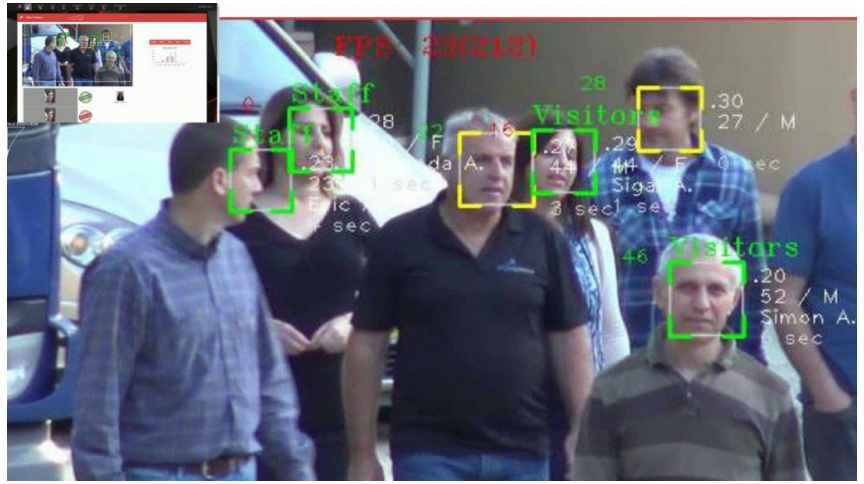
부산동고등학교 AI 메이커 동아리

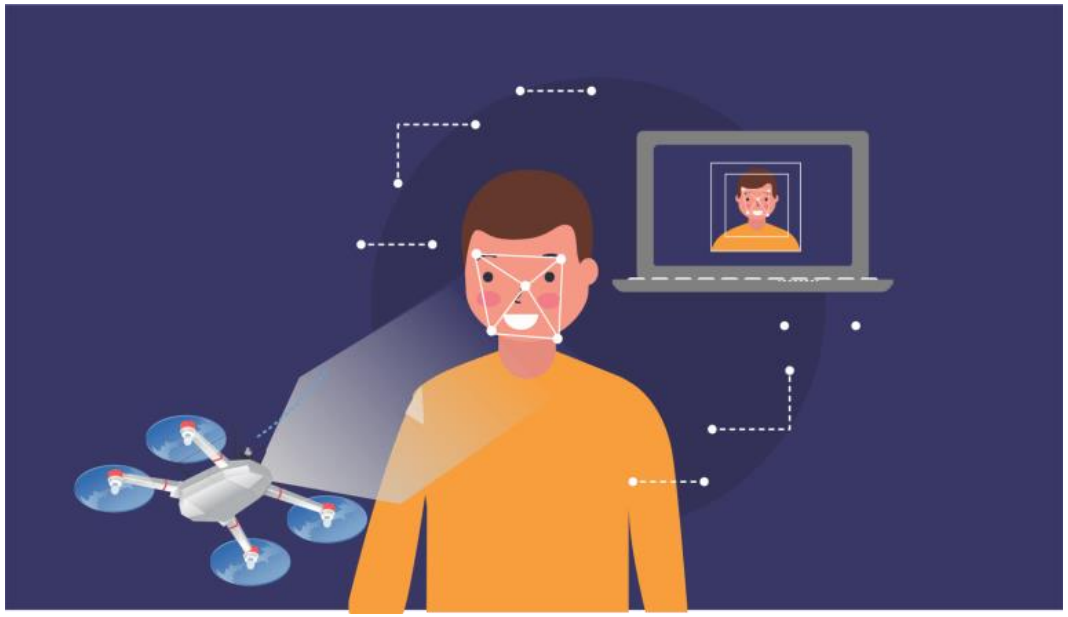


AI 드론 제작

동의과학대학교
컴퓨터정보과
김 종 현 교수 jhkim@dit.ac.kr

강의 내용

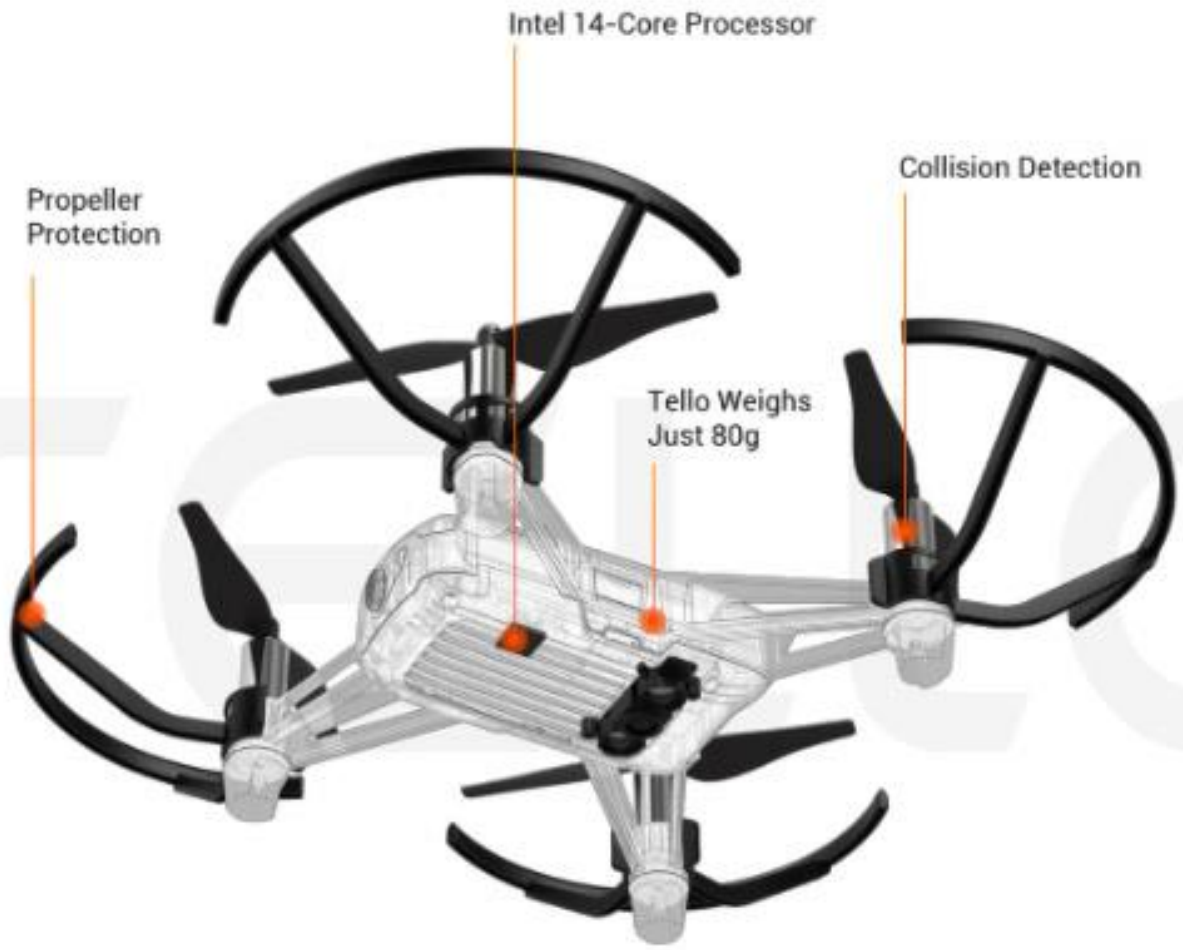
- DJI Tello 드론 기초
  - 드론의 종류, 비행원리, Tello 드론 구성요소(HW, SW) 등
  - 드론 앱을 사용한 드론 비행 기초 실습
  - 드론 비행시 주의 할 점
- Tello SDK를 이용한 파이썬 코딩(1)
  - 파이썬 프로그래밍 기초
  - DJITelloPy 모듈
  - 기본 동작 제어
    - takeoff, land, up/down, forward/backward, cw/ ccw 등
  - 키보드 제어
- Tello SDK를 이용한 파이썬 코딩(2)
  - OpenCV 기초
  - 드론 카메라 이미지 캡쳐 및 저장
  - 드론 동영상 전송 및 저장
- 파이썬 기반 AI 드론 코딩
  - Cascade Classifier를 이용한 안면 인식
  - 드론 제어(PID 제어)
  - following me 드론 제작
- 팀 프로젝트 : 창의적인 AI 드론 제작



