

Improvement of Altitude Measurement Algorithm Based on Accelerometer for Holding Drone's Altitude

Deok Yeop Kim[†] · Bo Ram Yun[†] · Sunghee Lee^{**} · Woo Jin Lee^{***}

ABSTRACT

Drones require altitude holding in order to achieve flight objectives. The altitude holding of the drone is to repeat the operation of raising or lowering the drone according to the altitude information being measured in real-time. When the drones are maintained altitude, the drone's altitude will continue to change due to external factors such as imbalance in thrust due to difference in motor speed or wind. Therefore, in order to maintain the altitude of drone, we have to exactly measure the continuously changing altitude of the drone. Generally, the acceleration sensor is used for measuring the height of the drones. In this method, there is a problem that the measured value due to the integration error accumulates, and the drone's vibration is recognized by the altitude change. To solve the difficulty of the altitude measurement, commercial drones and existing studies are used for altitude measurement together with acceleration sensors by adding other sensors. However, most of the additional sensors have a limitation on the measurement distance and when the sensors are used together, the calculation processing of the sensor values increases and the altitude measurement speed is delayed. Therefore, it is necessary to accurately measure the altitude of the drone without considering additional sensors or devices. In this paper, we propose a measurement algorithm that improves general altitude measurement method using acceleration sensor and show that accuracy of altitude holding and altitude measurement is improved as a result of applying this algorithm.

Keywords : Drone, Altitude Measurement, Altitude Holding, Accelerometer, IMU

드론의 고도 유지를 위한 가속도센서 기반 고도 측정 알고리즘 개선

김 덕 엽[†] · 윤 보 람[†] · 이 성 희^{**} · 이 우 진^{***}

요 약

드론은 비행 목적을 달성하기 위해 고도 유지를 필요로 하는 경우가 많다. 일반적으로 드론의 고도 유지 기능은 현재 측정되는 고도 정보에 따라 드론을 상승시키거나 하강시키는 작업을 반복하는 것을 의미한다. 고도 유지 중에 모터 속도 차이로 인한 추력의 불균형이나 바람 등의 외적 요인으로 인해 드론의 고도가 계속 변한다. 그럼에도 불구하고 고도를 유지하기 위해서는 기본적으로 계속해서 변하는 드론의 고도를 정확하게 측정해야 한다. 드론의 고도 측정 방법은 일반적으로 가속도센서를 사용한다. 이 방법은 적분 오차 누적으로 인한 측정값이 발산하는 문제와 드론의 기체 진동조차 고도 변화로 인지하는 문제가 존재한다. 그래서 상용 드론이나 기존 연구에서는 가속도센서를 제외한 별도의 센서를 추가하여 고도 측정에 사용한다. 그러나 추가하는 센서 대부분은 측정거리가 제한이 있으며 여러 센서들을 같이 사용하는 경우 센서 값들의 연산 처리가 많아져 고도 측정 속도가 지연될 우려가 있다. 따라서 드론의 고도 유지, 고도 측정 성능에 영향을 주지 않으면서 정확한 고도를 측정할 수 있는 방안이 필요하다. 본 논문에서는 가속도센서를 이용하는 일반적인 고도 측정 방법을 개선한 측정 알고리즘을 제안하고 본 알고리즘을 적용한 결과로 고도 유지와 고도 측정의 정확성이 향상됨을 보인다.

키워드 : 드론, 고도 측정, 고도 유지, 가속도센서, 관성항법장치

1. 서 론

최근 드론 산업이 급속히 발전하여 드론이 다양한 서비스

들을 제공할 수 있게 되면서 그 활용 범위가 확대되고 있다 [1-2]. 이러한 서비스를 충족시키기 위해서 대부분의 드론 비행은 고도 유지 기능을 필요로 한다. 예를 들어 촬영이나 감시 목적의 비행을 할 때 드론이 일정한 고도를 유지하지 못하고 움직이면 카메라 제어와 비행 제어를 동시에 해야 하기 때문에 비행 목적을 달성하기 어렵다.

고도 유지는 드론이 유지해야 하는 고도 정보를 기준으로 Fig. 1과 같이 수행된다. 현재 드론의 고도가 유지해야 하는 고도보다 낮으면 스로틀 수치를 올려 드론을 상승시키고 유지해야

※ 이 논문은 중소기업청에서 지원하는 2016년도 산학연 협력 기술개발사업 (No. C0442107)에 의하여 연구되었음.

