

## 공중충돌경보시스템 평가를 위한 통합 무인기 시뮬레이션환경 연구

# Study on the Integrated UAV Simulation Environment for the Evaluation of the Midair Collision Alarm System

문성엽 · 김주영 · 이동우 · 백경민 · 김진실 · 나중화\*  
한국항공대학교 항공전자공학과

Seong-yeop Mun · Ju-young Kim · Dong-woo Lee · Gyeong Min Baek · Jin Sil Kim · Jongwhoa Na \*

Department of Avionics & Electronics, Korea Aerospace University, Gyeonggi-do 412-791, Korea

### [요 약]

무인기 상용화를 위해서는 유인기 수준의 안전성을 확보할 수 있는 공중 및 지상의 충돌경보 및 회피시스템 (sense and avoid or SAA) 개발과 검증이 필요하다. SAA 검증을 위한 비행시험은 높은 시험비용과 사고위험 때문에 많은 시험사례(test case)를 검토하기 어려우므로 시뮬레이션 시험으로 보완하는 것이 필수적이다. SAA 시뮬레이션 시험을 위해 flight simulator, Matlab/Simulink 시뮬레이터와 항전장비 시뮬레이션 모델들이 서로 연동하는 통합 무인기 시뮬레이션 환경을 구축하였다. 사례연구로서 TCAS 충돌경보 simulink 모델을 개발하고 Flight Gear와 연동하여 통합 무인기 시뮬레이션 환경을 구축하였고 이를 encounter model을 이용하여 검증하였다. 통합 무인기 시뮬레이션 환경을 활용하면 항전장비 개발주기의 개념설계 단계부터 부품 및 시스템의 성능/신뢰성 분석을 시작할 수 있다.

### [Abstract]

For the commercialization of unmanned aircraft, we must validate the safety of the air/ground collision alert systems (CAS). The validation procedure of CAS requires the flight test which is not only expensive but also dangerous. To alleviate this problem, we need the simulation based validation process for the CAS. We developed an integrated UAV simulation (IUS) environment which interconnect the flight simulator, the Matlab/Simulink, and a target avionics simulation model. We developed the collision warning module of the TCAS and tested using IUS and flight encounter models. Using IUS, we can evaluate the performance and reliability of a target avionic system at the preliminary design stage of a development life cycle.

**Key words** : Sense and avoid, Traffic collision avoidance system, Unmanned aerial vehicle simulator, Encounter model, Integrated unmanned aerial vehicle simulation.

<http://dx.doi.org/10.12673/jant.2015.19.4.288>



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 27 May 2015; Revised 1 June 2015  
Accepted (Publication) 6 August 2015 (30 August 2015)

\*Corresponding Author; Jongwhoa Na

Tel: +82-2-300-0410

E-mail: [jwna@kau.ac.kr](mailto:jwna@kau.ac.kr)

## 1. 서론

구글, 아마존 등 글로벌 기업들이 무인항공기(무인기)를 활용하는 새로운 시장을 개척하려고 시도하고 있으나, 충돌사고 위험요인 때문에 FAA가 제한적으로 운용허가를 발급하여 상용 무인기 서비스의 사업화가 지연되고 있다. 무인기의 사업적 성공가능성을 인식한 많은 사람들이 군용으로 사용되던 무인기를 민수용으로 응용하려는 시도가 기존의 감시, 경찰 분야는 물론 농수산업 등 전통산업, 재난, 교통, 의료, 미디어 등 산업 전 분야로 확장되었다. 그러나, 무인기는 유인항공기와는 달리 통신두절이나 장비결함으로 인한 고장발생 시 사고발생 확률이 높은 위험한 장비이다. 2014년 6개월간 미국연방항공청(FAA)에 보고된 드론과 항공기와의 near-collision 횟수가 25회에 도달하는 것으로 보고되었으며, FAA의 UAS Accident and Incident Preliminary Report는 2010년부터 2014년 8월까지 104회의 무인기 충돌 사건 및 사고를 보고하였다 [1],[2]

무인기의 상용화의 조건은 유인기 수준의 안전성을 확보할 수 있는 공중 및 지상 충돌경보 및 회피 (sense and avoid or SAA) 시스템의 개발이다. 그러나 무인기용 SAA 장비의 성능을 검증하려면 충돌가능성이 높은 항로의 유·무인기 다수를 대상으로 SAA 기능을 검증해야 한다. SAA기능의 검증은 여러대의 유·무인기 항로를 시험해야 하므로 시험항목의 수가 매우 많고, 항공기가 충돌하는 상황에 대한 시험이므로 유사시 시험장비 및 인력의 손실이 발생할 가능성이 상존한다. 그러므로 SAA검증은 비행시험과 시뮬레이션 시험을 상호 보완적으로 수행하는 것이 필수적이다.

이번 연구에서는 무인기의 SAA 알고리즘의 중요한 구성요소의 하나인 공중 충돌경보 알고리즘 검증을 위하여 다양한 시뮬레이터들을 연동하는 통합 무인기 시뮬레이션 환경을 구축하였다. 기존의 충돌경보 알고리즘 검증은 MATLAB으로 구현하여 특정한 알고리즘만을 제한된 상황에서 검증하였다 [3],[4]. 본 연구는 충돌경보 알고리즘을 Simulink 모델로 개발하고 이를 FlightGear, X-Plane 등의 flight simulator, Simulink aerospace package, 그리고 지상제어시스템과 연동하는 통합 무인기 시뮬레이션 환경을 구축하여 기존의 검증 방법을 확장하였다.

제안된 시뮬레이션 환경의 장점은 세 가지로 요약할 수 있다. 먼저 무인기, 지상국, 항전장비 시뮬레이션 모델들이 연결된 통합 무인기 시스템 수준의 시뮬레이션 환경을 사용하면 항전장비 개발의 초기단계인 개념설계단계에서 전체 시스템 수준의 분석에서 알 수 있는 평가지표의 분석을 조기에 실시할 수 있다. 항전장비 개념설계 단계에서 경험적·단편적 방법으로 설계 및 구현하는 경우, 단위·통합시험 결과가 목표 성능 및 신뢰성에 미달하거나 설계변경이 발생하는 경우 고장감내 기능을 변경하고 재설계를 시작해야 한다. 그러나 개념설계 단계에 통합 시뮬레이션을 수행하여 이러한 반복 업무를 최소화함으로써 개발 기간 및 비용을 절감하고, 품질·신뢰성을 향상시킬 수

있다. 두 번째는 지상제어시스템의 지상조종사가 항공기를 직접 조종할 수 있으므로 설정된 정적(static) 시험뿐만 아니라 동적(dynamic) 시험을 동시에 시험할 수 있으므로 시험항목 (test cases)의 확장이 가능하다. 초기단계부터 광범위한 사례에 대하여 시험하면 개발단계의 중요시점 혹은 양산단계에서 발생하는 수정 및 리콜 (recall)로 인한 사업성 감소문제를 최소화 할 수 있다. 세 번째로 본 시뮬레이터를 신형 무인기 시스템의 개발초기부터 조종사·정비사 교육·훈련용으로 활용이 가능하므로 상용 서비스 시스템의 조기 사업화를 지원한다.