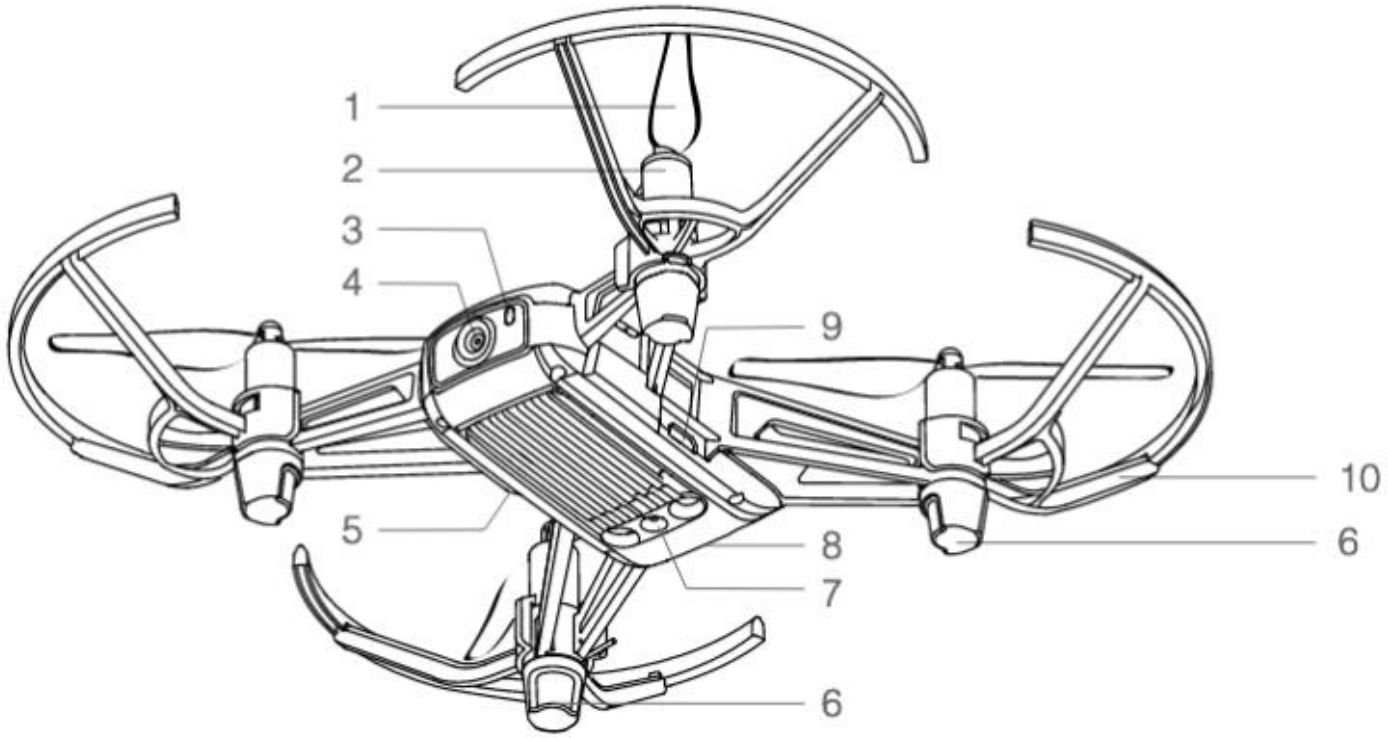
강의 자료, 소스 코드 다운로드

- https://github.com/BSDH-AI-Drone

Tello Drone 구성 요소



Tello Drone 구성 요소



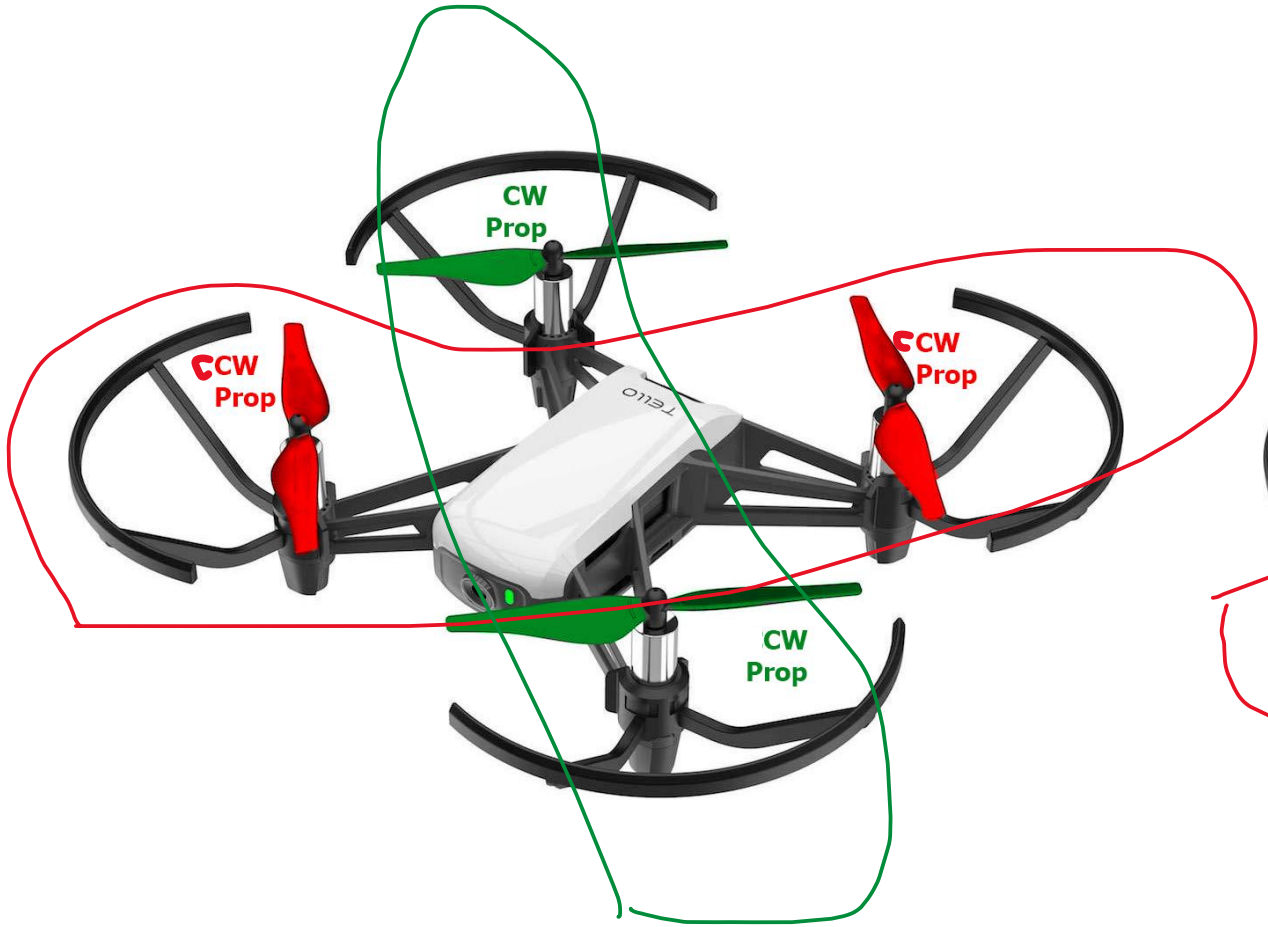
1. Propellers
2. Motors
3. Aircraft Status Indicator
4. Camera
5. Power Button
6. Antennas
7. Vision Positioning System
8. Flight Battery
9. Micro USB Port
10. Propeller Guards