† 준 회 원 : 경북대학교 컴퓨터학부 석사과정

** 준 회 원 : 경북대학교 컴퓨터학부 박사과정

*** 정 회 원 : 경북대학교 컴퓨터학부/소프트웨어기술연구소 교수

Manuscript Received : July 17, 2017

Accepted : August 3, 2017

* Corresponding Author : Woo Jin Lee(woojin@knu.ac.kr)

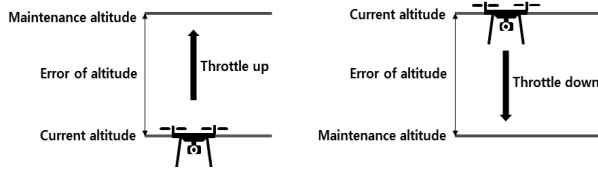


Fig. 1. Altitude Holding of the Drone

하는 고도보다 높으면 스로틀 수치를 내려 드론을 하강시킨다.

그러나 드론이 고도 유지 중에 몇 가지 원인으로 인하여 유지해야 하는 고도로부터 멀어지는 경우가 있다. 그 원인에는 모터 속도 차이로 인한 추력의 불균형, 바람, 드론의 상승이나 하강을 위한 스로틀 계산 값의 오차가 있다[3]. 바람은 환경적 요인이고 모터 속도 문제는 주로 하드웨어적 결함으로 인해 발생하는 것이므로 드론의 고도 유지 능력을 저해하는 주된 원인은 스로틀 계산 값 오차다. 이 오차는 드론을 현재 고도에서 유지해야 하는 고도로 움직이려 할 때 현재 고도가 정확하게 측정되지 않아서 발생한다. 따라서 안정적인 고도 유지를 수행하기 위해서는 현재 드론의 정확한 고도를 측정할 수 있어야 한다.

일반적으로 드론의 고도 측정은 관성항법장치에 내장된 가속도센서를 이용한다. 이 경우 가속도를 적분하여 고도를 계산하기 때문에 적분 오차가 누적된다. 또 드론 기체의 진동으로 인한 가속도 측정치의 변동 값도 고도 계산에 사용해서 측정 고도에 오차가 생긴다. 그래서 가속도센서의 이런 문제를 방지하고자 고도 측정에 기압계를 같이 사용한다. 하지만 기압계는 기압의 변화를 측정해 고도를 계산하는데 바람이나 온도 등 외적인 요인에 의해 민감하게 반응한다[4]. 예를 들어 드론을 작동시킬 때 프로펠러가 일으키는 바람이 지표면에 닿아 드론 주위의 기압이 낮아지면 고도가 0이 아닌 음의 값으로 측정된다. 결과적으로 가속도센서와 기압계를 같이 사용하더라도 측정값은 불규칙적으로 변하는 등 정확한 고도 측정을 기대하기 힘들다.

앞서 설명한 일반적인 드론의 고도 측정 방법의 문제점들 때문에 기존 연구들에서는 별도의 센서를 추가하여 다른 센서와 함께 고도 측정에 사용한다. 그 결과 기체가 무거워지고 제작비용이 증가하며 연산 처리가 많아지는 문제가 발생한다. 또 센서에 따라 측정 가능 거리의 한계가 있기 때문에 고도에 따라 정확도의 차이가 생긴다. 따라서 다른 센서를 추가하지 않고 고도 측정의 정확도를 높일 수 있는 방안이 필요하다.

본 논문에서는 일반적인 드론의 고도 측정 방법에서 발생하는 적분 오차, 진동으로 인한 측정 오차, 프로펠러의 바람 등의 문제들을 개선한 고도 측정 알고리즘을 제안한다. 또 제안한 알고리즘을 적용하여 고도 측정의 정확성과 고도 유지 능력이 향상됨을 보인다.

2. 관련 연구

2.1 가속도센서 기반의 거리 측정 기법

가속도센서를 이용한 이동거리 측정 기법[5]에서는 가속

도 크기에 따라 인자값과 임계값을 적용하고 사람의 보폭을 고려하여 이동거리를 측정한다. 이 시스템의 오차율은 8%로 기성품이 가진 오차율 13%보다 적어 비행자 이동거리 측정 시스템이 개선되었음을 보인다. 하지만 이 기법은 수평에 대한 거리 측정을 하는 것으로 수직 방향에 대한 거리 측정과는 기준 요소가 다르며 사람의 보폭을 고려한다는 점에서 드론의 고도측정에 적용하기에는 어려움이 있다.