개발된 통합 무인기 시뮬레이터를 검증하기 위하여 1단계로 FAA에서 지정한 encounter 모델을 개발된 통합 시뮬레이터에서 운항하여 Simulink로 구현된 충돌경보 알고리즘을 검증하였다. 후속연구에서는 ADS-B 및 EO/IR 기반 SAA 알고리즘의 Simulink모델 및 개발된 SAA 알고리즘을 탑재한 항공 임베디드 탑재장비 시뮬레이션 모델의 검증 연구를 수행할 계획이다.

본 논문의 구성은 다음과 같다. 2장은 시뮬레이션 기술동향에 대해서 검토하고, 3장은 무인기 공중충돌경보 Simulink 모델을 소개하고, 4장에서는 Matlab/Simulink와 Flight gear로 구성된 통합 무인기 시뮬레이션 환경에 대해서 설명한다. 그리고 5장에서는 무인기 시뮬레이터에 항공기 시뮬레이션 시험결과를 설명하며, 6장에서는 결론 및 향후 연구를 제시 한다.

## II. 공중충돌회피시스템 기술현황

무인항공기에는 충돌회피를 할 수 있는 상용 장비 및 인증기준이 마련되지 않았다[5]. 그러므로 무인항공기의 충돌회피는 유인항공기의 공중 충돌회피 시스템을 적용시켜 무인 공중충돌방지시스템을 구축하는 방법을 사용하고 있다. 유인항공기의 공중 충돌회피 시스템은 탑재기반 및 지상기반 기술로 분류할 수 있다. 본 논문에서는 탑재기반 기술에 관하여 연구하였으며 탑재기반 협업형 (cooperative) 및 단독형 (non-cooperative) 센서들을 표 1에 요약하였다[6].

표 1. 탑재기반 공중충돌회피 시스템의 분류  
Table 1. Classification of collision avoidance system.

기술 명	센서 명	비고
협업형	TCAS	- 모든 항공기 장착 필요 - 지상 장애물 감지불가
	ADS-B	
단독형	Radar	- 약 기상 조건 운용가능 - 고 비용, 중량, 고 전력
	Laser	
	Sonar	
	Electro-Optical	- 환경 조건에 민감 - 저비용, 경량, 저전력
	Infrared	
Acoustic		

## 2-1 공중 충돌경보 및 회피 기술개요

협업형 (cooperative) 센서는 복수의 항공기에 탑재되어 상호 간에 정보를 교환하여 충돌을 검출 및 회피하는 기술로서 이미 성능이 확인된 기술이다. 그러나 모든 항공기간에 호환되는 충돌방지 프로토콜을 갖춘 장비가 장착된 경우에만 적용가능하고 지형지물이나 건물 등의 지상 장애물은 감지 불가능하다는 한계가 있다.

단독형(non-cooperative) 센서는 능동형과 수동형 센서로 구분된다. 먼저 능동형 센서에는 레이더 및 레이저 시스템이 있다. 두 시스템은 시계가 좋지 않은 비행 상황에서도 운용이 가능한 장점이 있으나 운용비용이 높고, 소요전력과 중량의 문제로 무인항공기에 도입하기 용이하지 않다. 한편, 수동형 센서에는 electro-optical, infrared, acoustic 등이 있다. 이들은 능동형 센서보다 가볍고 유인항공기 조종사가 인지하는 것과 동일한 감각을 제공해주는 장점이 있지만 시계조건에 따라 충돌 위험 감지 성능이 민감하게 변하는 단점이 있다[5].

## 2-2 공중 충돌경보 및 회피 시뮬레이션 기술

무인기의 공중 충돌경보 및 회피 알고리즘 개발을 위한 연구 개발이 국내외적으로 활발히 진행되어 광범위하므로 본 연구는 상용 시스템에서 많이 언급되는 소형 무인기에 적합한 단독형 및 수동형 센서기반 기술에 대하여 간략하게 소개한다. 호주 Queensland 대학교에서는 2대의 카메라 이미지를 이용해서 충돌 상황을 시뮬레이션 하는 IBCASE (image based collision avoidance simulation environment)를 MATLAB으로 SAA 알고리즘을 개발하였다[7]. 나폴리 Federico II 대학교에서는 카메라를 이용한 SAA 알고리즘을 개발하고 항공기를 포함한 HILS (hardware in loop system) 검증을 수행하였다[8]. 국내에서는 서울대학교에서는 영상센서를 이용해서 무인항공기가 반응적 충돌회피 알고리즘을 제안하고 수치 시뮬레이션을 수행하였다[9]. 한국과학기술원은 근거리 비행체 탐지 및 추적 영상처리 알고리즘을 개발하여 소형 무인기의 충돌모사 자동비행 실험 중 촬영한 영상에 적용하였다[10].

일반적으로 항공에서는 유무인기가 혼용되므로 SAA는 수동형 및 능동형 센서의 융합센서를 이용한 SAA 알고리즘의 개발이 연구되었다. 네덜란드의 Delft 대학교에서는 EO, RADAR, TCAS, ADS-B를 융합한 알고리즘을 개발하였고 Flight-gear의 비행영상을 이용하여 Matlab SAA 알고리즘을 검증하였다[11]. 영국의 Cranfield 대학에서는 EO, IR, LIDAR, RADAR, TCAS, ADS-B센서를 융합하는 알고리즘을 개발하고 항공기에 탑재하여 검증하는 연구를 수행하였다[12].

본 연구는 충돌검출을 알고리즘을 부품수준에서 확장된 flight simulation 수준에서 수행할 수 있는 환경을 구축하는 연구를 지향한다. 통합 무인기 시뮬레이터 환경에서 충돌검출 알고리즘의 검증은 충돌검출 simulink 모델, 충돌검출 co-simulation 모델, 및 충돌검출 HILS 로 시험이 가능하며, 본

문에서는 simulink 모델의 시험결과를 제시한다.

## 2-3 TCAS(Traffic Alert and Collision Avoidance System)

공중충돌회피시스템 TCAS의 충돌 경보기능의 Simulink 모델을 Flightgear flight simulator 및 지상제어시스템과 연동하여 통합 무인기 시뮬레이션 환경에서 검증하였다. TCAS는 1981년에 FAA가 항공기 공중충돌방지시스템의 최소운용성능표준 (minimum operational performance standards, MOPS) TCAS 1에서 시작해서 현재는 TCAS IV까지 개발되었으며, ICAO는 기존 항공기에 2017년 까지 TCAS II 버전 7.1을 탑재할 것을 권고하고 있다. TCAS는 트랜스폰더를 이용해서 두 항공기 위치와 속도정보를 이용하여 최단접근지점 (closest point of approach, CPA)에 대한 도달시간을 계산하여 위험상황이 발생하면 위험 경보를 표시하는 Traffic Advisory (TA) 와 회피기동의 시작을 지시하는 Resolution Advisory (RA) 경보신호를 조종사에게 통보한다. RTCA DO-185에 정의되는 충돌회피기능은 다음 장에서 상세하게 설명하며, TCAS-II 규격은 참고문헌에 상세하게 설명되어 있다[13].