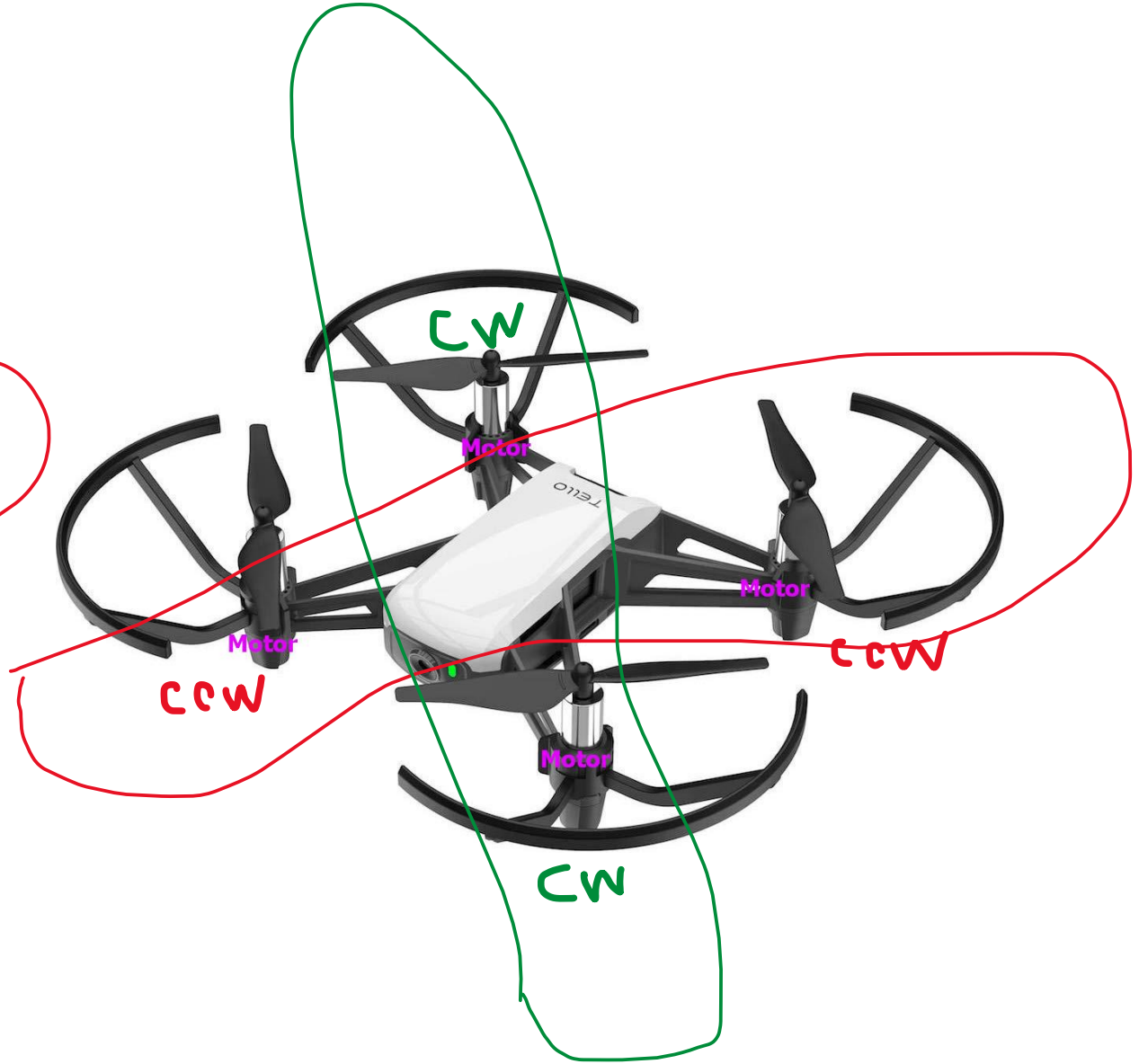
Tello 드론 사양

Weight	87 g
Dimensions	98×92.5×41 mm
Propeller	3 inches
Integrated Functions	Telemetric sensor
	Barometer
	LED
	Vision System
	Wi-Fi 2.4 GHz 802.11n
	Real-time streaming 720p
Port	USB battery charging port
Operating temperature range	from 0° to 40°
Operating frequency range	from 2.4 to 2.4835 GHz
Transmitter (EIRP)	20 dBm (FCC)
	19 dBm (CE)
	19 dBm (SRRC)

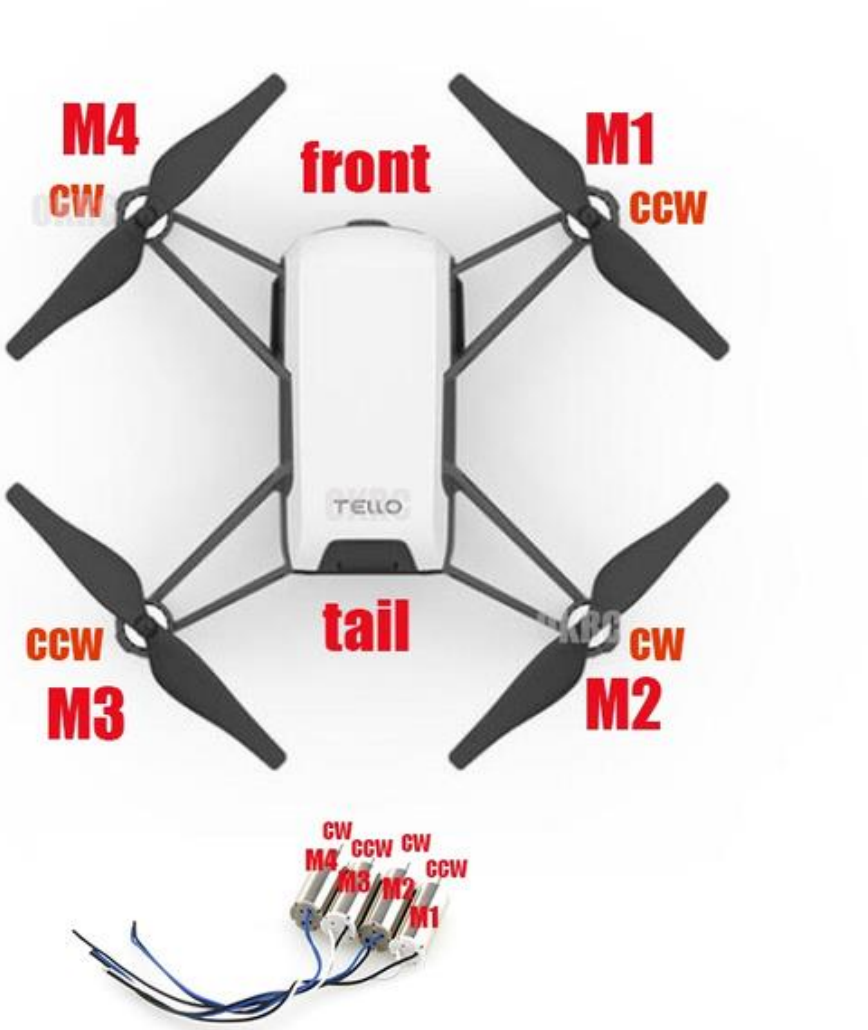
참고 : https://dl-cdn.ryzerobotics.com/downloads/Tello/Tello%20User%20Manual%20v1.4.pdf

프로펠러/ 모터

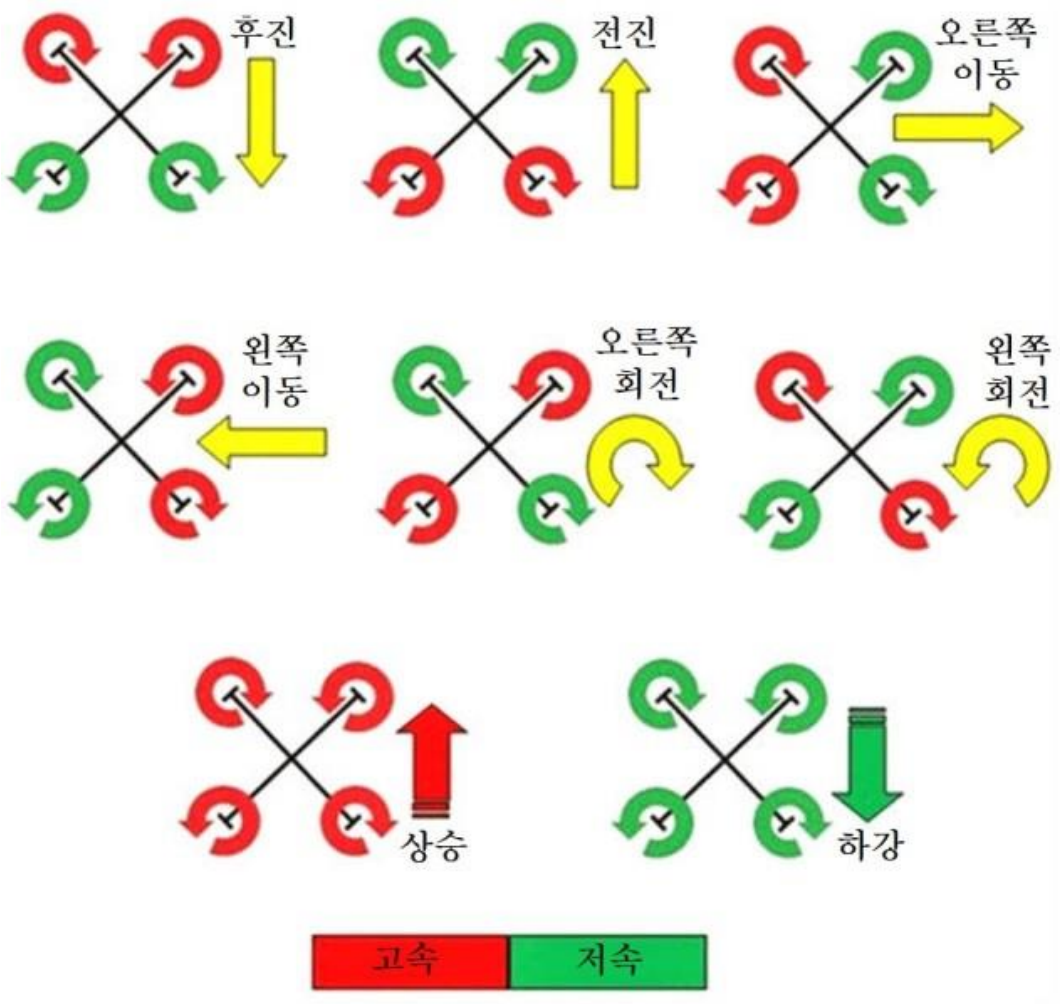

CW
Prop
CCW
Prop
CCW
Prop
CW
Prop


CW
Motor
Motor
CCW
CCW
Motor
Motor
CW

프로펠러/ 모터


M4
front
M1
cw
ccw
TELLO
tail
ccw
cw
M3
M2
cw
ccw
cw
ccw
M4
M3
M2
M1

드론의 비행 원리 : Quadcopter



Tello 드론 전용 앱

Tello App

Tello App can experience more flight modes of Tello, with real-time image-transmission interface and camera,video-recording functions, which can easily experience the fun of aerial-photography. Tello app can also set the parameters of the drone, upgrade the firmware and calibrate the drone. Therefore, the Tello app is an essential software for using the Tello.