관성항법장치, 카메라 및 라이다 센서를 융합하여 드론의 3차원 위치를 측정하는 기법[6]에서는 확장 칼만 필터를 이용한다. 관성항법장치와 카메라로부터 얻은 데이터를 칼만 필터로 융합하여 드론의 위치 및 거리를 측정한다. 이때 카메라의 느린 업데이트 속도 때문에 발생하는 측정 오차를 줄이기 위해 라이다 센서를 이용한다. 관성항법장치, 카메라 및 라이다 센서를 사용하여 드론 위치 측정의 정확성을 높였지만 시뮬레이션으로 실험한 것으로 실제 비행 시 발생할 수 있는 급격한 움직임, 드론 기체의 진동 등을 반영하지 않는다. 또한 라이다 센서는 측정거리 제한이 있어 높은 고도에서의 측정은 불가능하다.

가속도센서를 사용한 기존 연구들을 살펴보면 가속도센서로부터 발생하는 문제들을 개선하기 위해 사용 대상이나 환경을 적극적으로 이용한다. 사용 환경에서 파악할 수 있는 추가 정보들로 여러 상태를 판단하거나 오차를 줄이는데 이용한다. 드론 환경에서도 현재 드론의 상태 판단 및 비행제어를 위해 수많은 정보들을 관리한다. 따라서 드론 시스템에서 사용하는 정보들을 이용하여 일반적인 고도 측정 알고리즘에서 발생하는 문제점들의 개선을 할 수 있다.

2.2 기존의 고도 유지 기법

기압계와 소나센서를 이용한 기법[7]은 고도 구간별로 측정 센서 변경을 통해 드론의 고도 유지를 수행한다. 여기서 기압계, 소나, 위성항법장치 각각으로 측정된 고도 값에 대한 비교를 보여준다. 기압계만 사용했을 경우 측정 범위 한계는 거의 없지만 노이즈가 심하고 소나센서를 사용한 경우에는 측정 가능 거리가 약 7미터 정도로 명확하게 나타난다. 위성항법장치를 통한 고도 측정 결과는 노이즈도 크고 값이 불규칙적으로 나타난다. 고도 유지에는 GPS를 제외한 기압계와 소나센서만 사용한다. 7미터 이내에서는 소나센서, 그 외에는 기압계로 고도를 측정한다. 초음파와 기압계를 이용한 고도 유지 기법[8]도 초음파 센서를 측정 가능한 범위에서만 쓰고 그 외에는 기압계를 이용한다. 결국 별도의 센서를 추가하더라도 해당 센서의 측정 가능 범위에서만 고도의 정확도가 높은 것이다. 그래서 정확도는 낮지만 측정거리 제한이 없는 기압계나 가속도센서와 측정거리 제한은 있지만 정확도는 높은 센서를 같이 사용한다.

소나센서와 가속도센서의 융합을 통한 고도 유지 기법[9]은 측정거리 제한 때문에 측정 가능한 범위에서만 센서를 사용하는 방법에서 벗어나 센서를 융합하여 사용하는 방법이다. 소나센서는 측정가능 범위에서 높은 정확도를 보여주지만 드론에 장착하여 테스트했을 때 드론 진동으로 인한 측정 오차가 크게 발생한다. 이 문제를 해결하고 측정 정확

도를 높이기 위해 칼만 필터를 이용하여 두 센서 신호 데이터를 처리하고 융합한다. 그러나 융합을 통한 고도 측정 결과는 소나센서만 사용했을 때보다는 정확도가 높아졌지만 측정 오차가 크게 개선되지 않았다.

드론의 고도 유지에 필요한 핵심 정보는 유지해야 하는 고도와 현재 드론의 고도, 보정 스로틀 값이다. 보정 스로틀 값은 유지해야 하는 고도와 현재 측정 고도간의 오차로부터 계산하며 일반적으로 PID 제어 방법을 이용한다. 실제로 울트라소닉 센서를 사용한 기법[10]과 레이저 레이더를 사용한 기법[11]에서도 고도 측정에 사용하는 센서는 서로 다르지만 고도 유지를 위한 스로틀 계산에는 공통적으로 PID 제어 방법을 사용한다. 본 논문에서도 고도 유지를 위한 보정 스로틀 계산에 PID 제어 방법을 사용한다.

3. 드론의 고도 유지를 위한 고도 측정 알고리즘 개선

3.1 고도 유지 알고리즘

드론의 고도 유지 알고리즘은 크게 세 개의 과정을 거친다. 첫 번째는 가속도센서 또는 다른 센서를 이용해 드론의 현재 고도를 측정하는 과정이며 두 번째는 목표인 유지 고도를 달성하기 위한 보정 스로틀 값을 계산하는 과정이다. 마지막 세 번째 과정은 계산한 보정 스로틀 값을 현재 스로틀 값에 더해주는 과정이다. Fig. 2는 드론의 고도 유지 알고리즘 전체 과정을 나타낸 것이다.

