

GPS 주행 및 초음파센서 회피기동 드론 시스템

김세훈, 정주용, 박민호, 조희제, 권기환, 정순호
부경대학교 컴퓨터공학과

e-mail : sehoon787@naver.com

GPS Driving and Ultrasonic Sensors Evasive Maneuvering Drone System

Se-Hun-Kim, Ju-Young-Jeong, Min-Ho-Park,
Hee-Je-Cho, Gi-Hwan Kwon, Soon-Ho Jung
Dept of Computer Engineering, Pukyong University

요 약

최근 드론 규제를 완화하겠다는 정부의 발표로 앞으로 여러 분야에서 드론이 상용화되어 이용될 것으로 예상되지만, 자율비행에 있어서 장애물을 회피하는 문제가 상용화의 걸림돌로 남아있다. 이에 본 논문에서는 GPS 기반으로 드론을 직접 제작하고, 초음파 센서를 활용하는 회피기동을 구현하여 비교적 저렴한 가격으로 자율비행 드론 시스템을 설계하여 앞으로 나오게 될 스마트 모빌리티 제품군들의 상용화에 기여할 수 있기를 기대한다.

1. 서론

최근 2025년부터 드론을 이용한 규제를 완화하겠다는 정부 발표가 있었다. 따라서 앞으로 드론이 상용화되어 여러 분야에서 이용될 것이라 예상된다. [1]

드론은 어떠한 지점 간 효율적으로 이동하는 것이 주된 이용 목적이다. 여기서 드론 서비스의 상용화를 위해서는 자율비행을 수행해야 하는데, 이 때 GPS나 실내좌표 데이터를 기반으로 목적지로 정확한 이동을 수행한다.[2]

하지만 주택가나 학교 내 등 실제 환경 속에서 이러한 좌표기반의 자율비행은 비상상황의 대처가 미약했고 이는 상용화의 걸림돌이 되어왔다. 따라서 이번 연구에서 비상상황 중 장애물 회피에 대한 해결책을 제시해보고자 한다. 현재 드론 상용화를 두고 안전사고에 대한 방지책이 지속적으로 요구되고 있고[3], 미국과 중국 등에서 충돌감지 회피 기술을 연구 개발 중이며 특히 출원을 한 사례가 있다.[4] 배송 드론을 이용하는 회사들은 현재 회피기동을 구현하기 위해서, 비교적 가격이 비싼 임베디드 칩셋을 활용하여 장애물을 회피하는 방안을 제시했다. 하지만 이보다 더 저렴한 가격의 임베디드 칩셋을 활용하여 회피기동을 구현하여 비용적 우위를 점하는 것 또한 유의미한 연구가 될 것이라 생각된다.[5]

이에 본 논문에서는 픽스호크와 라즈베리 파이를 바탕으로 GPS기반으로 기동하는 드론을 직접 제작하고, 비교적 가격이 저렴한 초음파 센서를 이용한 회피기동을 설계 및 구현하였다. 가격이 싸고 쉽게 구할 수 있는 초음파센서를 통하여 비용절감을 하며 장애물을 감지하고 회피할 수 있는 기능을 구현함으로써 장애물 회피기능을 구현하기 위해 높은 가격을 치러야 했던 기존 시스템보다 비용적 측면에서 우위를 점할 수 있다. 이러한 장애물 회피 알고리즘을 적용한 배송 드론은 생산비용의 효과적인 절약과 파

손방지 및 드론 주행의 안정성을 확보할 수 있고 또한 해당 회피기동 시스템을 다른 스마트 모빌리티 분야에도 활용할 수 있을 것이라 기대된다.

2. 관련연구

2.1 픽스호크(Pixhawk)

픽스호크 또는 픽스호크 프로젝트(Pixhawk project) 또는 'PX4'는 원격조정 드론 및 자동 조종 장치는 무인 및 자율 항공기를 지향하는 오픈 소스 자동 조종 시스템인 아두파일럿의 하위 프로젝트로 개발되었다. 저렴한 비용과 가용성으로 소규모의 원격 조종 항공기에서 아마추어가 사용할 수 있었으며 세계적으로도 많은 자율 시스템 연구소 및 자동 제어 실험실에서 이를 공동연구하고 개발을 지원하고 있다. 본 논문에서는 드론이 GPS 정보를 바탕으로 자율비행을 수행할 수 있도록 드론을 제어하는 역할을 한다.