Download on the App Store

Requires iOS 9.0 or later.

Download for Android

Android version 4.4.0 or later.

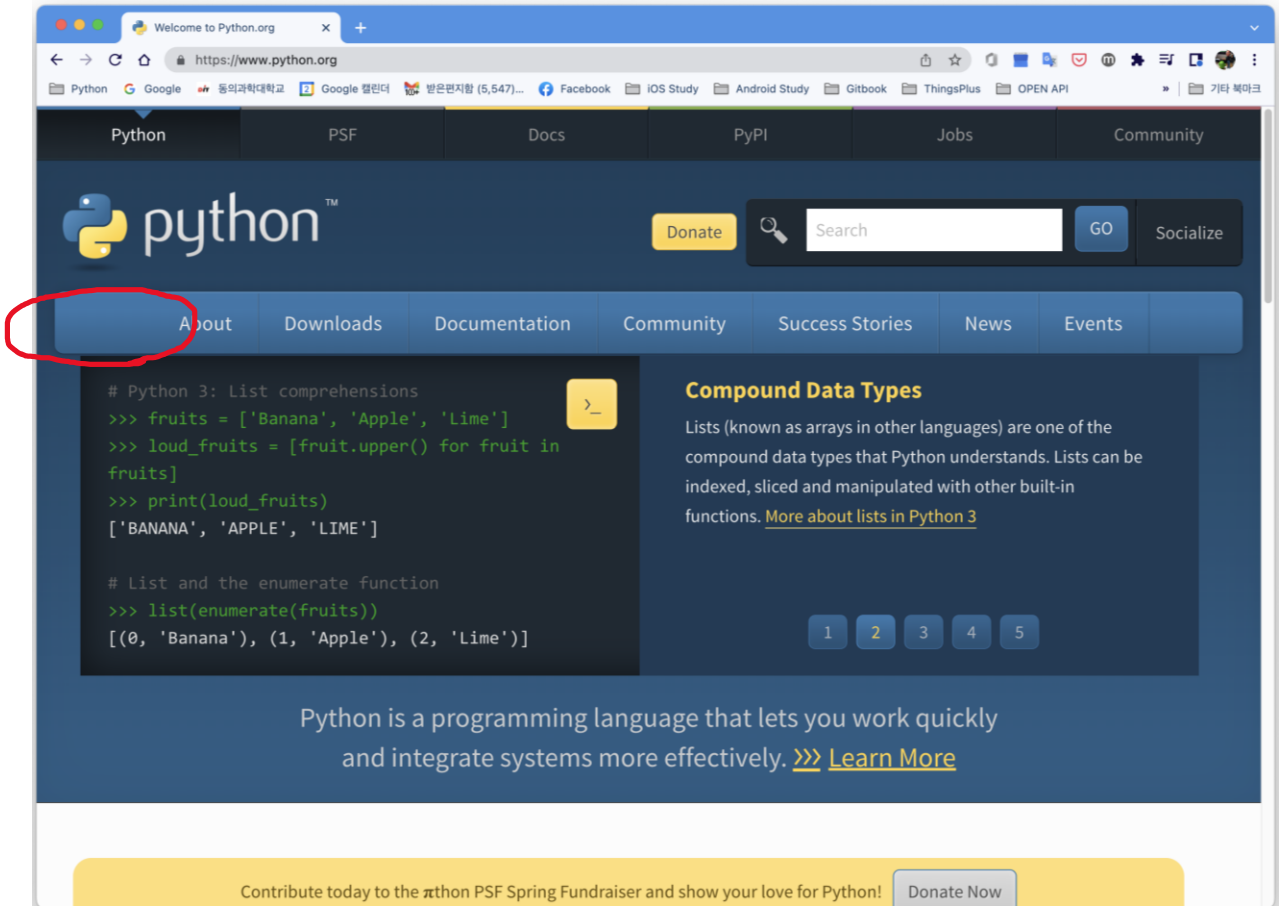
Tello 드론 전용 앱

- Tello 사용자 매뉴얼 : https://bit.ly/3ygby6T

Tello SDK를 이용한 파이썬 코딩(1)

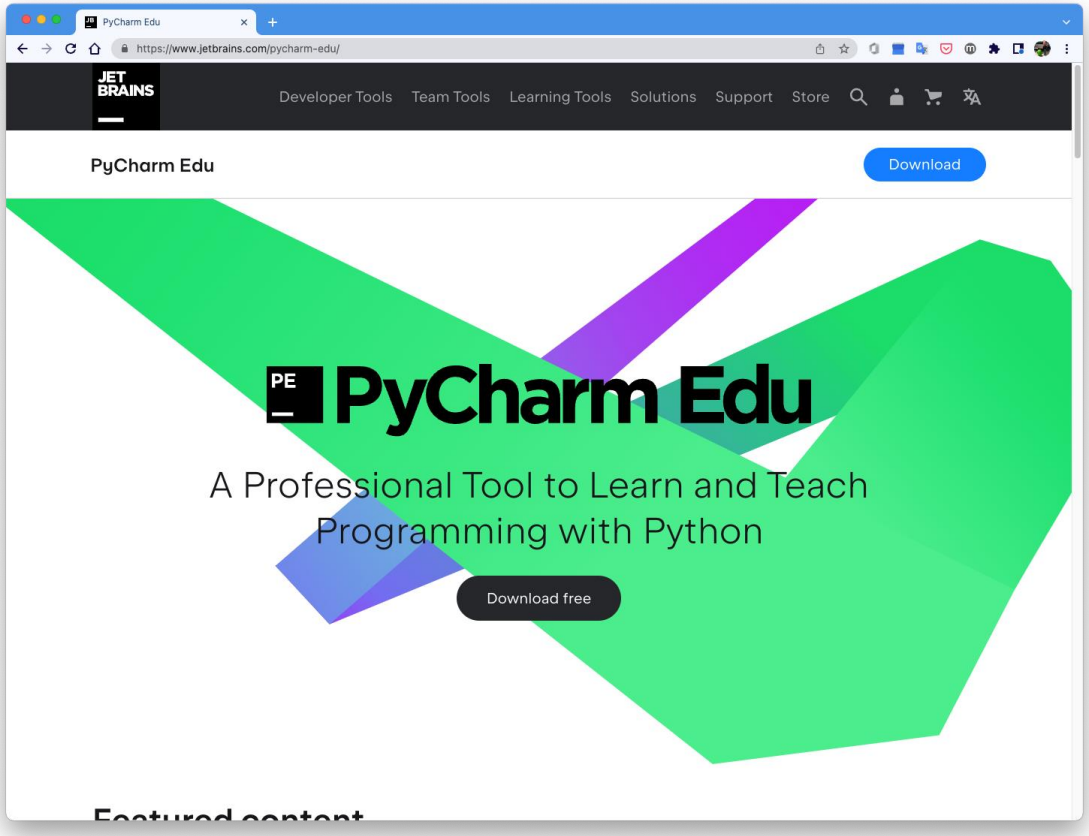
파이썬 설치

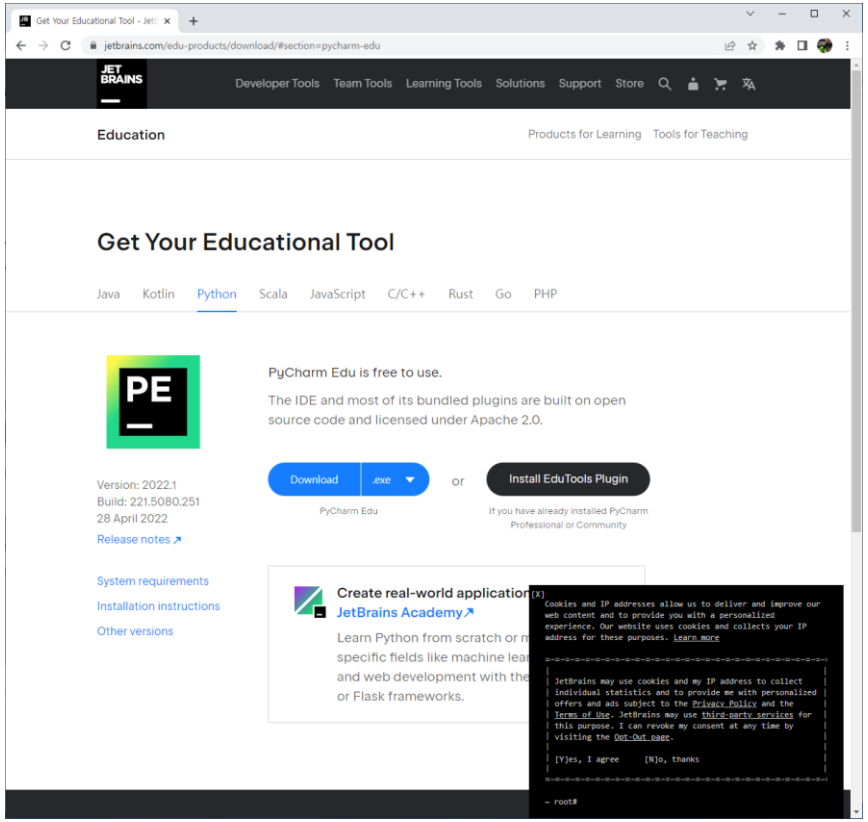
- 파이썬 공식 사이트 : https://www.python.org/
- 파이썬 3.7 ~ 3.8 다운로드