먼저 가속도센서나 다른 센서를 이용해 드론의 현재 고도를 측정하는 과정은 고도 유지의 수행 여부와 상관없이 현재 고도 파악을 위해 반복한다. 고도의 측정은 사용하는 센서와 고도 계산에 여러 센서들을 융합하여 사용하는지 독립적으로 사용하는지에 따라 다르게 수행된다.

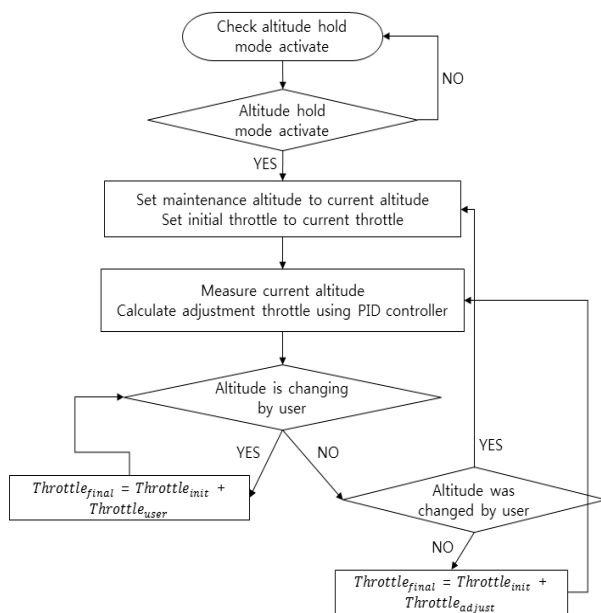


Fig. 2. Altitude Holding Algorithm of Drones

목표인 유지 고도를 달성하기 위한 보정 스로틀 값을 계산하는 과정은 계산을 위해서 PID 제어 방법을 사용한다. 제어 방법과 입력 값들은 알고리즘에 따라 조금씩 다를 수 있다. 일반적인 방법의 경우 입력 값으로 속도, 현재 가속도, 직전 가속도, 유지 고도, 현재 측정 고도를 사용한다. 먼저 유지 고도와 측정 고도간의 오차를 통해 고도에 대한 P 제어를 수행하여 유지 고도 달성에 필요한 속도를 계산한다.

그 다음 계산한 속도와 현재 속도간의 오차로 속도에 대한 PI 제어를 수행하고 현재 가속도와 직전 가속도로 속도에 대한 D 제어를 수행한다. 이렇게 고도에 대한 P 제어와 속도에 대한 PID 제어가 이루어져 최종적으로 보정 스로틀 값이 계산된다.

세 번째는 실제 드론의 고도 유지를 위해 계산한 보정 스로틀 값을 현재 스로틀 값에 더해주거나 사용자가 유지 고도를 변경하고자 할 때 유지 고도 정보를 변경해주는 부분이다. 고도 유지를 수행할 때 모터 값 계산을 위해 사용되는 최종 스로틀 값은 Equation (1)과 같이 계산된다. 최종 스로틀 값인 $Throttle_{final}$ 은 고도 유지를 시작할 때의 스로틀 값인 $Throttle_{init}$ 에 보정 스로틀 값인 $Throttle_{adjust}$ 을 더해준 것이다.

$$Throttle_{final} = Throttle_{init} + Throttle_{adjust} \quad (1)$$

사용자가 드론을 상승시키거나 하강시켜서 유지하고자 하는 고도를 바꾸는 중에는 최종 스로틀 값에 보정 스로틀 값을 더해주지 않고 사용자의 입력 스로틀의 일부를 더해준다. 그리고 유지 고도를 상승이나 하강이 끝나는 시점의 고도로 변경한다. 결과적으로 드론의 고도 유지는 유지 고도와 현재 고도간의 오차로 PID 제어를 수행하여 보정 스로틀 값을 구하는 것이 핵심이기 때문에 현재 고도를 정확하게 측정하는 것이 중요함을 알 수 있다.

3.2 일반적인 드론 고도 측정 방법

일반적인 드론의 고도 측정은 측정된 가속도에서 추출한 선형가속도에 추가 필터링 과정을 거친다. 그리고 기압계로 측정된 고도 정보를 함께 이용하여 최종적인 측정 고도를 계산한다. Fig. 3은 일반적인 드론의 고도 측정 과정을 나타낸 것이다.