2.2 안드로이드 스튜디오(Android Studio)

구글이 안드로이드 앱 개발을 위해 JetBrains 사의 IntelliJ IDEA를 기반으로 만든 통합 개발 환경이다. 2014년 10월부터 이클립스를 대체해 안드로이드의 공식 IDE가 되었다. 무료로 제공되고 있으며 Windows, macOS, 리눅스를 지원한다. 본 논문에서는 목표지점의 GPS 정보를 받아오면서 드론의 라즈베리 파이로 목적 좌표의 GPS를 보내, 드론이 목적지를 지정하도록 앱을 설계하였다.

2.3 라즈베리 파이(Raspberry pi)

라즈베리 파이(Raspberry pi)는 영국의 라즈베리 파이(Raspberry Pi)재단에서 교육용 프로젝트의 일환으로 개발된 초소형/초저가의 컴퓨터다. 본 논문에서는 드론에서

제어하는 코드를 구현하여, GPS 정보를 어플리케이션을 통해서 입력받고, 드론에 타겟 지점으로 이동하는 명령을 내리는 동시에 GPIO를 기반으로 초음파 센서와 연결되어 직접 회피기동의 연산을 수행한다.

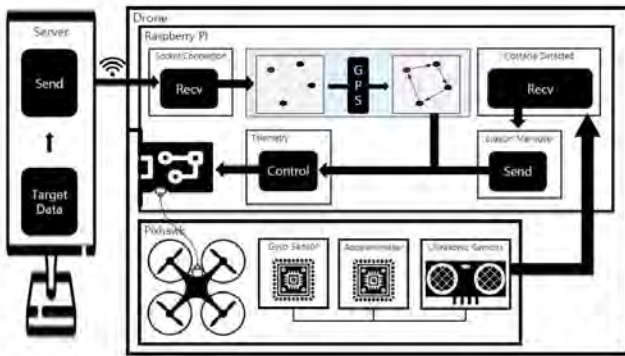
2.4 초음파 센서(Ultrasonic sensor)

초음파 센서란 주파수가 20kHz를 넘는 사람이 듣지 못하는 음파를 송, 수신하여 센서 앞의 물체를 감지하는 센서를 의미한다. HC-SR04 모듈의 경우 2cm ~ 4m의 측정 거리를 갖는다. 본 논문에서 제작한 드론은 해당 모듈을 사용하였으며 라즈베리 파이의 GPIO로 신호를 전송한다. 만약 비행 중에 초음파 센서에서 3m내에 물체가 있다고 판단할 경우 경로 상에 장애물이 있다는 것으로 간주하여 상승기동을 통해 이를 회피한다.

3. 드론 주행 및 회피기동 시스템

3.1 시스템 구조도

그림 1은 본 논문의 자율비행 드론 시스템의 핵심적인 기능을 나타낸 시스템 구조도이다.



(그림 1) 시스템 구조도

GPS를 활용한 드론은 라즈베리 파이와 PixHawk로 구성되어있다. PixHawk속에 내장된 자이로센서 및 가속도 센서와 별도로 부착되어있는 GPS 모듈을 통해 GPS 정보를 지속적으로 갱신하며 오차범위 2.5m이내(GPS성능에 따라 오차범위 개선 가능)의 주행을 한다. 서버에서는 클라이언트가 되는 라즈베리 파이와 소켓 통신을 통해 경로에 대한 GPS좌표를 전송하고 이때 라즈베리파이는 GPS와 자이로센서를 이용해 드론의 진행방향을 제어한다. 가속도계는 드론이 다음 목적지까지 이동할 수 있도록 속도를 제어한다. 초음파 센서를 통해 장애물을 감지하면 초음파 센서는 라즈베리파이의 GPIO로 신호를 보낸다. 신호를 받은 라즈베리파이는 초음파 센서의 값을 지속적으로 받아 거리 내에 장애물을 탐지하지 않을 때까지 상승기동하고 본 목적지로 주행한다.

현재 드론은 기술적으로 6000m정도 상승이 가능하며 항공안전법 제 199조에 따라 해당 항공기 중심으로 수평거리 600m 범위 안 지역에 있는 가장 높은 장애물의 상단

에서 300m 고도를 유지할 수 있다. 공중에서 가장 빈도가 높게 탐지되는 장애물은 고층 건물 또는 새와 같은 비행 생명체이다. 우선 건물의 경우 좌우로 회피하여 또 다른 건물을 마주하는 것보다 고도를 상승시켜 다른 건물을 비행 범위 내에서 없애는 것이 보다 효율적이라 생각되고 새는 활강을 하여 하강하고 상승은 속도가 빠르지 않으므로 이 또한 상승기동으로 회피하도록 설계하였다.