PyCharm(파이썬 통합개발도구) 설치

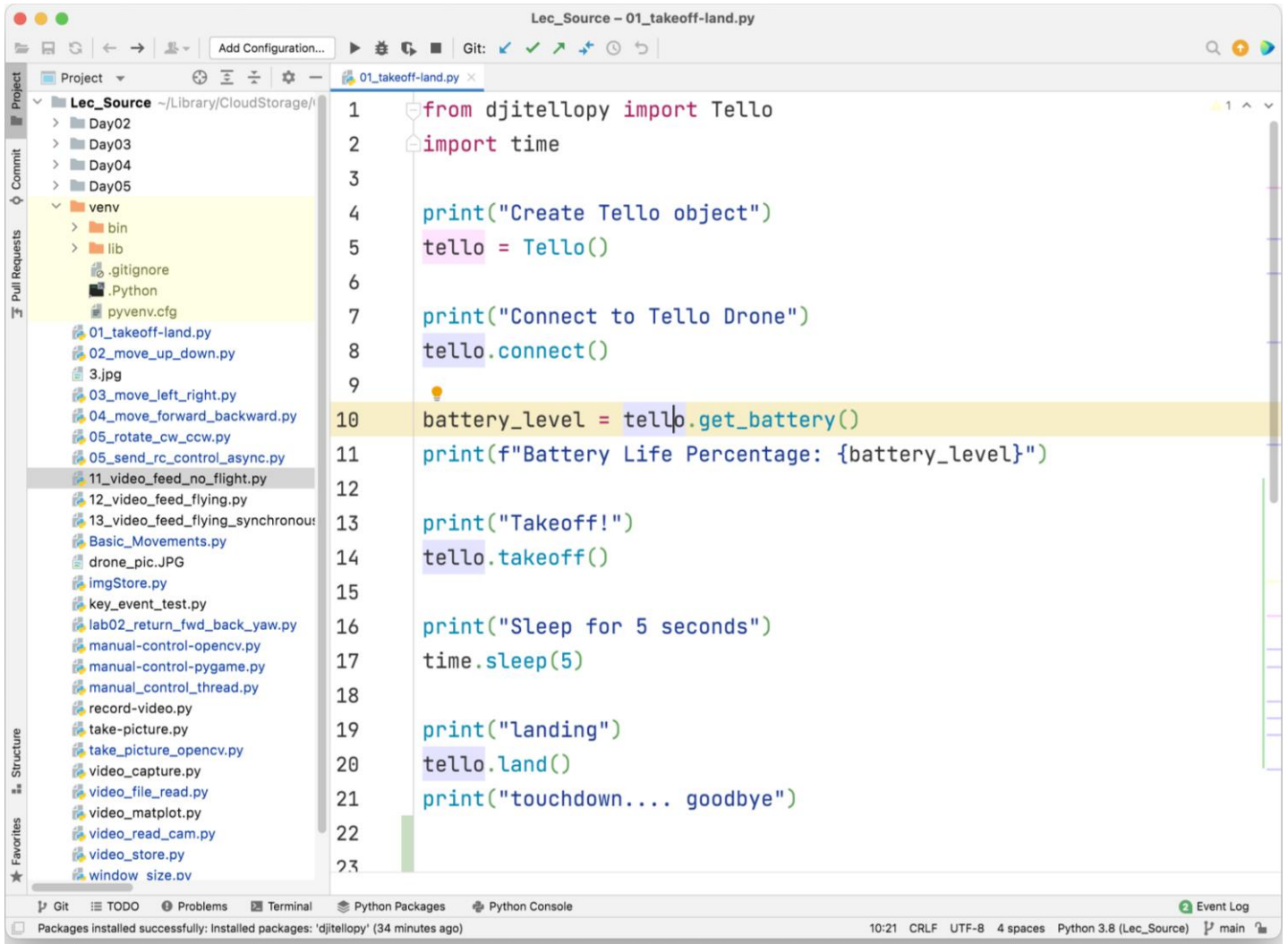
- Pycharm Edu 다운로드 및 설치
  - https://www.jetbrains.com/ko-kr/pycharm-edu/





PyCharm 사용하기

- https://blog.dalso.org/language/python/13534



파이썬 기초 프로그래밍

PYTHON BASICS

- Python Basics 다운로드
  - https://bit.ly/3yiBxxz

```
Code:
print('Hello World')
myData = 'Hello World'
print(len(myData))
print(type(myData))
Result:
Hello World
11
<class 'str'>
```

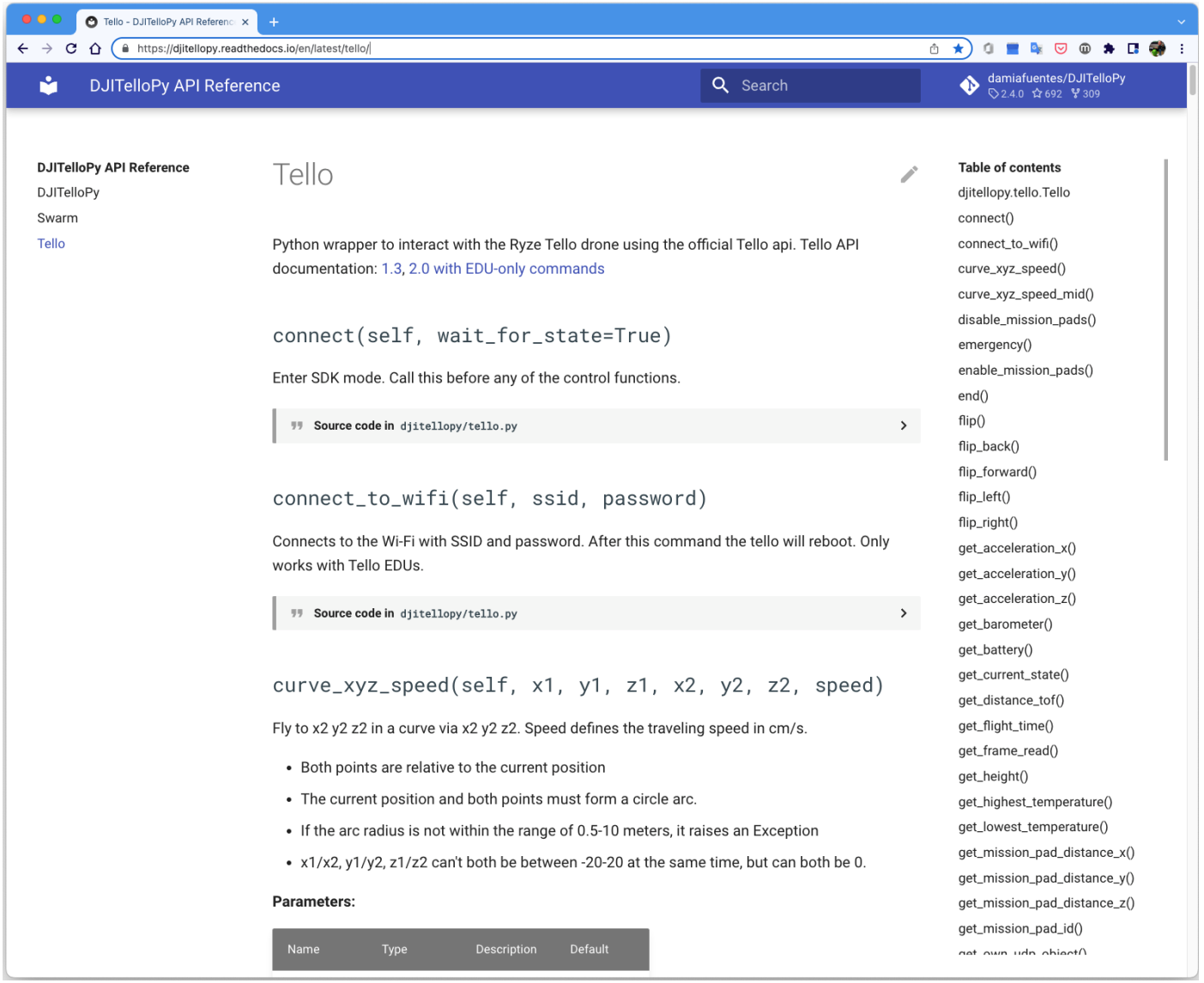
by Murtaza Hassan

DJITelloPy 모듈

- API : https://djitellopy.readthedocs.io/en/latest/tello/
- DJITelloPy 모듈 설치
  - PyCharm
    - [setting]->[Project]->[Python Interpreter] -> + 'djitellopy'
  - Terminal
    - pip install djitellopy

