

Tello 군집드론실습

인텔코리아 이 해 영

여러 대의 드론을 어떻게 동시에 제어할까?



여러 대의 드론을 어떻게 동시에 제어할까?

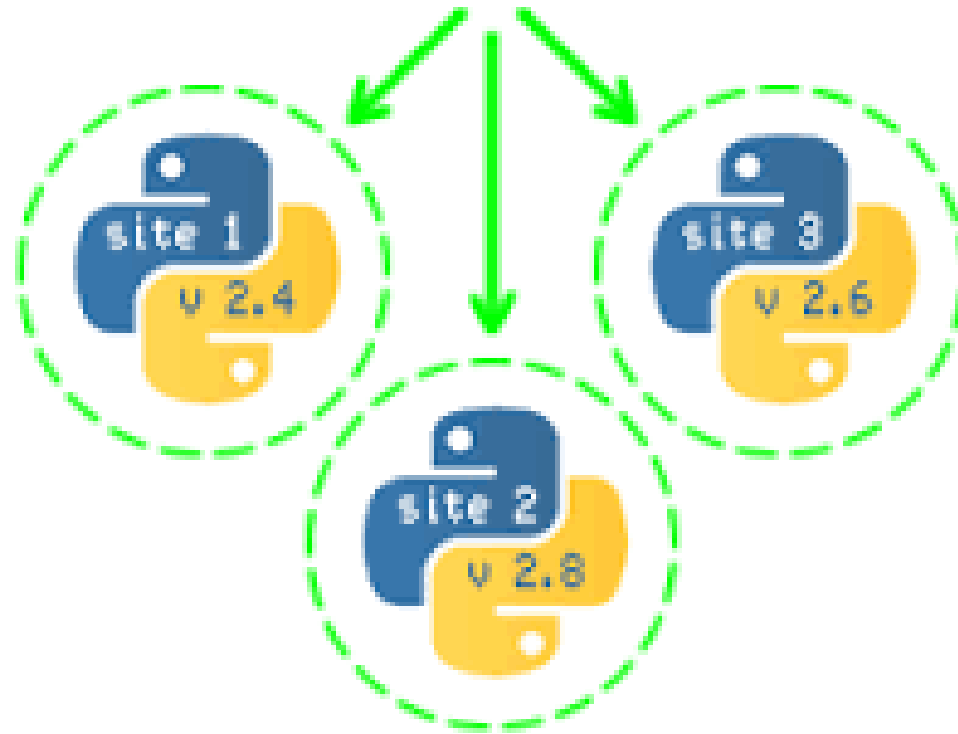


순서

1. 실습을 위한 가상환경을 만듭니다.
2. 군집 비행 코드 실행을 위해서 필요한 환경을 세팅합니다.
3. 군집 드론에 사용할 WiFi 공유기를 지정합니다.
4. formation_setup.py에서 맨 마지막 줄 set_ap 에 WIFI 공유기의 ID, PW 를 기록해줍니다.
5. 핸드폰에 어플(Tello)을 설치하고, 핸드폰에서 드론의 WiFi를 찾아 연결한 후 드론의 펌웨어를 최신으로 업데이트 합니다. (설정-더보기-'...' -펌웨어버전-업데이트)
6. 핸드폰과 드론의 WIFI 연결을 종료합니다 .
7. 여러대의 드론을 동시에 제어하기 위해 하나의 WiFi망(1번의 WiFi)에 연결되도록 합니다. 사용할 드론들을 하나씩 연결합니다. 첫번째 드론을 PC와 와이파이로 연결합니다. 그리고 formation_setup.py 를 실행 합니다. 연결되었다는 문구와 함께 드론은 재부팅되며 연결이 완료되고 드론의 와이파이는 PC에 종속되어 보이지 않습니다.
8. 두번째, 세번째 드론도 동일하게 합니다.
9. 이렇게 스테이션 상태가 되면 multi_tello_test.py를 이용해 군집비행을 준비 합니다.
10. 군집 비행 명령어는 Tello_EDU_CMD.txt에 저장 합니다.

1. 실습을 위한 가상환경 만들기

> virtualenv



1. 실습을 위한 가상환경 만들기

- python 라이브러리 차이에 의해 발생하는 충돌을 사전방지해주기 위해 가상환경이 필요합니다.
- virtualenv 이라는 모듈을 통해 가상환경을 만들어줍니다.

```
// “swarm” 이라는 이름의 디렉토리 만들기
$ mkdir swarm
// 만들어진 “swarm” 디렉토리로 이동한 후 “drone” 이라는 가상환경 만들기(인터넷 연결이 되어있는
상태여야 합니다)
$ cd swarm
$ virtualenv -p python drone
// 만들어진 “drone” 가상환경에 접속하기
$ source drone/bin/activate
```