## III. 무인항공기 공중충돌회피 시스템 모델링

### 3-1 공중충돌회피 시스템 원리

이 장에서는 충돌회피 시스템의 Simulink 시뮬레이션 모델에 대하여 설명하고, 다음 장에서는 flight simulator, 지상제어 시스템, 항공기 역학 시뮬레이션과 개발된 충돌회피 시뮬레이션 모델의 연동에 대하여 설명한다. RTCA DO-185에 정의되는 충돌회피기능은 감지기능, 추적기능, 충돌위험판정기능, 그리고 HW:SW 고장에 대응하는 fail-safe 기능으로 구분 된다 [13]. 공중충돌 경보시스템은 접근하는 항공기 간에 대한 충돌위험을 판단하기 위해 거리시험과 고도시험을 수행한다.

거리 시험은 두 항공기의 최단접근지점 (closest point of approach, CPA)에 대한 도달시간과 미리 지정된 RA(resolution advisory) 와 TA(traffic advisory) 기준 값과 비교하여 충돌 위험을 판단한다. 이때 사용되는 기준 값은 항공기의 고도에 따라서 결정되는 민감도 (sensitivity level) 수준에 따라 지정된다[13]. 고도 (altitude) 시험은 두 항공기의 고도 차이를 설정된 기준 값과 비교하여 판단한다. 거리 시험과 고도 시험 결과 값을 비교하여 RA와 TA를 출력한다. 본 논문에서는 항공기와 침입기의 현재 속도, 방향을 이용하여 CPA 점을 예상한다. 그 때의 CPA에 도달하는 시간 (Time of CPA:  $\tau$ )과 두 항공기간의 고도 차이를 계산해서 TA, RA 발행여부를 결정한다.

### 3-2 CPA도달시간 및 고도계산

주어진 상황에서 TA와 RA를 판단하려면 두 비행기 간의

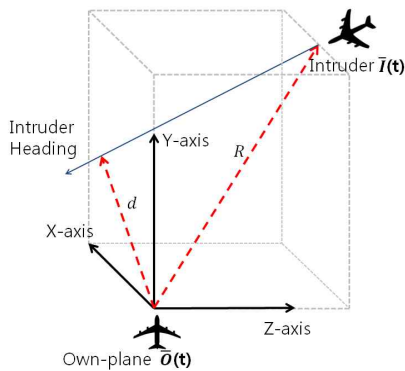


그림 1. Own-flight과 Intruder의 이동경로  
Fig. 1. Movement path of own-flight and Intruder.

CPA도달시간 (Time of CPA :  $\tau$ ) 과 고도차이가 필요하다. 그림 1에서 CPA도달시간을 구하기 위해서는 항공기 (own flight)의 초기위치  $\vec{O}_0$ 와 속도  $\vec{V}_0$ , 침입기 (intruder)의 초기 위치  $\vec{I}_0$ 와  $\vec{V}_I$ 라고 하면, 시간이  $t$ 일 때 항공기 위치  $\vec{O}_0(t)$ 와 침입기 위치  $\vec{I}_0(t)$ 는 다음과 같다.

$$\vec{O}_0(t) = \vec{O}_0 + \vec{V}_0 \times t \quad (1)$$

$$\vec{I}_0(t) = \vec{I}_0 + \vec{V}_I \times t \quad (2)$$

이때 두 항공기간 거리  $R$ 은 다음과 같다.

$$R = \vec{I}_0(t) - \vec{O}_0(t) \quad (3)$$

최소거리도달시간  $t_Q$ 는 거리  $R$ 의 제곱의 미분 값이 0이 되는 시점으로서 다음과 같이 정리할 수 있다 [14].

$$t_Q = - \frac{(\vec{O}_0 - \vec{I}_0)(\vec{V}_0 - \vec{V}_I)}{(\vec{V}_0 - \vec{V}_I)^2} \quad (4)$$

CPA 값이 0이라면, 두 항공기가 동일 위치에 존재한다고 할 수 있다. 결과적으로 CPA도달시간은 아래와 같다.

$$\tau = - \frac{(\vec{O}_0 - \vec{I}_0)(\vec{V}_0 - \vec{V}_I)}{(\vec{I}_0(t) - \vec{O}_0(t))^2} \quad (5)$$

### 3-3 공중충돌경보시스템 Simulink 모델

Matlab/Simulink를 이용해서 4개의 모듈로 구성된 공중충돌경보시스템 시뮬레이션 모델을 그림 2에 설명하였다. 첫 번째 모듈은 회피지시시간 및 고도계산을 위한 자료들의 형태를 변환하는 데이터 변환 모듈이다. 두 번째 모듈은 최소 거리도달시

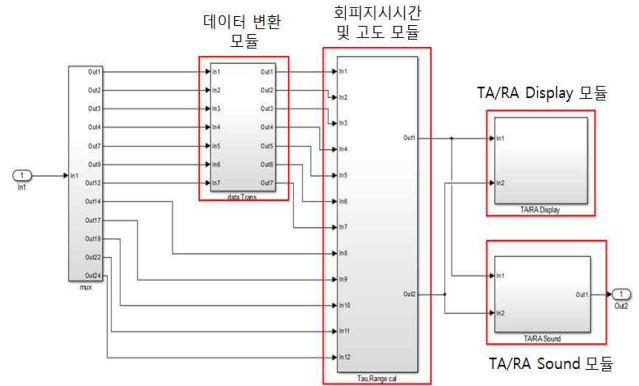


그림 2. 공중충돌경보시스템 Simulink 모델  
Fig. 2. Air collision alarm system Simulink model.

간 및 고도 계산 모듈이다. 세 번째 모듈은 TA와 RA 일 때의 지시등(indicator lamp)을 표시하는 모듈이다. 마지막 4번째 모듈은 TA와 RA일 때 음성경보를 출력해주는 모듈이다.

#### 1) 데이터 변환 모듈

데이터 변환 모듈은 Geodetic 데이터를 ECEF (earth-centered earth-fixed) 좌표계의 데이터로 변환한다. Geodetic 데이터는 위도 경도 고도 (longitude, latitude, altitude: LLA) 좌표계의 데이터로 구성된다. 데이터 변환 모듈의 구성은 그림3에서 확인할 수 있다. ECEF 좌표계 입력데이터는 회피지시시간 및 고도 연산 모듈에 전달된다. 데이터 변환 모듈은 M/S에 LLA to ECEF Position library를 이용한다.