3.2 소스코드

그림 2는 본 논문에서 구현한 초음파 센서 소스코드를 나타낸다.

```

1 def distance():
2     # set Trigger to HIGH
3     GPIO.output(GPIO_TRIGGER, True)
4
5     # set Trigger after 0.01ms to LOW
6     time.sleep(0.00001)
7     GPIO.output(GPIO_TRIGGER, False)
8
9     StartTime = time.time()
10    StopTime = time.time()
11
12    # save StartTime
13    while GPIO.input(GPIO_ECHO) == 0:
14        StartTime = time.time()
15
16    # save time of arrival
17    while GPIO.input(GPIO_ECHO) == 1:
18        StopTime = time.time()
19
20    # time difference between start and arrival
21    TimeElapsed = StopTime - StartTime
22    # multiply with the sonic speed (34300 cm/s)
23    # and divide by 2, because there and back
24    distance = (TimeElapsed * 34300) / 2
25
26    return distance
    
```

(그림 2) 초음파 센서 장애물 감지 코드

본 논문에서는 라즈베리 파이에 초음파 센서 모듈을 사용하기 위해서 파이썬의 GPIO 라이브러리를 사용해 초음파 센서를 제어한다.

```

while flytime <= 40:
    if dist < 300 and dist >= 100: # 3M from obstacle
        print("(Go)Detect obstacle")
        i = i + 1
        while True:
            print("(Go)Altitude: ", vehicle.location.global_relative_frame.alt)
            # break and return from function just below target altitude.
            send_attitude_target(roll_angle=0, pitch_angle=0,
                yaw_angle=None, yaw_rate=0, use_yaw_rate=False,
                thrust=0.6)
            if vehicle.location.global_relative_frame.alt >= i * 0.95:
                print("(Go)Reached target altitude")
                break
            time.sleep(1)
        else:
            print("(Go)Go Forward")
            loc_point = LocationGlobalRelative(lat1, long1, 1)
            vehicle.simple_goto(loc_point, groundspeed=1)
            # Send a new target every two seconds
            # for a complete implementation of follow me you'd want adjust this delay
            time.sleep(1)
        dist = distance()
        if dist >= 100:
            print("(Go)Distance from Obstacle : " + str(dist))
            flytime = time.time() - starttime
            print("(L)Set General Landing Mode")
            vehicle.mode = vehicle.mode("LAND")
            time.sleep(1)
            print("(Go)Close vehicle object")
            vehicle.close()
            print("(Go)Ready to leave to next Landing Point")
    
```

(그림 3) 장애물 감지 시 회피기동 수행 소스코드

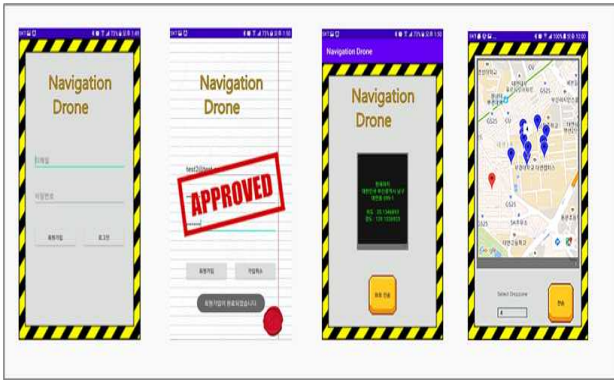
그림 3은 위의 소스코드에서 초음파센서로 장애물을 감지 시 회피기동 기능을 수행하는 프로그램 소스코드이다. 초음파 센서로부터 전방에 장애물이 감지되었을 때 send_attitude_target 함수를 이용하여 Thrust값을 조정해 고도를 상승시키는 방식으로 장애물을 회피한다.

3.3 구현



(그림 4) GPS 기반 자율비행 드론

그림 4는 직접 제작한 GPS기반 자율비행 드론 본체이다. GPS 정보를 기반으로 드론은 자신의 위치를 찾으며, 입력 받은 목적지로 향한다. 이때 드론 기체 정면 및 측후면에 장착된 초음파 센서를 이용하여, 주위의 장애물로부터 기체의 거리를 측정한다. 거리값(Distance from Obstacle)이 3m 이하일 때 회피 상승 기동을 수행한다.



(그림 5) 목적지 GPS 좌표 전송 어플리케이션

그림 5는 드론을 제어하는 서버로 목적지의 GPS 좌표를 입력하고, 전송하는 어플리케이션이다. 서버는 사용자 어플리케이션으로부터 받은 드론은 목적지의 GPS좌표를 드론의 라즈베리 파이로 전송하고, 라즈베리 파이는 전달받은 목적지에 따라서 자율비행을 수행한다.



