알고리즘을 사용해 생성된 전역 회피 경로를 왼쪽, 새로운 근접 고정 장애물에 의한 지역 회피 경로와 새롭게 생성된 전역 회피 경로를 오른쪽에 보인다. 그림 12는 다수의 장애물과 동적 장애물이 존재하는 가상 시뮬레이션 공간을 보인다.

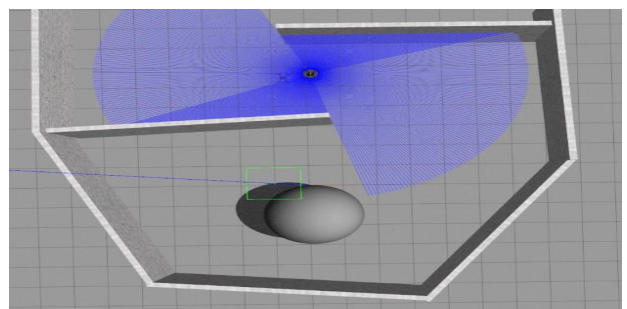


그림 12. 다수의 고정 장애물 및 동적 장애물 시뮬레이션 공간  
 Fig. 12. Multiple fixed obstacles and dynamic obstacles simulation space.

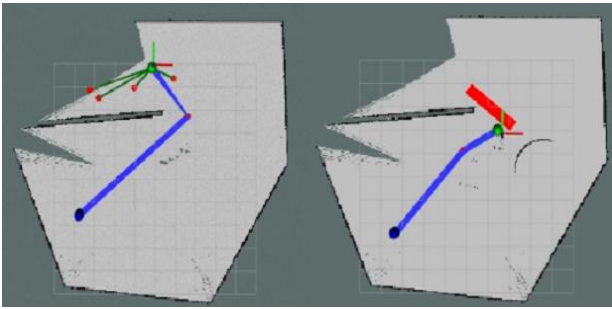


그림 13. 전역 회피 경로 생성(왼쪽), 동적 장애물에 의한 지역 및 전역 회피 경로

Fig. 13. Global avoidance path generation (left), Local avoidance path generation and global avoidance path regeneration by dynamic obstacle (right).

그림 12에 보인 시뮬레이션 공간 상 Modified RRT\*-Smart 알고리즘을 사용해 생성된 전역 회피 경로를 그림 13의 왼쪽, 동적 장애물에 의한 지역 회피 경로와 새롭게 생성된 전역 회피 경로를 그림 13의 오른쪽에 보인다.

결과적으로, 제안하는 장애물 회피 경로 생성 알고리즘을 이용하여 그림 7은 VFH 알고리즘을 통한 근접 동적 장애물에 대한 지역 회피 경로 생성을 보였으며, 그림 9는 RRT\*-Smart의 단점을 보완하기 위해 제안하는 Modified RRT\*-Smart의 i)노드가 생길 때마다 목표 지점과의 직선경로 여부를 판단하여 장애물이 없는 경우 직선 경로를 생성하고, ii)새로운 노드가 목표 지점 방향으로 생성되도록 방향성을 부여해주며, iii)노드 간의 길이를 확률적으로 나누어 일정한 단위의 길이가 아닌 랜덤 단위의 길이로 노드를 퍼뜨려 경로 생성 기법을 이용하여 보다 효율적인 전역 회피 경로를 생성하였다. 표 1을 통하여 제안하는 Modified RRT\*-Smart 알고리즘이 기존의 RRT\*-Smart 알고리즘에 비해 목표 지점까지의 전역 회피 경로를 찾기까지 걸린 시간과 생성되어진 노드 개수, 그리고 거리 비용 값이 감소되었음을 볼 수 있다.

IV. 결 론

본 논문에서는 2-D Lidar를 통해 얻은 장애물 정보로 생성된 실시간 히스토그램을 이용하여 VFH 기반의 지역 회피 경로 생성 알고리즘과 Hector SLAM 지도를 기반으로 Modified RRT\*-Smart를 이용한 전역 회피 경로 생성 알고리즘을 제안하였다. 3-D 공간 GAZEBO 시뮬레이터 상에서 VFH 알고리즘을 이용하여 동적 장애물 회피도 가능함을 보였으며, Modified RRT\*-Smart 알고리즘을 이용하여 기존 RRT\*-Smart 알고리즘보다 효율적인 전역 회피 경로를 생성함을 보였다. 또한 VFH와 Modified RRT\*-Smart를 동시에 적용하여 효율적인 지역 및

표 1. 상기 데이터 수치는 50회의 실험 결과의 평균값임  
Table 1. The data values are average values of 50 experimental results.

|                     | Path Generation Time | Number of Nodes | Distance Cost |
|---------------------|----------------------|-----------------|---------------|
| RRT*-Smart          | 5.94 seconds         | 25.72           | 16.79 meter   |
| Modified RRT*-Smart | 0.98 seconds         | 10.28           | 15.77meter    |