가속도센서로부터 측정된 값은 중력가속도와 선형가속도를 포함한다. 고도를 계산하기 위해서 중력가속도를 제거하고 선형가속도 값만 추출한다. 그 다음 Equation (2)과 같은 저역 통과 필터를 이용하여 높은 선형가속도 성분의 값을 제거하여 평균치에 근접한 선형가속도의 값을 추출한다. 식에서 Acc_{filter} 는 필터링된 선형가속도, $Acc_{filter-1}$ 는 이전의 필터링된 선형가속도, α 는 가속도 값 측정 주기를 고려한 가중치이며 acc 는 중력가속도를 제거한 가속도의 값을 나타낸다. 그리고 Equation (2)를 통해 얻은 선형가속도에 포함된 드리프트를 제거하기 위해 데드밴드를 적용시킨다.

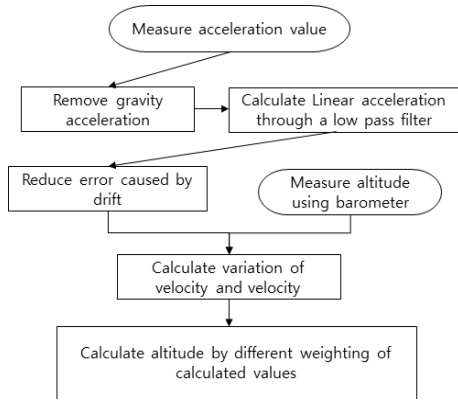


Fig. 3. General Altitude Measurement Process for Drones

$$Acc_{filter} = Acc_{filter-1} + \alpha (acc - Acc_{filter-1}) \quad (2)$$

드리프트를 제거한 후 현재 기압계로 측정된 정보들을 이용해 기압계 고도를 계산한다. 기압계를 이용하여 계산된 고도 값은 가속도센서에서 측정된 가속도 값과 함께 가중치를 다르게 하여 속도를 계산하는데 사용한다. 또 가속도센서로부터 구한 고도 값과 기압계를 이용하여 측정된 고도 값을 가중치를 다르게 하여 고도를 계산한다. 속도와 고도 계산을 할 때 가속도와 기압계를 함께 사용하기 때문에 둘 중 하나만 오차가 발생하여도 고도 측정값에 오차가 생긴다.

3.3 가속도센서를 이용한 드론 고도 측정 알고리즘

일반적인 드론의 고도 측정 방법을 참고하여 가속도 값에 중력가속도 제거, 저역 통과 필터를 통한 선형 가속도 추출 및 드리프트로 생기는 오차를 필터링한다. 그리고 평균 가속도 계산 과정에 고도 유지 인지 알고리즘을 이용한 개선 작업을 수행한다. Fig. 4는 개선한 평균 가속도 계산 과정을 나타낸다.

고도 유지 인지 알고리즘은 먼저 연속된 선형 가속도 값을 이용하여 드론의 고도 변화 상태를 확인한다. 만약 현재

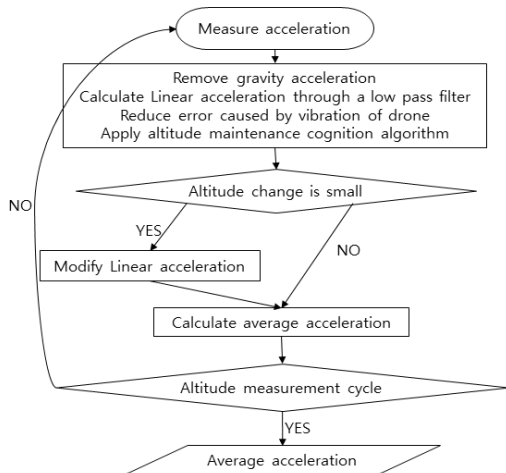


Fig. 4. Calculation Process of Average Acceleration

드론의 고도 변화가 적어 고도가 어느정도 유지되는 상태라면 가중치를 고려한 선형가속도를 보정하여 평균 가속도를 계산한다. 이때 고도 측정 주기와 일치하면 평균 가속도를 추출하고 일치하지 않으면 평균 가속도 계산 과정을 다시 수행한다. 이 과정을 거쳐 나온 출력 값인 평균 가속도는 고도 계산에서 이용된다.

Fig. 5는 가속도센서 적분 오차 누적 문제를 해결하고 스로틀 값과 윈도우를 적용하여 개선한 고도 측정 과정을 나타낸다.