2. 환경 세팅하기

- . 군집비행 코드 실행을 위해 필요한 모듈을 설치합니다.(인터넷 연결되어야함)
- . “netifaces”, “netaddr” 모듈이 필요합니다.

```
// netifaces 모듈 설치하기  
$ pip install netifaces  
// netaddr 모듈 설치하기  
$ pip install netaddr
```

- netifaces : 네트워크 관련 인터페이스, IP 주소, 네트워크 정보들을 쉽게 가져올 수 있게 해주는 파이썬 라이브러리
- netaddr : 네트워크 주소를 표현하고 조작하기 위한 파이썬 라이브러리
(IPv4 & IPv6 주소, 서브넷, MAC 주소)

3. WIFI 공유기 지정



4. 군집 비행 파이썬 코드 다운로드

- www.github.com/hy310/swarmdrone

```
// github에서 군집드론 파이썬 코드 다운로드  
$ git clone www.github.com/hy310/swarmdrone
```

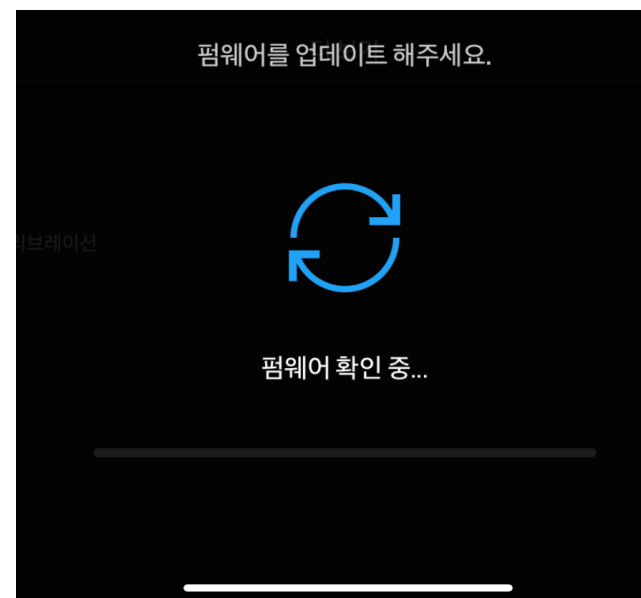
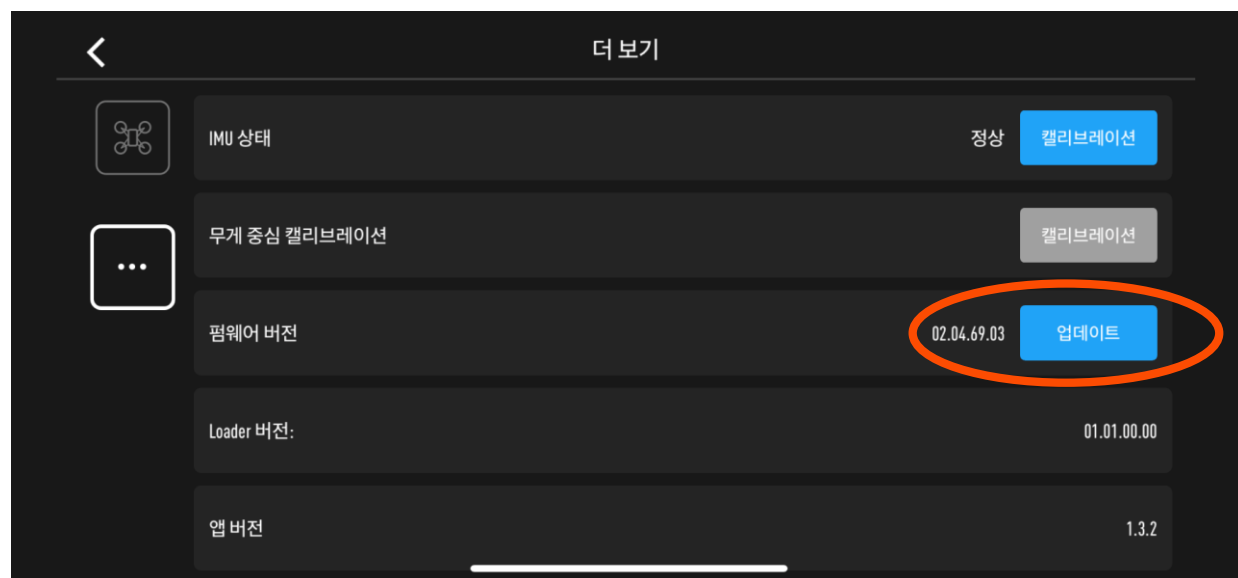
4. formation_set.py

```
1  import socket
2
3  def set_ap(ssid, password):
4      """
5      A Function to set tello in AP mode
6      :param ssid: the ssid of the network (e.g. name of the Wi-Fi)
7      :param password: the password of the network
8      :return:
9      """
10     my_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM) # socket for sending cmd
11     my_socket.bind(('', 8889))
12     cmd_str = 'command'
13     print('sending command %s' % cmd_str)
14     my_socket.sendto(cmd_str.encode('utf-8'), ('192.168.10.1', 8889))
15     response, ip = my_socket.recvfrom(100)
16     print('from %s: %s' % (ip, response))
17
18     cmd_str = 'ap %s %s' % (ssid, password)
19     print('sending command %s' % cmd_str)
20     my_socket.sendto(cmd_str.encode('utf-8'), ('192.168.10.1', 8889))
21     response, ip = my_socket.recvfrom(100)
22     print('from %s: %s' % (ip, response))
23
24     # example of setting Tello into command mode
25     # only works if server is connected to Tello
26     set_ap('SK_WiFi7CFA', '1501054383')
```