#### 2) 최소거리도달시간 및 고도 계산 모듈

앞의 3.2절의 수식을 이용하여 계산 모듈을 구성하였다. 입력력은 항공기와 침입기의 위치와 속도이며, 출력은 회피지시시간과 고도에 따른 상태 값이다. 출력 값이 1일 때는 정상 상태, 2일 때는 TA, 3일 때는 RA의 출력을 의미한다.

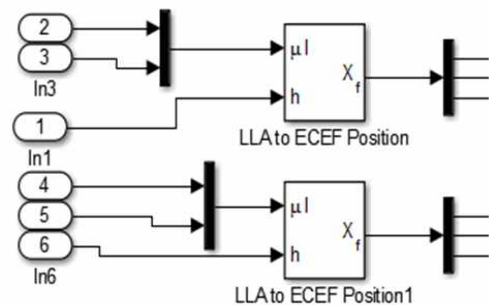


그림 3. 데이터 변환 Simulink 모듈  
Fig. 3. Data conversion Simulink module.



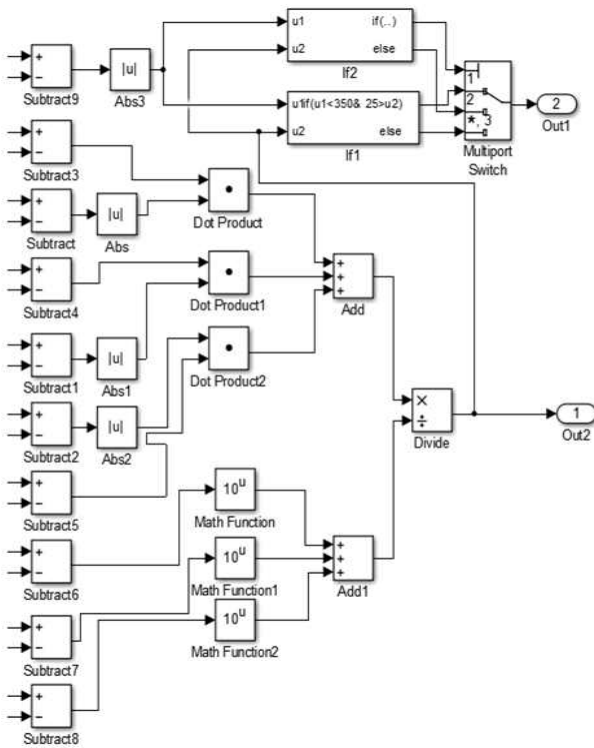


그림 4. 최소거리도달시간 및 고도 계산 Simulink 모듈  
 Fig. 4. Time to minimum distance and altitude calculations Simulink module.

3) Time to Minimum Distance and altitude calculations Simulink 모듈

최소거리도달시간 및 고도계산 모듈은 airframe dynamics에서 생성되는 기체의 상태 정보 회피지시시간 및 고도 데이터를 연산한다. 모듈의 구성은 그림 4에서 확인 할 수 있다. 모듈은 ECEF 데이터와 속도 데이터를 사용하여 회피지시시간과 고도 데이터를 산출한다.

4) Traffic Advisory/Resolution Advisory Display 모듈

TA/RA Display 모듈은 3.2절에서 정의한 작동원리에 따라서 비행기의 현재 상태를 출력하며, 구성은 그림 5로 확인할 수 있다. 안전상태일 때는 지시등 중 일반 등에 녹색등, TA상황일 때는 TA 등에 황색등, RA상황일 때는 RA 등에 적색등이 표시 된다.

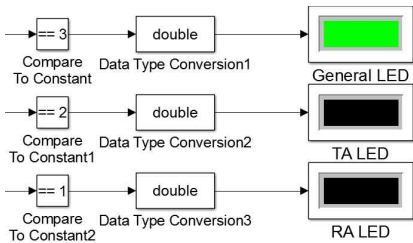


그림 5. TA/RA display simulink 모듈  
 Fig. 5. TA/RA display Simulink module.

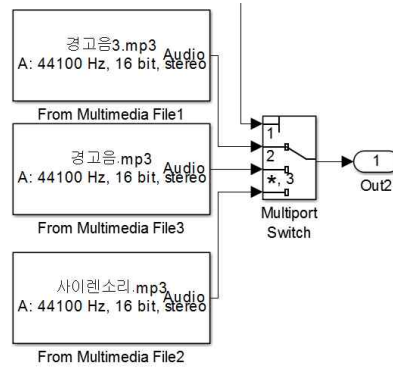


그림 6. TA/RA sound Simulink 모듈  
 Fig. 6. TA/RA sound Simulink module.

5) Traffic Advisory/Resolution Advisory Sound 모듈

TA/RA sound는 최소거리도달시간 및 고도 계산모듈의 출력에 따라서 지시 상태를 음성으로 출력한다. 예를 들면, 평시에는 항공기 소리를 출력하고 TA 상황일 경우에는 “Traffic, traffic” 소리를 출력 RA상황 일 경우에는 “Climb, climb” 소리를 출력한다. TA/RA sound의 구성은 그림 6로 확인할 수 있다.

IV. 공중충돌경보시스템시뮬레이션환경

4-1 무인기 시뮬레이션 시스템 개요

무인기 공중충돌경보 시뮬레이션의 시스템의 구성요소는 항공기 역학 시뮬레이터, 공중충돌회피시뮬레이션 모델, 및 flight simulator로 구성된다 (그림 7). 먼저 항공기 조종은 시뮬레이터 입력장치 (조이스틱)을 이용하거나 운항제정보가 기록된 파일을 이용한다. 항공기체, 충돌회피시스템 및 지상통제 시스템은 Matlab/Simulink (M/S)로 구현하였다. 먼저 기체 역학 시뮬레이션 모델은 M/S의 aerospace package의 airframe dynamics 모듈을 이용하였다. 기체에서 발생하는 좌표, 속도 등의 기체 상태정보는 공중충돌회피 시스템으로 입력되어 TA, RA 계산에 사용된다. 지상통제장치는 M/S의 그래픽 표시기능을 이용하여 구현하였다. 항공기와 침입기 두 대 및 운항 형상은 FlightGear flight simulator의 그래픽 기능을 이용하였다.

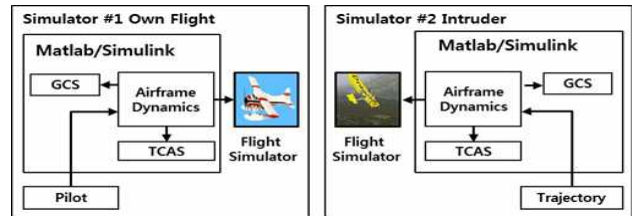


그림 7. 항공기와 침입기 공중충돌경보시스템 검증을 위한 통합 무인기 시뮬레이터  
 Fig. 7. Integrated UAV simulator for validation of air collision alarm system of airplane and intruder.