DJITelloPy 모듈 API :

https://djitellopy.readthedocs.io/en/latest/tello/

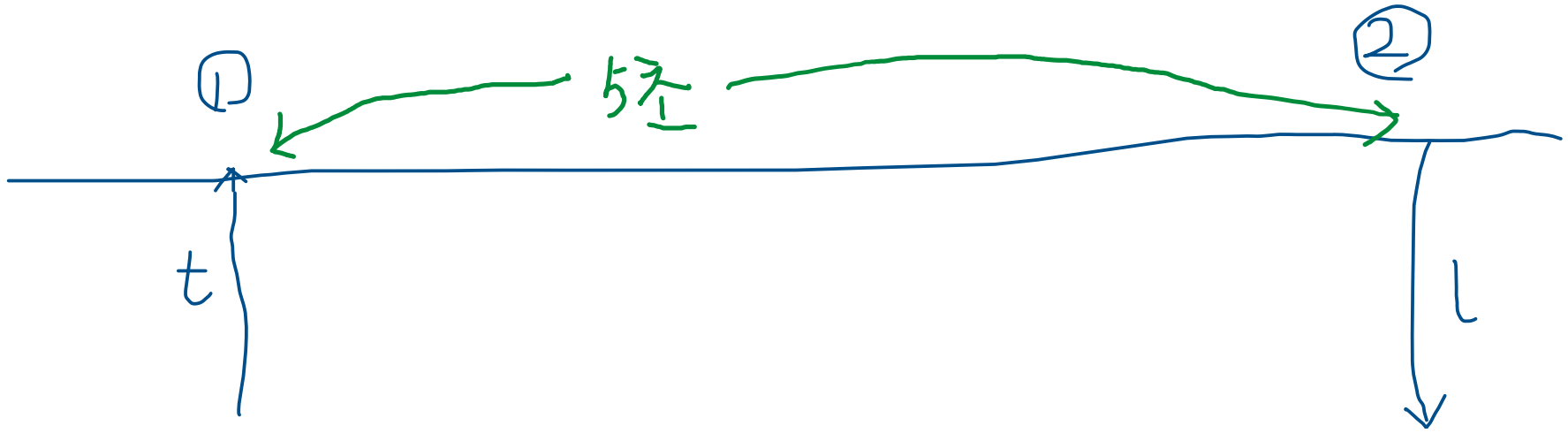


기본 동작 제어

- takeoff/ land
  - `takeoff(), land()`
- move up/ down
  - `move_up(), move_down()`
- move left/ right
  - `move_left(), move_right()`
- move forward/ backward
  - `move_forward(), move_backward()`
- rotate_cw_ccw
  - `rotate_clockwise(), rotate_counter_clockwise()`
- send_rc_control_async
  - `send_rc_control(self, left_right_velocity, forward_backward_velocity, up_down_velocity, yaw_velocity)`
- Example Code : https://github.com/damiafuentes/DJITelloPy/tree/master/examples

실습 01

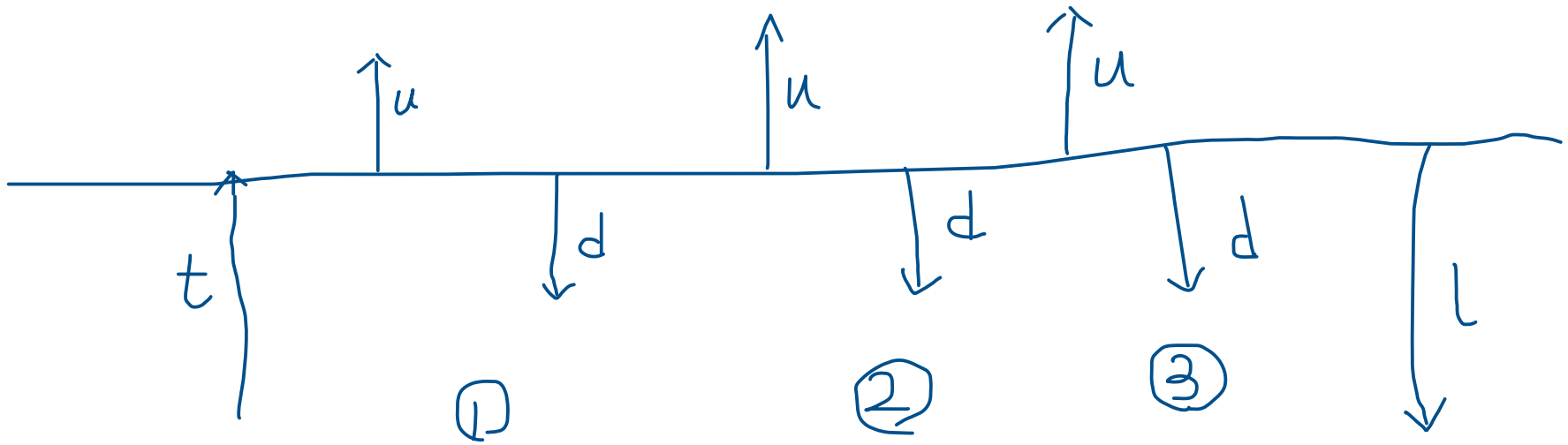
- takeoff -> landing



```
from djitellopy import Tello
import time

print("Create Tello object")
tello = Tello()

print("Connect to Tello Drone")
tello.connect()

battery_level = tello.get_battery()
print("Battery Life Percentage: “, battery_level)

print("Takeoff!")
tello.takeoff()

print("Sleep for 5 seconds")
time.sleep(5)

print("landing")
tello.land()
print("touchdown.... goodbye")
```

실습 02

- takeoff -> 3회 up(40) -> down(40) 반복 -> landing



- 참고

```
from djitellopy import Tello

tello = Tello()

tello.connect()
tello.takeoff()

tello.move_left(100)
tello.rotate_clockwise(90)
tello.move_forward(100)

tello.land()
```

반복문 이용하기 : for in range()

```
from djitellopy import Tello

myTello = Tello()
myTello.connect()
myTello.takeoff()

myTello.move_up(30)
myTello.move_down(30)

myTello.move_up(50)
myTello.move_down(50)

myTello.move_up(50)
myTello.move_down(50)

myTello.land()
```

1

2

3

```
from djitellopy import Tello

myTello = Tello()
myTello.connect()
myTello.takeoff()

for i in range (0,3) :
    myTello.move_up(30)
    myTello.rotate_counter_clockwise(90)
    myTello.move_down(30)

myTello.land()
```

파이썬 함수 이용하기 : def

```
from djitellopy import Tello

myTello = Tello()
myTello.connect()
myTello.takeoff()

myTello.move_up(30)
myTello.move_down(30)

myTello.move_up(30)
myTello.move_down(30)

myTello.move_up(30)
myTello.move_down(30)

myTello.land()
```

1

2

3

함수 정의

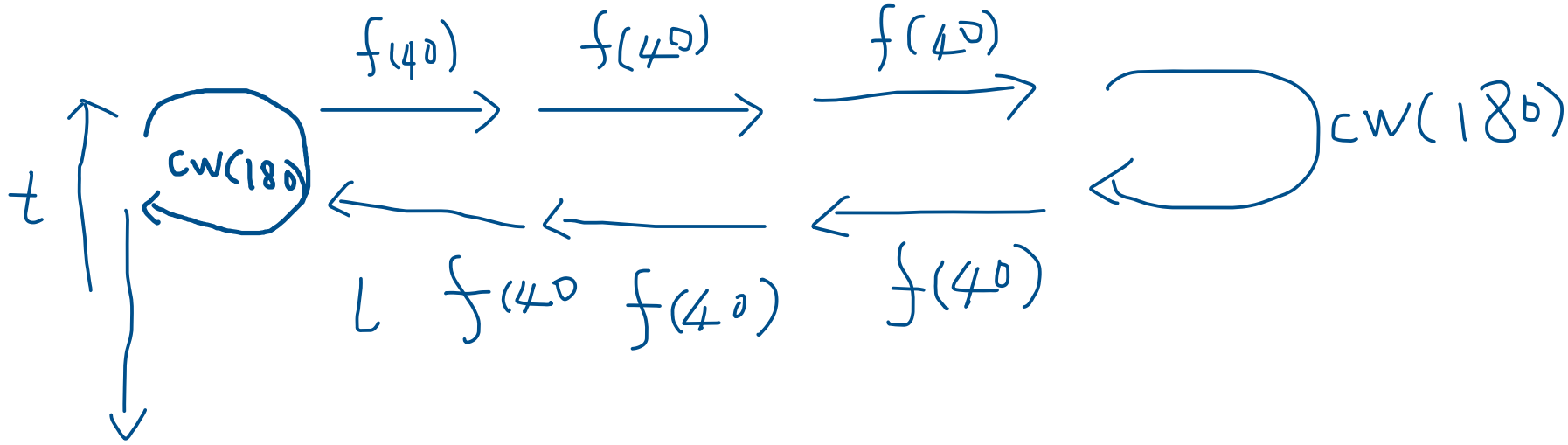
반복문

```
def move_up_down(t):
    myTello.move_up(t)
    myTello.move_down(t)

for i in range(3):
    t = 30
    move_up_down(t)
```