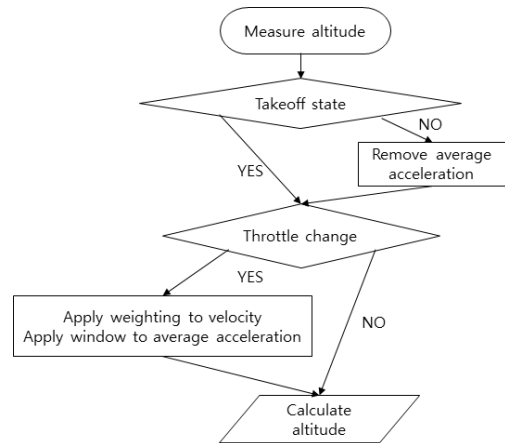


Fig. 5. Altitude Measurement Process

드론이 상승하기 전까지는 고도 값이 0으로 계산되어야 하지만 드론의 진동으로 인해 가속도가 측정되어 고도 값이 변한다. 이 문제를 해결하기 위해 사용자가 드론에 주는 신호인 스로틀 값을 이용하여 초기 고도 상승 상태를 판단한다. 현재 스로틀 값이 드론 이륙에 필요한 최저 기준 값보다 낮으면 이륙하지 않은 상태라고 판단하여 평균 가속도를 고도 계산에 사용하지 않는다. 현재 스로틀 값이 기준 값보다 커서 드론이 이륙한 상태라고 판단되면 평균 가속도를 적용하여 고도를 계산한다. 다음으로 평균 가속도 계산 과정에서 판단된 고도 변화 상태를 이용하여 고도 계산에 적용되는 속도를 달리한다. 고도 변화가 적은 상태면 속도 적분 오차 누적의 문제를 방지하기 위해 속도에 가중치를 고려해준다. 또한 고도 변화가 적은 상태라도 드론의 진동으로 인해 비정상적인 평균 가속도가 발생하므로 평균 가속도의 범위를 제한하기 위해 윈도우를 적용한다.

최종적으로 고도 계산은 드론 비행 시 발생하는 여러 가지를 고려하여 나온 평균 가속도를 Equation (3)에 적용하여 수행한다. 식에서 Alt는 고도 값, a는 가속도, t는 가속도 센서로부터 가속도 값을 측정된 시간을 나타낸다.

$$Alt = \frac{a}{2} + t^2 \quad (3)$$

설명한 평균 가속도 계산 과정과 고도 측정 과정에서의 개선을 통해 드론 비행 시 가속도 측정 및 고도 측정에서 발생하는 여러 문제들이 해결된다. 또 가속도 값 적분 오차 누적의 문제까지 보정되어 고도 측정의 오차율이 줄어든다.

4. 고도 측정 및 고도 유지 실험결과

실험에서 사용한 드론은 연구를 위해 자체 제작한 쿼드콥터이고 테스트는 실내 비행장에서 수행되었다. 비행장은 Fig. 6처럼 높이 200cm의 펜스가 둘러싸고 있으며 이 펜스를 고도 유지 테스트에 기준으로 이용하였다. 고도 측정에 영향을 끼치는 요인을 최소화하기 위해 장애물은 배치하지 않고 제자리에서 호버링 비행을 수행하였다.

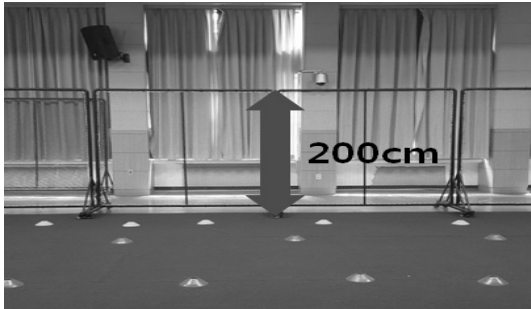


Fig. 6. Indoor Airfield

먼저 고도 측정 변화를 확인하기 위해 조이스틱을 이용하여 드론이 비슷한 고도를 유지하도록 조종하였다. Fig. 7은 기존 방법과 제안한 방법을 적용한 드론의 호버링 비행 중 고도 변화 그래프이다.