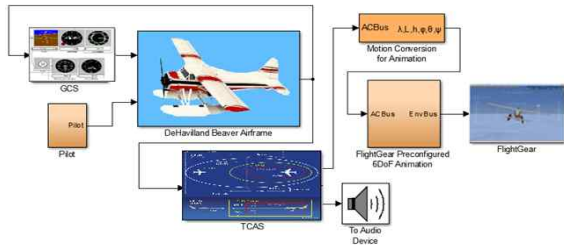


그림 8. 공중충돌경보시스템 검증을 위한 통합 무인기 시뮬레이터  
 Fig. 8. Integrated UAV simulator for validation of air collision alarm system.

4-2 무인기 시뮬레이션 시스템 구성요소

그림8은 FlightGear Matlab/Simulink 통합 무인기 시뮬레이터 시스템의 구성요소를 상세하게 설명한다. 통합 무인기 시뮬레이터에서 항공기는 조종사의 입력 또는 미리 설정된 trajectory 파일을 이용하여 운항한다. Simulink의 airframe dynamics는 항공기의 속도, 고도, LLA (longitude latitude altitude) 데이터를 TCAS 모듈과 FlightGear flight simulator, 그리고 지상통제장치의 항공계기 display 부로 출력한다. FlightGear는 항공기와 침입기의 형상 및 이동을 컴퓨터 그래픽으로 표시한다. TCAS 모듈은 airframe dynamics의 상태정보와 transponder를 통해서 입력되는 침입기의 속도, 고도, 및 LLA 데이터를 이용하여 TA, RA 조건을 판정한다. 지상통제장치는 Simulink를 이용하여 항공기 고도, 속도 상태정보를 표시한다.

1) Pilot

사용자는 조이스틱등과 같은 flight simulator의 일반적인 입력장치들을 이용해서 운용하는 조종 명령 데이터를 항공기에 전송한다. 조종기 이외에 지정된 항로 데이터를 활용하는 경우 trajectory file을 항공기 역학계에 입력하는 것도 가능하다.

2) Environment:

그림 9의 Environment는 대기 및 풍속모델과 M/S Aerospace Tool로 구성된다. Flight Gear 인터페이스에 필요한 LLA(longitude, latitude, altitude) 데이터의 생성을 위하여 M/S의 airframe dynamics 모델의 Geoid 모델을 이용하였다. geoid 모델의 LLA 좌표 값은 지구의 중심을 원점으로 하는 3차원 직교좌표계인 WGS84모델을 기반으로 중력 값을 적용한다.

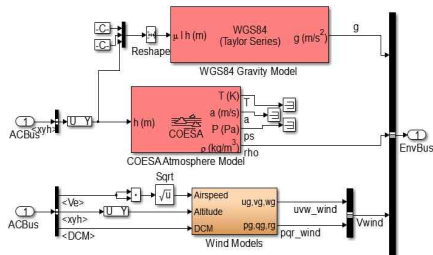


그림 9. Environment Simulink 모듈  
 Fig. 9. Environment Simulink module.

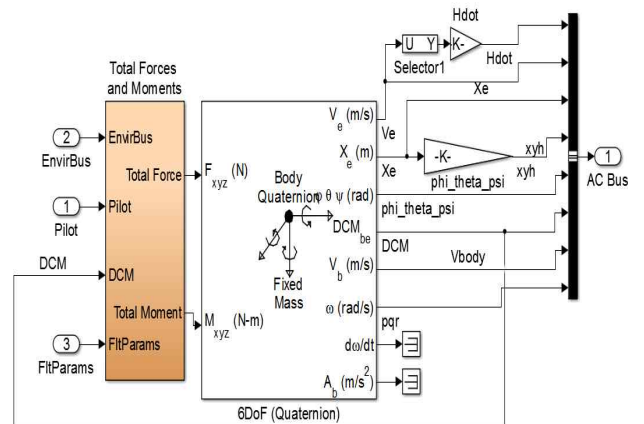


그림 10. Airframe Dynamics Simulink 모듈  
 Fig. 10. Airframe dynamics Simulink module.

3) Airframe Dynamics

그림 10과 같이 M/S airframe dynamics모델에서는 사용자의 조종 데이터와 airframe dynamics를 기반으로 생성되는 항공기의 운항 데이터를 계산한다. 또한, 내부에서는 PID 제어를 이용하여 roll, pitch, yaw, 그리고 속도벡터를 계산한다.

4) FlightGear Interface

FlightGear Interface모듈은 M/S에서 FlightGear로 데이터를 전송할 때 사용하는 모듈이다. 그림 11은 motion data를 FlightGear에서 필요한 data로 변경해주는 Motion Conversion모듈과 변경한 데이터를 FlightGear로 전송하는 6DoF Animation 모듈로 구성되어 있다.

5) 지상통제시스템

그림 12의 지상통제시스템(ground control system: GCS) 모듈에서는 항공기의 운항 상태를 실시간으로 확인할 수 있다. 각 계기의 역할을 보면 horizon은 비행기의 수평상태를 altimeter는 고도를 airspeed는 현재 속도를 heading indicator는 비행 방향을 마지막으로 climb rate는 비행기의 수직속도를 확인할 수 있다.

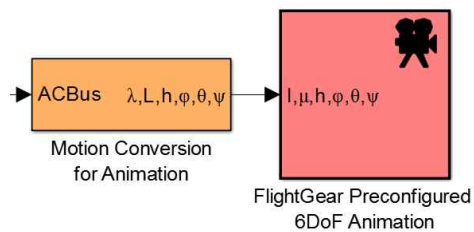


그림 11. FlightGear와 Simulink interface 모듈  
 Fig. 11. FlightGear와 Simulink interface module.

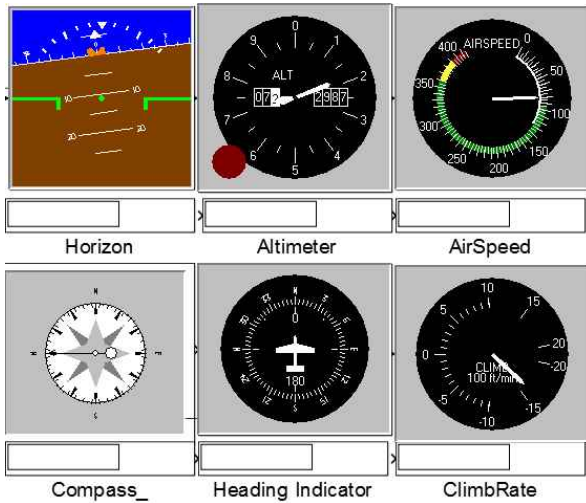


그림 12. 지상통제시스템 Simulink 모듈  
 Fig. 12. GCS Simulink module.

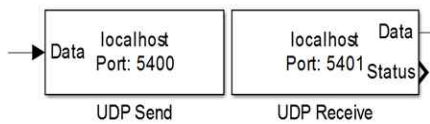


그림 13. UDP 통신 Simulink 모듈  
 Fig. 13. GCS Simulink module.