setup_ap('공유기WIFI ID', 'password')
//사용할 WiFi 공유기 ID, PW 를 넣어주세요.
ex) ('intel_tello', 'intel1234')

5. 펌웨어 업데이트

- 핸드폰에 어플(Tello)을 설치하고, 핸드폰에서 드론의 WiFi를 찾아 연결한 후 드론의 펌웨어를 최신으로 업데이트 합니다. (설정-더보기-'...' -펌웨어버전-업데이트)



7. 드론들을 하나의 WIFI 망으로 연결하기

- . 사용할 드론들을 하나씩 연결합니다. 첫번째 드론을 PC와 와이파이로 연결을 합니다. 그리고 formation_setup.py 를 실행 합니다.
- . 연결되었다는 문구와 함께 연결이 완료되고 드론의 와이파이는 PC에 종속되어 보이지 않습니다. 두번째, 세번째 드론도 동일하게 합니다.

```
// PC에서 첫번째 드론 WiFi ID를 찾아서 연결하고 하기 코드를 실행 합니다.  
$ python formation_setup.py  
// PC에서 두번째 드론 WiFi ID를 찾아서 연결하고 하기 코드를 실행 합니다.  
$ python formation_setup.py  
// 나머지 드론들도 동일하게 합니다.
```

9. multi_tello_test.py를 이용해 군집비행을 준비

- 스테이션 상태가 되면 multi_tello_test.py를 이용해 군집비행을 준비 합니다.
- 실행 하고자 하는 군집 비행 명령어를 Tello_EDU_CMD.txt에 저장 합니다.
- 명령창에서 실행할 때 multi_tello_test.py 뒤에 txt 파일을 붙여줍니다.
ex. multi_tello_test.py Tello_EDU_CMD.txt
- 하기는 Tello_EDU_CMD.txt 예제 입니다.

```
scan 2  
battery_check 20  
correct_ip  
1=0TQDG41EDBB2GR  
2=0TQDG3SEDBWKF2  
*>takeoff  
sync 10  
*>land
```

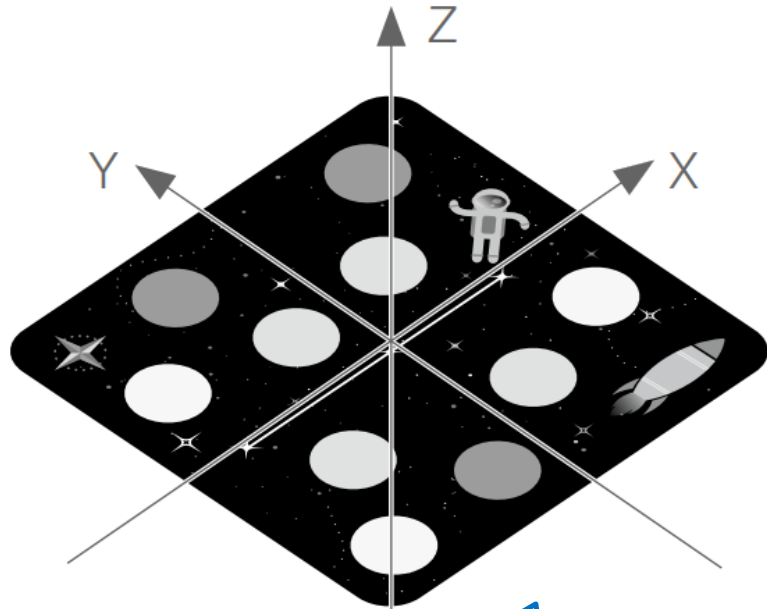
```
scan 2 → 2대의 드론을 군집비행 하도록 지정.  
battery_check 20 → Battery 가 20% 미만으로 되면 비행을 멈추기.  
correct_ip → 드론의 SN 넘버와 와이파이를 연결 하기.  
1= 0TQDG41EDBG3GR → 첫번째 드론의 SN 넘버 적어 넣기.  
2= 0TQDG3SEDDDKF2 → 두번째 드론의 SN 넘버 적어 넣기.  
*>takeoff → *> 는 연결된 모든 드론에 적용. 1>은 첫번째 드론에만 적용.  
Sync 10 → 10 초동안 sync.  
*>up 30  
sync 15  
1>up 30  
2>down 30  
sync 15  
*>land → 착륙
```