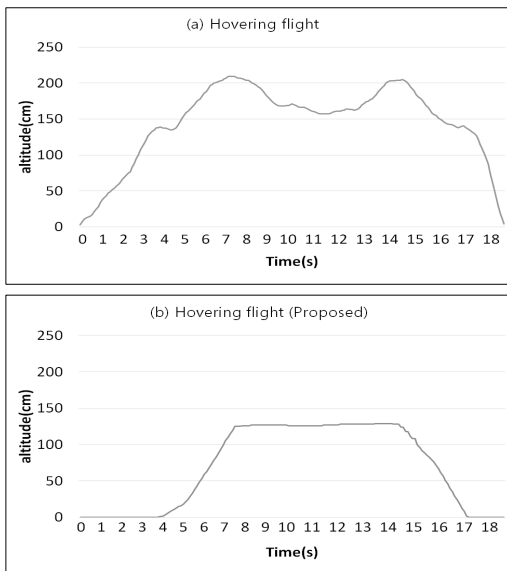


Fig. 7. Altitude Graph During Hovering Flight

Fig. 7(a)의 그래프는 기존 고도 측정 방법을 사용하여 약 130cm에서 호버링 비행을 한 실험에서의 고도 변화를 나타낸다. 실제 드론의 비행 고도와 다르게 측정된 고도 값은 157~209cm로 분포되어 나타났다. 실제 고도 값과 측정된 고도 값의 오차율은 약 40~60%로 고도 측정이 정확히 이루어지지 않았다. Fig. 7(b) 그래프는 제안한 고도 측정 방법을 사용하여 약 120cm에서 호버링 비행을 한 실험에서의 고도 변화를

나타낸다. 측정된 고도 값은 125~129cm로 분포되어 실제 드론의 비행 고도와 비슷하게 나타났다. 실제 고도 값과 측정된 고도 값의 오차율은 약 7%로 나타났다.

그 다음 두 방법을 적용하여 고도 유지 능력을 비교하기 위해 드론을 이륙시킨 후 약 200cm 높이에서 고도 유지를 수행하였다. Fig. 8은 고도 유지 수행 결과 그래프이다.

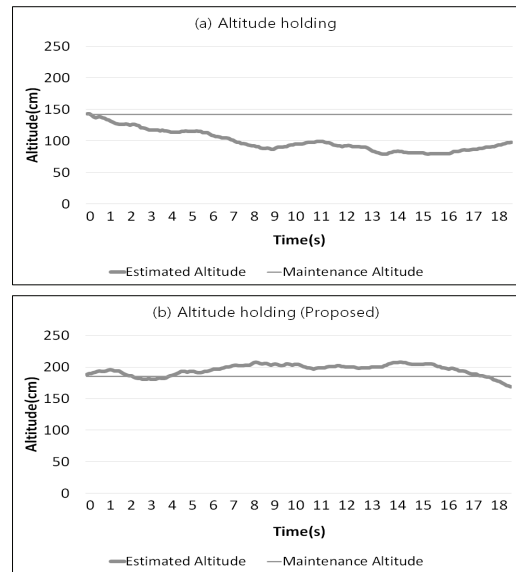


Fig. 8. Altitude Graph During Altitude Holding

Fig. 8(a) 그래프는 기존 측정 방법을 사용한 고도 유지 실험에서의 고도 변화를 나타낸다. 실제 드론 고도와 다르게 유지 고도가 142cm로 나타났다. 또 상승과 하강을 반복하면서 측정된 고도는 80cm~120cm 범위에 분포하였다. Fig. 8(b) 그래프는 제안한 측정 방법을 사용한 고도 유지 실험에서의 고도 변화를 나타낸다. 유지 고도 값은 185cm로 실제 드론의 고도와 비슷하게 나타났다. 또 상승과 하강을 반복하면서 측정된 고도는 180~205cm 범위에 분포하였다.

Table 1은 두 방법을 적용한 고도유지 실험에서 드론이 유지 고도로부터 움직인 오차 범위와 보정 스로틀 값의 최대, 최소, 평균을 나타낸 표이다. 실험은 두 방법 각각에 대하여 20번씩 수행되었다. 실험결과 유지 고도로부터의 오차 범위가 약 33% 줄어들었다. 또 유지 고도와 현재 고도간의 오차가 적어져 보정 스로틀 값도 약 15% 작게 나타났다.