6) UDP 통신 모듈

충돌경보 시스템은 두 항공기가 좌표정보를 상호간에 교신하는 상황을 가정하였다. 이 기능은 Simulink의 그림 13의 UDP 통신 모듈을 이용하여 구현한다.

V. 무인항공기 공중충돌경보 시뮬레이션 검증

5-1 공중충돌경보시스템 검증을 위한 Encounter 모델

공중충돌경보시스템을 검증하기 위해 FAA William J. Hughes Technical Center (WJHTC)에서 생성한 Encounter 모델들을 사용하였다[15]. 그림 14는 시뮬레이션 검증에 사용한 Encounter 모델들이다. Class 0-9은 교차(crossing)하는 경우의 모델들이고 10-19는 교차하지 않는(non-crossing) 경우의 모델들이다. 5.2절에서는 MATLAB 단독으로, 5.3절에서는 통합 무인기 시뮬레이션을 이용하여 검증하였다.

5-2 Matlab을 이용한 공중충돌경보시스템 검증

공중충돌경보시스템 알고리즘을 검증하기 위해 Matlab 3D Plot 기능을 이용하여 encounter class 0/10 경우를 시험하였다. 적색은 Own flight, 청색은 intruder로 동일 고도에서 충돌 상황 테스트를 했다. 그림 15의 flight trajectory 그림은 정면으로 두 항공기가 접근하는 것을 가정한 모델이다.

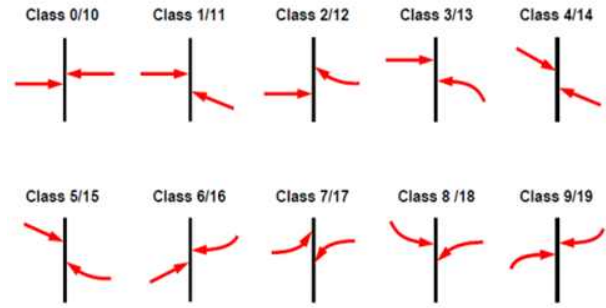


그림 14. Encounter 모델 (0-9 계획된 교차, 10-19 계획된 비-교차)  
 Fig. 14. Encounter model (0-9 planned crossing, 10-19 planned non-crossing).

그림 16을 보면 flight tau data인 회피지시시간이 지속적으로 줄어드는 것을 확인 할 수 있다. 마지막으로 그림 17을 보면 안전단계인 3, TA단계인 2, RA 단계인 1로 데이터가 변하는 것을 확인 할 수 있다.

그림 18은 항공기와 침입기의 고도차이가 2000 m인 안전한 경우를 가정한 모델로, 회피지시시간( $\tau$ )은 그림 19와 같이 감소하지만, 고도차이가 크기 때문에 안전한 상태인 3단계가 유지됨을 그림 20에서 확인 할 수 있다.

5-3 FlightGear 연동 시뮬레이션 테스트

Matlab Simulink와 FlightGear의 통합 무인기 시뮬레이터를 이용해서 encounter모델 class 3/13일 때 공중충돌경보시스템 알고리즘을 검증하였다. 공중충돌방지시스템의 결과는 TA, RA display모듈의 지시등(indicator lamp)로 확인한다. 그림21 상단은 원거리의 침입기가 검출되어 지시등이 초록색으로 안전상태를 표시하며, 중간그림은 침입기가 근접하여 TA 상황이 발생하여 노란색 지시등이 점등되고, 하단의 그림은 RA상황으로 빨강색 지시등이 표시되며 두 항공기가 충돌하는 장면을

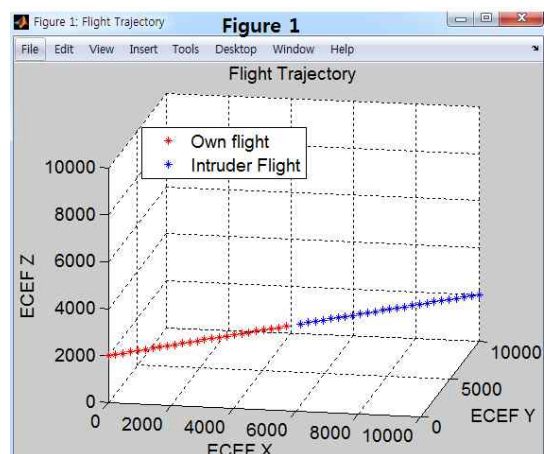


그림 15. 동일 고도 충돌 상황 test. 테스트 비행경로  
 Fig. 15. Flight path of equal altitude collision situation.

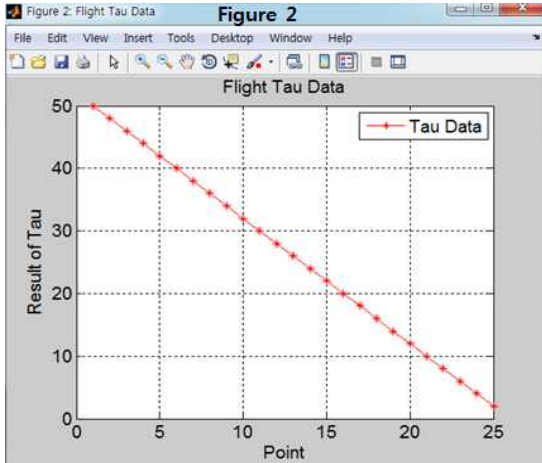


그림 16. 동일 고도 충돌 상황에서 회피지시시간의 변화  
 Fig. 16. Tau Changes of equal altitude collision situation.

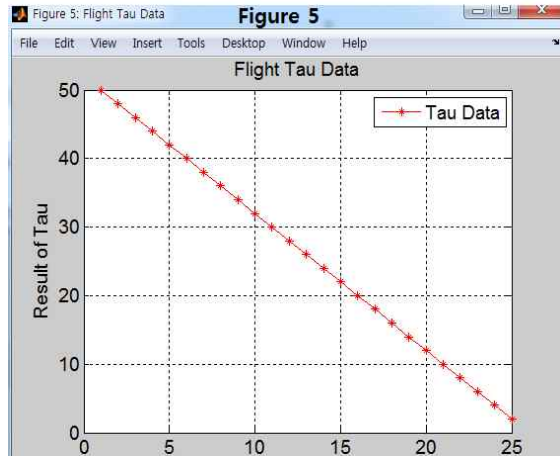


그림 19. 회피지시시간의 변화 (고도 2000 m)  
 Fig. 19. Tau changes (alt. 2000 m).