(그림 6) 자율비행 드론의 회피기동 수행 도식

그림 6은 본 논문의 자율비행 드론 시스템의 회피기동을 수행하는 모습을 그림으로 나타낸 것이다.

```

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Sep 23 15:17:39 2020
jff@raspberrypi:~$ python3 GPS_test2.py
Going towards first point for 40 seconds ...
Take off!
Basic pre-arm checks
Arming motors
Waiting for arming...
Taking off!
Altitude: 0.049
Altitude: 0.095
Altitude: 0.135
Altitude: 1.165
Altitude: 1.741
Altitude: 1.802
Reached target altitude
Set default/target airspeed to 3
Angle positioning and move toward
Go Forward
Go Forward
Go Forward
Go Forward
WARNING:autopilot:Battery 1 is low 0.50V used 20 mAh
WARNING:autopilot:Battery Failure
Go Forward
Go Forward
Go Forward
Go Forward
Go Forward
Distance from Obstacle : 166.32331605725952
Detect Obstacle
5
Altitude: 2.51
Altitude: 2.674
Altitude: 3.218
Altitude: 3.628
Altitude: 4.143
Altitude: 4.744
Altitude: 5.114
Reached target altitude
Go Forward
Go Forward
    
```

(그림 7) 코드 출력 결과

그림 7은 본 논문에서 설계 및 구현한 회피기동 코드의 로그로 실제로 회피기동을 수행할 시 출력되는 결과이다. 처음 드론이 이륙할 때 정해진 대로 고도(Altitude)를 상승시키고, 만약 장애물이 없다면 그대로 진행(Go Forward)한다. 그러다 만약 장애물이 감지된다면 (Detect Obstacle) 장애물이 감지되지 않을 때 까지 고도를 상승시키고, 만약 장애물이 감지되지 않는다면 다시 처음 지정해 둔 경로대로 목적지까지 이동한다.

4. 결론

본 논문에서는 가격이 저렴한 초음파 센서를 활용하여 GPS를 기반으로 자율비행을 수행하는 드론을 직접 제작하였고, 회피기동 알고리즘을 추가하여 장애물 회피기동 기술을 구현하였다. 또한, 실제로 좌표 기반 자율 비행 시험을 통해 GPS 기반의 자율 비행 중 장애물이 나타날 시 드론이 회피기동을 수행 한 후, 다시 정해진 경로대로 자율 비행을 마치는 것을 직접 확인할 수 있었다. 본 논문에서 직접 설계한 회피기동 알고리즘을 활용하여 외부에서 GPS를 기반으로 드론을 운용하는 것 뿐만 아니라 실내에서도 상대좌표를 기반으로 드론을 운영할 때 더욱 안정적인 자율비행을 수행하게 만드는 것을 기대할 수 있다. 해당 기술은 드론을 포함한 모든 스마트 모빌리티들이 좌표를 기반으로 자율 주행할 때 적용할 수 있으므로 다른 스마트 모빌리티들도 적은 비용으로도 장애물을 회피하는 움직임을 구현하여 자율주행의 효율을 높일 수 있을 것으로 예상된다. 이러한 초음파 센서를 통한 회피기동과 GPS 기반 자율주행 기술은 앞으로의 4차 산업혁명 속에서 자율주행 제품 상용화에 기여할 것으로 기대된다.

참고문헌

- [1]. 국토교통부, ‘한국형 도시항공교통(K-UAM) 로드맵 발표’, www.molit.go.kr, 20200604.
- [2]. “코로나는 드론 배송에 날개를 달아줄까”, <한겨레>, 20200622.
<http://www.hani.co.kr/arti/science/future/950340.html#csidx6f1d44217039247a15ed2216cf479dc>
- [3]. 한국항공우주연구원, 정연득, “드론 사고 위험성을 줄이기 위한 충돌 회피 기술 동향” e-정책정보센터, 2016.03.
- [4]. 과학기술일자리진흥원, “드론 기술 및 시장동향 보고서” S&T Market Report, vol. 67, 2019.08
- [5]. 한국전자통신연구원, Jeong Hun, “드론을 이용한 물류서비스 추진 방향”, 우정경영연구소, 2015
- [6]. 국가법령정보센터, 항공안전법 시행규칙, 제 199조(최저비행고도), [시행 2020. 5. 27.] [국토교통부령 제730호, 2020. 5. 27., 일부개정]