Table 1. Comparison Table of Altitude Holding Results

		Altitude holding	Altitude holding (Proposed)
Gap from Maintenance Altitude (cm)	Max	45.27	33.71
	Min	15.66	11.99
	Avg	30.14	20.22
Adjustment Throttle (Signal Strength)	Max	96	71
	Min	-123	-96
	Avg	17.52	14.90

5. 결 론

드론의 고도 유지는 보정 스로틀 값 계산을 위해 정확한 고도 측정이 필요하다. 본 논문에서는 고도 측정의 정확도를 높이고자 일반적인 드론의 고도 측정 방법에서 나타나는 여러 문제점들을 개선한 가속도센서 기반의 고도 측정 알고리즘을 제안하였다. 실험결과 제안한 방법으로 측정한 드론의 고도 오차율은 약 10%로 기존의 방법보다 정확한 고도 측정이 가능함을 보였다. 그리고 고도 유지 실험결과 측정 고도가 더 정확해지면서 기존 방법에 비해 유지 고도가 실제 드론의 고도와 비슷하게 나타났다. 또 드론의 상승과 하강이 더 좁은 범위 안에서 이루어져 더 안정적인 고도 유지가 가능하였다.

References

- [1] S. I. Oh, "A Case Study Civilian Drone," in *Proceedings of the KOSBE Summer Conference*, pp.315-318, 2015.
- [2] P. J. Kim and H. J. Kim, "Design of an Image-Based Quadrotor Controller for Automatic Landing on Shipboard," in *Proceedings of the KSAS Conference*, pp.442-446, 2013.
- [3] Y. S. Lee, E. Kim, and J. M. Kim, "A Study on the Drones which the Flight Covers Indoor using by a Beacon," in *Proceedings of the IEEK Summer Conference*, pp.1624-1626, 2015.
- [4] H. Yu, S. B. Cheon, Y. J. Lee, T. S. Gang, and G. I. Jee, "Altitude Measurement Using Differential Altitude Sensors and DGPS for a UAV," in *Proceedings of the KSAS Conference*, pp.399-402, 2004.
- [5] S. H. Park, J. H. Lee, S. W. Kim, J. H. Im, and J. Y. Ryu, "Moving Distance Measurement System using a Accelerometer Sensor," *The KIICE Journal*: Vol.16, No.6, pp.1300-1305, 2012.
- [6] H. Deilamsalehy and T. C. Havens, "Sensor fused three-dimensional localization using IMU, camera and LiDAR," in *Proceedings of the IEEE SENSORS Conference*, pp.1-3, 2016.
- [7] H. Liu, M. Liu, X. Wei, Q. J. Song, Y. J. Ge, and L. Wang, "Auto altitude holding of quadrotor UAVs with Kalman filter based vertical velocity estimation," in *Proceedings of the WCICA Conference*, pp.4765-4770, 2014.
- [8] Y. G. Yusuf and H. W. Agung, "Optimizing ultrasonic and barometric sensors for quadcopter's altitude-hold using YoHe V1.2 PID and KK V2.0 boards," in *Proceedings of the ISSIMM Conference*, pp.37-42, 2016.
- [9] D. Hetenyi, M. Gotzy, and L. Blazovics, "Sensor fusion with enhanced Kalman Filter for altitude control of quadrotors," in *Proceedings of the SACI Conference*, pp.413-418, 2016.
- [10] A. H. Ahmed, A. N. Ouda, A. M. Kamel, and Y. Z. Elhalwagy, "Attitude stabilization and altitude control of quadrotor," in *Proceedings of the ICENCO Conference*, pp.123-130, 2016.

- [11] J. Zhao, Y. Li, and Z. Pei, "Design on altitude control system of quad rotor based on laser radar," in *Proceedings of the AUS Conference*, pp.105-109, 2016.



김 덕 업

e-mail : ejrduq77@naver.com
 2016년 경북대학교 컴퓨터학부(학사)
 2016년~현 재 경북대학교 컴퓨터학부
 석사과정
 관심분야 : Embedded Software Testing



윤 보 램

e-mail : yunstar1105@naver.com
 2016년 경북대학교 응용생명과학부(학사)
 2016년~현 재 경북대학교 컴퓨터학부
 석사과정
 관심분야 : Embedded Software Testing
 & Software Simulation



이 성 희

e-mail : lee3229910@gmail.com
 2013년 경북대학교 컴퓨터학부(학사)
 2015년 경북대학교 컴퓨터학부(석사)
 2015년~현 재 경북대학교 컴퓨터학부
 박사과정
 관심분야 : Embedded Software &
 Distributed Computing Testing



이 우 진

e-mail : woojin@knu.ac.kr
 1992년 경북대학교 컴퓨터학부(학사)
 1994년 KAIST 전산학과(공학석사)
 1999년 KAIST 전산학과(공학박사)
 1999년~2002년 ETRI 선임연구원
 2002년~현 재 경북대학교 컴퓨터학부
 교수
 관심분야 : Embedded Software Testing & Embedded Software
 Development Environment