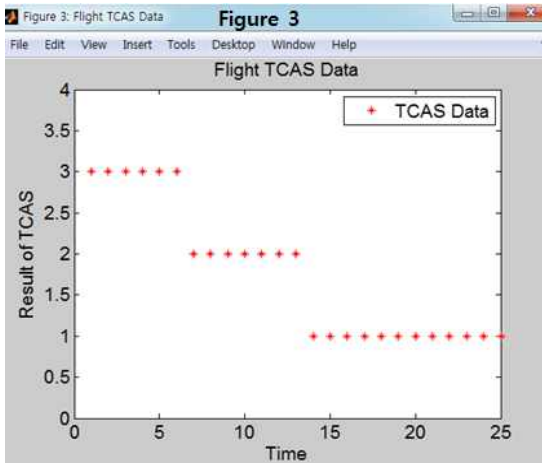


그림 17. 동일 고도 충돌 상황에서 시간에 따른 TCAS 결과  
 Fig. 17. TCAS result of equal altitude collision situation.

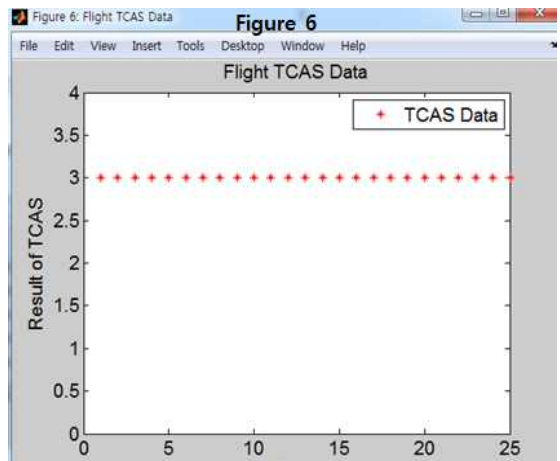


그림 20. TCAS 결과 (고도 2000 m)  
 Fig. 20. TCAS result (alt. 2000 m).

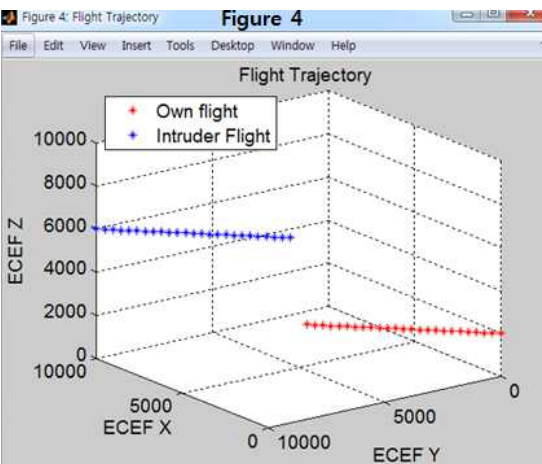


그림 18. 2000 m 고도 차 접근 상황 테스트 비행경로  
 Fig. 18. Flight path of 2000 m altitude collision situation test.

flight simulator 화면으로 확인할 수 있다. 그림 22는 두 항공기가 RA 상황이 발생한 후 회피기동을 적절하게 수행하여 충돌을 회피한 상황을 표시하며 최종적으로 초록색 지시등의 점등을 확인할 수 있다.

## VI. 결 론

본 논문에서는 무인기 공중 충돌경보 및 회피 알고리즘 개발 및 검증을 지원하기 위하여 Matlab/Simulink, Flight Gear 비행 시뮬레이터를 연동하는 통합 무인기 시뮬레이션 환경을 구축하였다. 통합 환경을 검증하기 위하여 충돌경보알고리즘을 Simulink 모델로 개발하고 통합 무인기 시뮬레이터에 연동한 뒤, 다양한 조우모델(encounter model)을 적용하여 시뮬레이션 시험을 수행하였다.



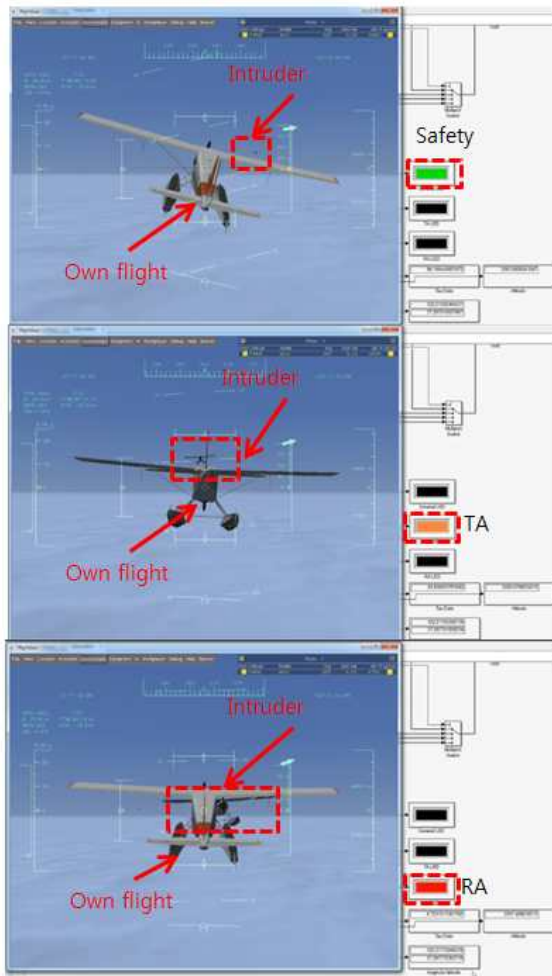


그림 21. FlightGear 연동 시뮬레이션 (RA 상황 후 충돌)  
 Fig. 21. FlightGear co-simulation (Collision after RA).

개발된 시뮬레이터는 기존의 TCAS/SAA 연구에서 수행했던 LRU 수준의 시뮬레이션을 확장하여 (1) 항공기 수준의 시뮬레이션을 지원하여 항전장비 인증 및 인증 무인기 개발 기간 및 비용의 절약이 가능하며, (2) 항적을 이용한 정적 시험 및 조종장치를 이용한 동적 시험을 수행할 수 있어서 test coverage가 확장 가능하며, (3) 개발사업 초기단계부터 조종사의 교육·훈련용으로 활용할 수 있으므로 조기 사업화가 가능하다. FAA 조우 모델을 기반으로 항적 자료를 생성하여 시뮬레이션 시험을 수행하고 충돌위험 발생상황을 확인하였다.

개발된 통합 무인기 시뮬레이터는 무인기 상용화의 핵심기술인 ADS-B와 EO/IR, LiDAR, 초음파 등의 센서를 통합하는 SAA 항공 임베디드 시스템의 검증에 활용이 가능하다.