전역 회피 경로가 생성되는 것을 확인하였다. 임무 수행 무인항공기는 원하는 목표지점까지의 경로 비용뿐만 아니라 근접 고정 장애물 및 동적 장애물을 회피할 수 있는 안정성도 중요시 되므로 본 논문에서 제시한 방법은 무인항공기 회피 기동을 가능하게 한다. 향후에는 전역 회피 경로 이동 중 지속적인 전역 회피 경로 생성을 통해 기존 경로와의 거리 비용 값을 비교하여 비용이 더 적은 경로를 회피 기동하도록 회피 경로를 재설정하는 연구가 필요하다. 또한 3-D 공간상의 지도를 기반으로 전역 경로를 생성하고 자율 비행하도록 하는 연구도 필요할 것으로 보인다.

Acknowledgments

본 연구는 한국항공우주연구원 위탁사업 2018-04-301, “회전적 무인항공기 장애물 인식 및 충돌 회피 연구(3/3)”의 지원을 받았습니다.

References

[1] A. J. Barry, and R. Tedrake, “Pushbroom stereo for high-speed navigation in cluttered environments,” in *International Conference on Robotics and Automation (ICRA)*, Seattle: WA, pp. 3046-3052, 2015.

[2] M. Siam, R. ElSayed, and M. ElHelw. “On-board multiple target detection and tracking on camera-equipped aerial vehicles,” in *2012 IEEE International Conference on Robotics and Biomimetics*, Guangzhou: China, pp. 2399-2405, 2012.

[3] S. Kohlbrecher, J. Meyer, T. Graber, K. Petersen, U. Klingauf, and O. V. Stryk “Hector open source modules for autonomous mapping and navigation with rescue robots,” *Robot Soccer World Cup*, 17th ed. Heidelberg, Germany: Springer, 2013.

[4] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng “ROS: an open-source Robot Operating System,” *ICRA Workshop on Open Source Software*, Vol. 3, No. 3.2, pp. 1-6, 2009.

- [5] S. W. Lee, Y. G. Moon, S. H. Kim, and M. C. Lee, "Navigation technique of unmanned vehicle using potential field method," *Transaction of Korean Society of Automotive Engineers*, Vol 19, No. 4, pp. 8-15, 2011.
- [6] A. H. Hamad, and F. B. Ibrahim, "Path planning of mobile robot based on modification of vector field histogram using Neuro-Fuzzy algorithm," *International Journal of Advanced Computer Technology(IJACT)*, Vol. 2, No. 3, pp. 129-138, 2010.
- [7] Y. K. Hwang, and N. Ahuja, "A potential field approach to path planning," *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 1, pp. 23-32, February, 1992.
- [8] F. Islam, J. Nasir, U. Malic, Y. Ayaz, and O. Hasan, "RRT\*-Smart: Rapid convergence implementation of RRT\* towards optimal solution," in *International Conference Mechatronics and Automation(ICMA)*, Chengdu: China, pp. 1651-1656, 2012.
- [9] J. Ge, F. Sun, and C. Liu, "RRT-GD : An efficient rapidly-exploring random tree approach with goal directionality for redundant manipulator path planning," in *Robotics and Biomimetics (ROBIO0)*, Qingdao: China, pp. 1983-1988, 2016.
- [10] W. G. Aguilar, D. S. Sandoval, J. Caballeros, L. G. Alvarez, A. Limaico, G. A. Rodríguez, and F. J. Quisaguano, "Graph Based RRT Optimization for Autonomous Mobile Robots", in *International Conference on Intelligent Science and Big Data Engineering(IScIDE)*, Springer: Switzerland, pp. 12-21, 2018.



**고 하 윤 (Ha-Yoon Ko)**

2017년 2월 : 한국항공대학교 정보통신공학 (공학사)  
 2017년 3월 ~ 현재 : 한국항공대학교 일반대학원 항공전자정보공학과 석사과정  
 ※관심분야 : 영상처리, 패턴인식



**백 중 환 (Joong-Hwan Baek)**

1981년 2월 : 한국항공대학교 항공통신공학 (공학사)  
 1987년 7월 : 오클라호마주립대학원 전기 및 컴퓨터공학 (공학석사)  
 1991년 7월 : 오클라호마주립대학원 전기 및 컴퓨터공학 (공학박사)  
 1992년 ~ 현재 : 한국항공대학교 항공전자정보공학부 교수  
 ※관심분야 : 영상처리, 패턴인식, 멀티미디어, 가상현실



**최 형 식 (Hyoung-Sik Choi)**

2000년 2월 : 울산대학교 항공우주공학 (공학사)  
 2002년 2월 : 울산대학교 항공우주공학 (공학석사)  
 2011년 8월 : KAIST 항공우주공학 (공학박사)  
 2002년 ~ 현재 : 한국항공우주연구원 무인기체계부 책임연구원  
 ※관심분야 : 무인기, 비행제어, 시뮬레이션, 충돌회피