함수 호출

실습 03

- takeoff -> fwd(40) -> fwd(40) -> fwd(40) -> cw(180) -> fwd(40) -> fwd(40) -> fwd(40) -> cw(180) -> land
- fwd(40)을 반복문으로 구현해 보자



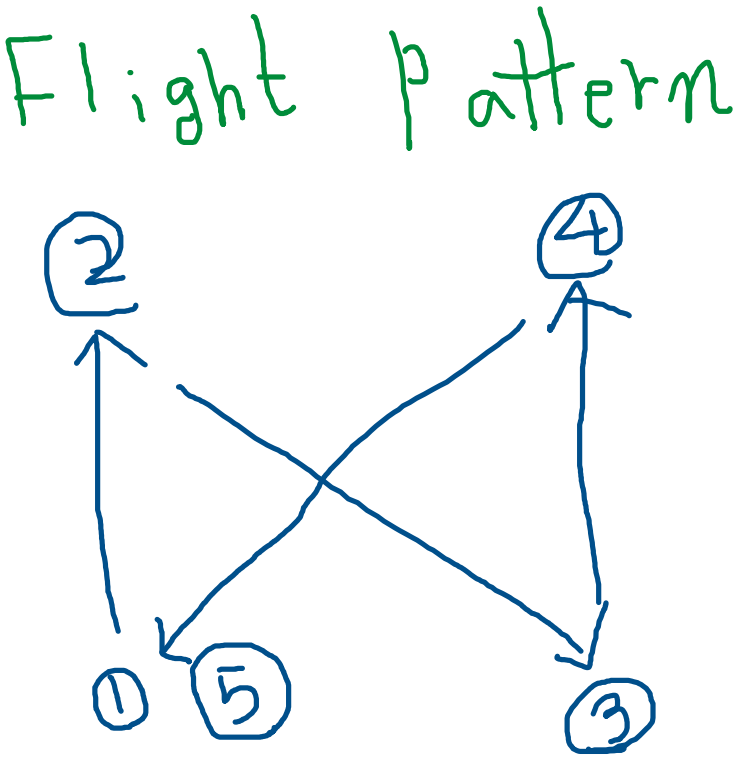
input() 함수를 사용한 드론 제어

```
from djitellopy import tello

myTello = tello.Tello()

myTello.connect()

battery_level = tello.get_battery()
print(battery_level)

while True:
    command = int(input("Enter Command!"))
    print(command, end="\n")

    if (command == 1):
       myTello.takeoff()
    elif (command == 2):
       myTello.move_up(30)
    elif (command == 3):
       myTello.move_down(30)
    elif (command == 4):
       myTello.land()
    else:
       break

print("Drone mission completed!")
```

실습 04 : input() 함수를 이용한 드론 조정기 만들기

1. Takeoff()
2. move_up(20)
3. move_down(20)
4. move_left(20)
5. move_right(20)
6. move_forward(20)
7. move_backward(20)
8. rotate_clockwise(90)
9. rotate_counter_clockwose(90)
10. flip_back()
11. flip_forward()
12. flip_left()
13. flip_right()
14. land()

실습 05 : cross flight 미션



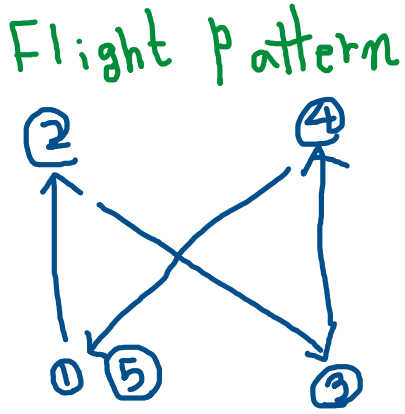
`go_xyz_speed(self, x, y, z, speed)`

Fly to x y z relative to the current position. Speed defines the traveling speed in cm/s.

Parameters:

Name	Type	Description	Default
`x`	`int`	-500-500	*required*
`y`	`int`	-500-500	*required*
`z`	`int`	-500-500	*required*
`speed`	`int`	10-100	*required*

실습 05 : cross flight 미션




```
from djitellopy import Tello
import time

print("Create Tello object")
tello = Tello()

print("Connect to Tello Drone")
tello.connect()

battery_level = tello.get_battery()
print(battery_level)

print("Takeoff!")
tello.takeoff()

time.sleep(1)
```

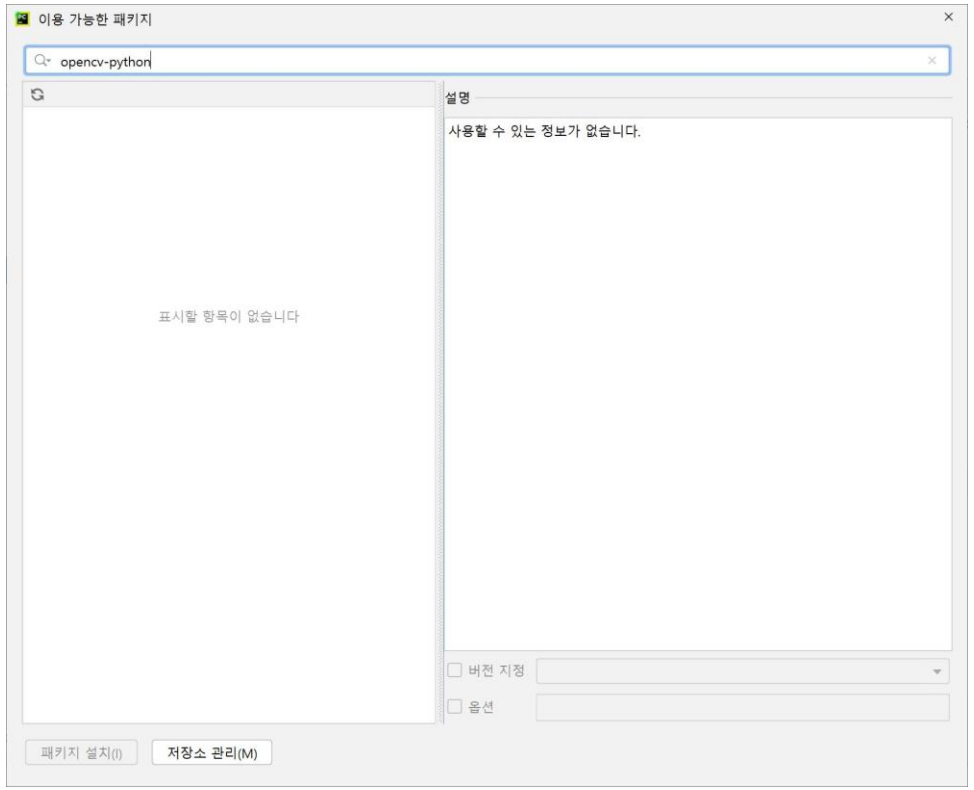
```
travel_distance_cm = 50
#tello.go_xyz_speed(x,y,z, speed)

# x - (+)foward/(-)backwards
# y - (+)left/(-)right
# z - (+)up/(-)down
tello.go_xyz_speed(0, 0, travel_distance_cm, 20)
print("sleep")
time.sleep(0.5)
tello.go_xyz_speed(0, travel_distance_cm, -travel_distance_cm, 20)
print("sleep")
time.sleep(0.5)
tello.go_xyz_speed(0, 0, travel_distance_cm, 20)
print("sleep")
time.sleep(0.5)

# x - (+)foward/(-)backwards
# y - (+)left/(-)right
# z - (+)up/(-)down
tello.go_xyz_speed(0, -travel_distance_cm, -travel_distance_cm, 20)
print("sleep")
time.sleep(0.5)

print("landing")
tello.land()
print("touchdown.... goodbye")
```