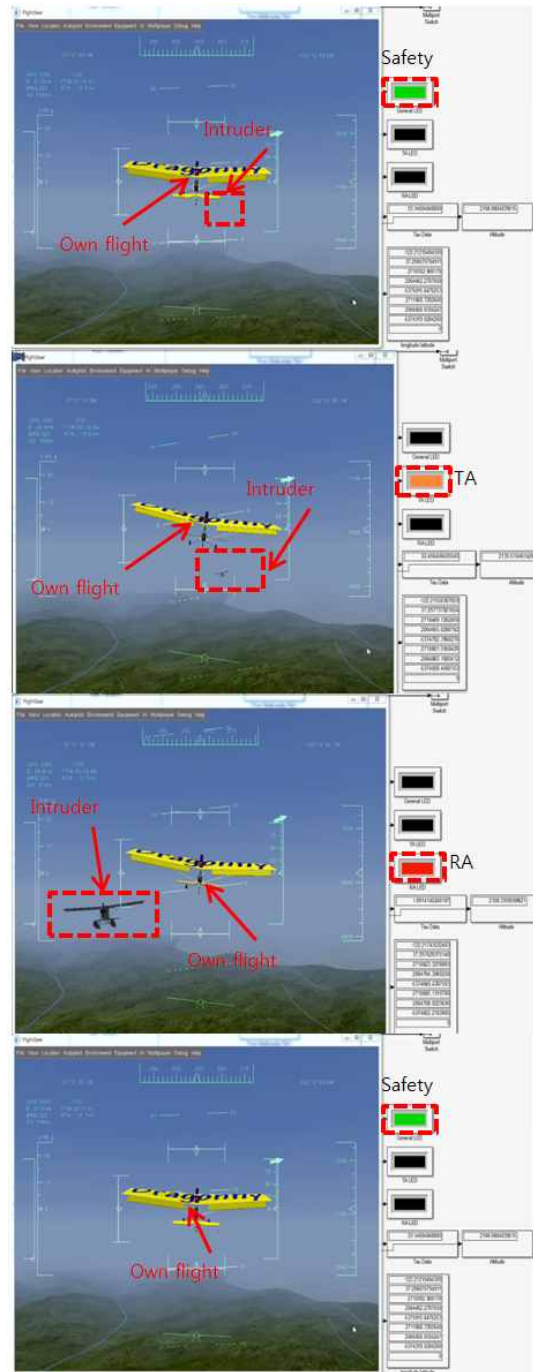


그림 22. FlightGear 연동 시뮬레이션 (RA 상황 후 회피)  
 Fig. 22. FlightGear co-simulation(avoidance after RA).

### 감사의 글

본 연구는 경기도의 경기도지역협력연구센터(GRRC) 사업의 일환으로 수행하였음. [GRRC항공2014-12-005, 고가용성 사물인터넷(IoT) 게이트웨이 개발]

**참고 문헌**

[1] Washington Post(2014, November). Near mid-air collisions with drones [Internet]. Available: <http://www.washingtonpost.com/wp-srv/special/national/faa-drones/>

[2] FAA. UAS accident and incident preliminary reports [Internet]. Available: <http://www.asias.faa.gov/pls/apex/f?p=100:446:0::NO:446>.

[3] B. K. Jun, Performance improvement of the collision avoidance algorithm for the TCAS-II system with safety critical software, M.S. dissertation, Korea Aerospace University, Gyeonggi-do, Korea, 2011.

[4] Y. R. Kim, A development of 3-D resolution algorithm for aircraft collision avoidance, M.S. dissertation, Gyeonggi-do, Korea Aerospace University, Korea, 2012.

[5] S. H. Oh, “Technical development trend of ground-based collision avoidance for UAVs”, *Aerospace Technology Trends*, Vol. 10, No.2 , pp. 26-35, Dec. 2012.

[6] R. K. Barnhart, S. B. Hoffman, D. M. Marshall, and E. Shappee, *Introduction to Unmanned Aircraft Systems*, 1st ed. London, UK: CRC Press, 2012.

[7] S. C. Degen, Reactive image-based collision avoidance system for unmanned aircraft systems, M.S. dissertation, University of Queensland, Australia, 2011.

[8] L. Forlenza, Vision based strategies for implementing sense and avoid capabilities onboard unmanned aerial systems, M.S. dissertation, University of Naples, Italy, 2011.

[9] H. J. Choi, Reactive collision avoidance of unmanned aerial vehicles using vision sensors, Ph.D. dissertation, Seoul National University, Korea, 2014.

[10] S. W. Cho, “An image processing algorithm for detection and tracking of aerial vehicles in short-range,” in *50th IEEE Conference on Decision and Control and European Control Conference*, Orlando: FL, pp. 7482-7487, 2011.

[11] G. Berghuis, Simulation of a detect and sense system, M.S. dissertation, Delft University of Technology, Netherlands, 2010.

[12] P. Rodriguez, R. Sabatini, A. Gardi, S. Ramasamy, “A novel system for non-cooperative UAV sense-and-avoid,” in *European Navigation Conference 2013*, Vienna: Austria, pp. 1-11, 2013.

[13] J. Collins, Airborne sense and alert collision warning and avoidance system, M.S. dissertation, University of Auckland, New Zealand, 2011.

[14] Federal aviation administration, Introduction to TCAS-II version 7.1, U.S. Department of Transportation, U.S., Advisory Circular HQ-111358, 2011.

[15] B. J. Chludzinski, Evaluation of TCAS II version 7.1 Using the FAA fast-time encounter generator model, MIT Lincoln Laboratory, MA, Project Report ATC-346, 2009.



**문 성 업 (Seong-Yeop Moon)**  
 2015년 2월 : 한국항공대학교 항공전자공학과 (공학학사)  
 2015년 ~ 현재 : 한국항공대학교 대학원 항공전자공학과 공학석사 과정  
 ※ 관심분야 : 무인항공기, 고신뢰성 시스템, 임베디드 시스템



**김 주 영 (Kil-Dong Hong)**  
 2014년 2월 : 한국항공대학교 항공전자공학과 (공학학사)  
 2014년 ~ 현재 : 한국항공대학교 대학원 항공전자공학과 공학석사 과정  
 ※ 관심분야 : 무인항공기, 고신뢰성 시스템, 임베디드시스템



**이 동 우 (Dong-Woo Lee)**  
 2014년 8월 : 한국항공대학교 항공전자공학과 (공학박사)  
 2014년 ~ 현재 : 한국대학교 항공전자연구소 연구원  
 ※ 관심분야 : 영상처리, 고신뢰성 시스템, Fault Tolerant Design



**백 경 민** (Gyeong-Min Baek)

2013년 8월 : 한국항공대학교 항공전자공학과 (공학학사)  
2013년 ~ 현재 : 한국항공대학교 대학원 항공전자공학과 공학석사 과정  
※ 관심분야 : 영상처리, 임베디드 시스템



**김 진 실** (Jin-Sil Kim)

2015년 2월 : 한국항공대학교 항공전자공학과 (공학학사)  
2015년 ~ 현재 : 한국항공대학교 대학원 항공전자공학과 공학석사 과정  
※ 관심분야 : 고장감내, 고신뢰성 시스템



**나 종 화** (Jong-Whoa Na)

1995년 2월 : 아리조나대학교(미) 컴퓨터공학과 (공학박사)  
2005년 ~ 현재 : 한국항공대학교 항공전자 및 정보통신공학부 교수  
※ 관심분야 : 컴퓨터 시스템, 영상처리, 고신뢰성 시스템