10. 군집비행

- 하기 명령어를 통해 군집 비행을 실행합니다. **이 때 명령어를 입력하기 전 set_ap에 입력한 WiFi 공유기에 pc를 먼저 연결한 이후,** 아래 명령어를 입력합니다.
- 군집비행을 변경하고자 하면 원하는 내용을 Tello_EDU_CMD.txt에 기록 합니다.
- Tello 의 군집 비행시 드론간의 움직임의 편차를 줄이기 위해서는 어떤 방법이 있을 까요?

```
$ python multi_tello_test.py Tello_EDU_CMD.txt
```

참조) 미션패드 사용하기



로켓 모양 쪽이 (+) X축 방향,
바람개비 모양 쪽이 (+) Y축 방향입니다.

드론이 미션패드를 인식하면 미션패드의 X, Y,
Z축을 인식해 스스로 회전하여 정렬됩니다.

- 드론 하나당 4개의 미션패드가 들어있습니다. 미션 패드 위에 적힌 숫자대로 각각 m1, m2, m3, m4입니다.
- 미션패드를 이용하면 내가 원하는 위치에 드론을 정렬시킬 수 있어서 미션패드를 사용하지 않을 때보다 더 쉽게 질서있게 움직일 수 있습니다.
- 미션패드 명령어는 jump를 사용합니다. jump는 현 위치에서 미션패드를 인식해 자리를 먼저 잡은 후 작성한 명령어가 실행되고, 그 후 두번째 미션 패드로 이동하여 그 미션패드의 축에 맞게 정렬되는 것입니다.
jump x좌표(앞뒤), y좌표(좌우), z좌표(위아래), 속도, 회전, 미션패드1, 미션패드2
* 회전의 경우 특수하게 미션패드2로 가서 정렬이 된 후 제일 마지막으로 회전합니다.
- 한 단계 명령이 끝난 후 sync 를 통해 모든 드론들을 다시 전체적으로 준비시켜 주고 sync가 끝나면 다음 명령을 동시에 진행합니다.
- 안전한 비행을 위해서는 한번 명령을 시행할 때 100cm 이상, sync 40 이상을 권장합니다.

참조) 미션패드를 이용한 실습

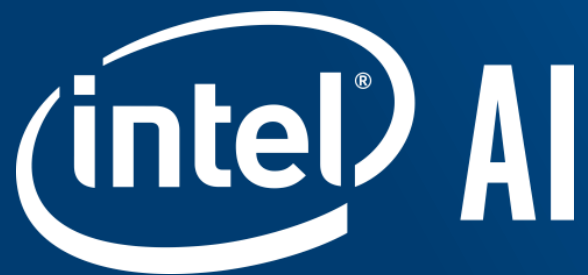
```
scan 4
battery_check 20
correct_ip
1=0TQDG41EDBB2GR
2=0TQDG3SEDBWKF2
3=0TQDG42EDBNHG5
4=0TQDG41EDB84MX
*>mon                                // 미션패드를 이용
*>takeoff
sync 40
1>jump 100 0 100 60 0 m1 m2           // 1번드론의 경우 미션패드 1번 기준으로 x축방향으로 100cm 가고,
2>jump 100 0 100 60 0 m2 m3           // z축 방향으로(높이) 100cm가고, 속도는 60. 그 이후 미션패드 2를 인식해서
3>jump 100 0 100 60 0 m3 m4           // 그 축방향대로 드론 회전되며 자동정렬됨
4>jump 100 0 100 60 0 m4 m1           // 나머지 드론도 이런 방식대로 움직이게 됨
sync 40
1>jump 100 0 100 60 0 m2 m3
2>jump 100 0 100 60 0 m3 m4
3>jump 100 0 100 60 0 m4 m1
4>jump 100 0 100 60 0 m1 m2
sync 40
*>land
```

미션패드 4개를 어떻게 배치하면 위의 코드처럼 forward 값만을 가지고
드론 4개가 원을 그리며 회전하게 할 수 있을까요?

텔로 명령어 example

- takeoff
- sleep 5
- up 30
- sleep 5
- left 20
- sleep 5
- right 20
- sleep 5

- forward 20
- sleep 5
- back 20
- Sleep 5
- cw 90
- sleep 5
- land



Thank you