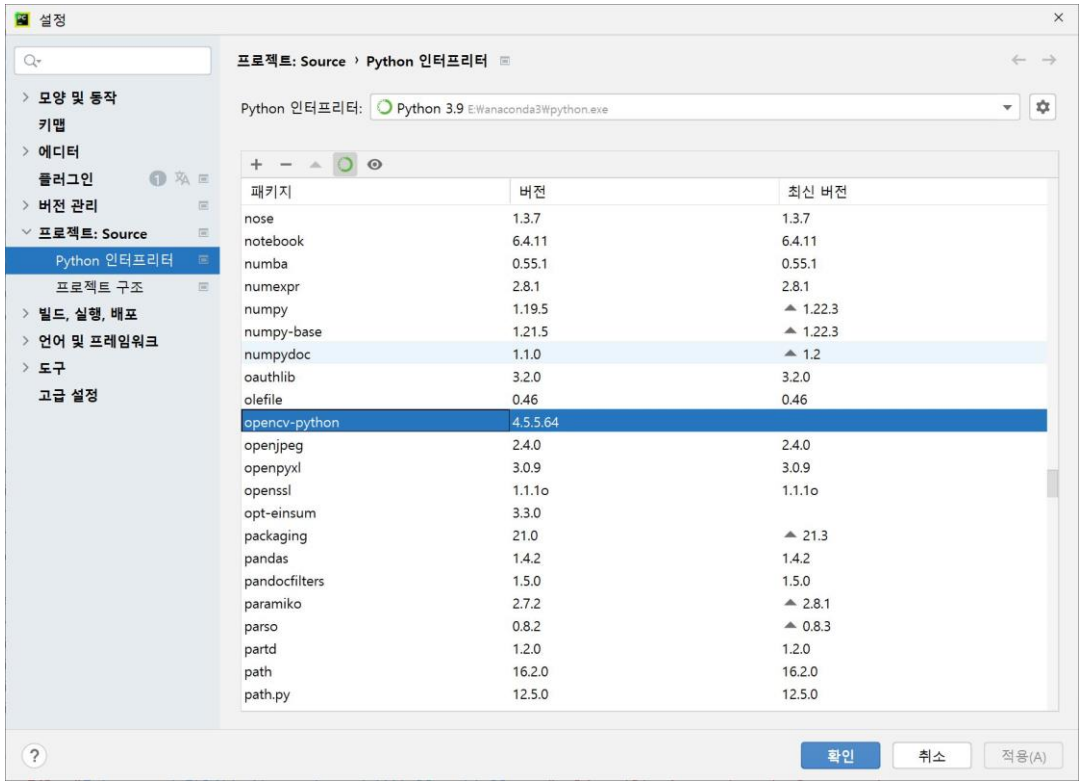
Opencv를 이용한
드론의 이미지, 동영상 처리

https://opencv.org/

파이썬 Opencv 모듈 설치

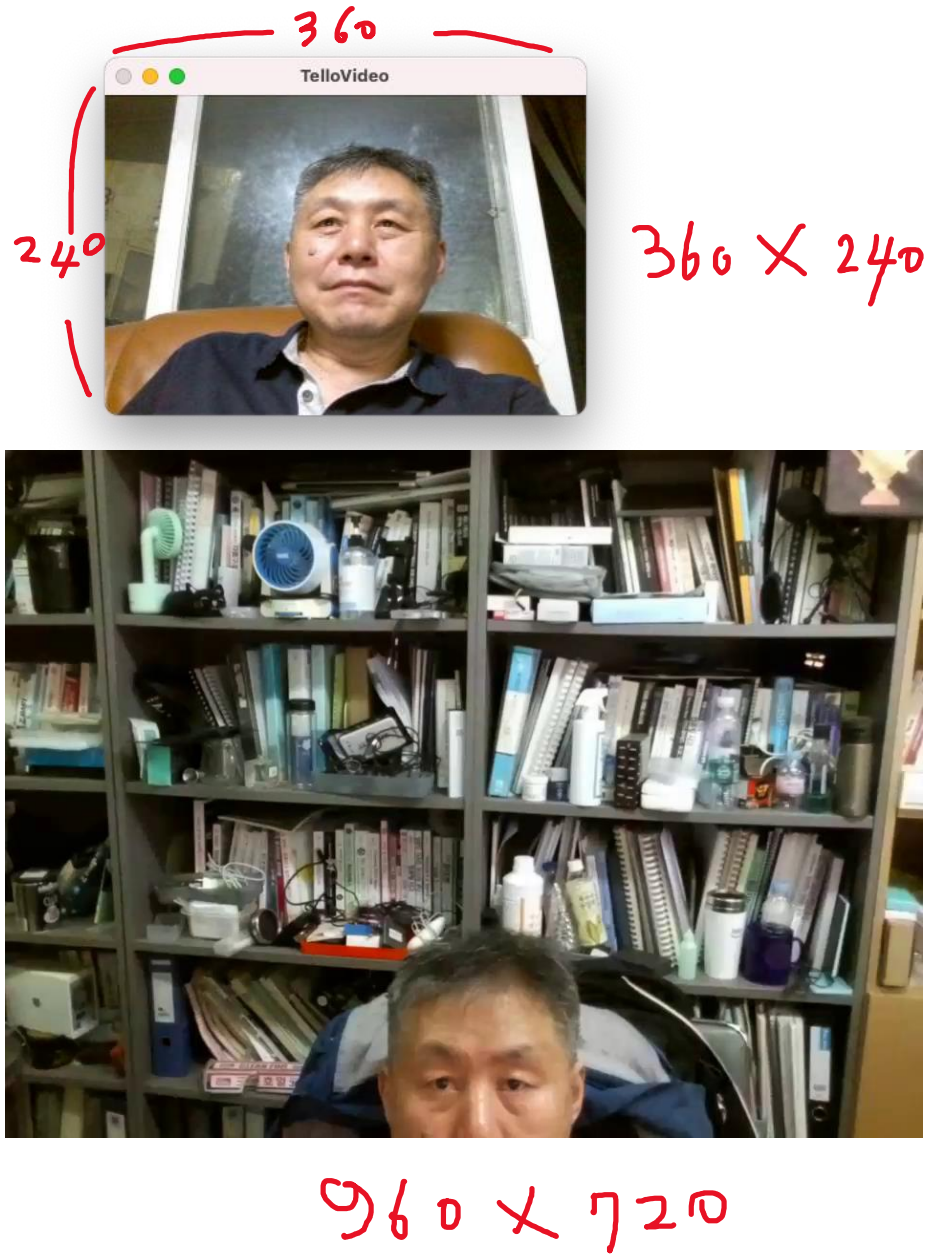
- 파이참
  - 설정 -> 프로젝트 -> Python 인터프리터 -> + -> opencv-python





드론 사진 촬영, 저장하기

```python
import cv2
from djitellopy import Tello

battery_level = tello.get_battery()
print(battery_level)

tello = Tello()
tello.connect()

tello.streamon()
frame_read = tello.get_frame_read()

tello.takeoff()
cv2.imwrite("picture.png", frame_read.frame)

tello.land()
```



Opencv로 드론 비디오 읽기, 보기

```
from djitellopy import tello
import cv2
import time

tello = tello.Tello()
tello.connect()

battery_level = tello.get_battery()
print(battery_level)

print("Turn Video Stream On")
tello.streamon()

time.sleep(2)
```

```
while True:
    # read a single image from the Tello video feed
    frame_read = tello.get_frame_read()
    tello_video_image = frame_read.frame

    # 이미지 크기 조정 " resize()
    image = cv2.resize(tello_video_image, (360, 240))

    # use opencv to write image
    if image is not None:
        cv2.imshow("TelloVideo", image)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

tello.streamoff()
cv2.destroyWindow('TelloVideo')
cv2.destroyAllWindows()
```

드론 비디오 촬영, 전송 : Opencv 키보드 제어

```
from djitellopy import tello
import cv2
import time

tello = tello.Tello()
tello.connect()

battery_level = tello.get_battery()
print(battery_level)

time.sleep(2)
tello.streamon()

frame_read = tello.get_frame_read()
```

```
while True:

    frame = frame_read.frame
    frame = cv2.resize(frame, (360*2, 240*2)) # 이미지 크기 조정

     while True:
         frame = frame_read.frame
         if frame is not None:
             cv2.imshow("TelloVideo", frame)
             k = cv2.waitKey(5) & 0xFF
             if k == ord('q'):
                 break
             elif k == ord('t'):
                 tello.takeoff()
             elif k == ord('u'):
                 tello.move_up(50)
             elif k == ord('c'):
                 tello.rotate_clockwise(360)
             elif k == ord('l'):
                 tello.land()
             else :
                 print('no key!!!')

tello.streamoff()
cv2.destroyWindow('TelloVideo')
cv2.destroyAllWindows()
```

Opencv로 드론 키보드 제어하기

- manual-control-opencv

```
from djitellopy import Tello
import cv2, time

tello = Tello()
tello.connect()
battery_level = tello.get_battery()
print(battery_level)

tello.streamon()
frame_read = tello.get_frame_read()
frame = cv2.resize(frame, (360*2, 240*2))
```

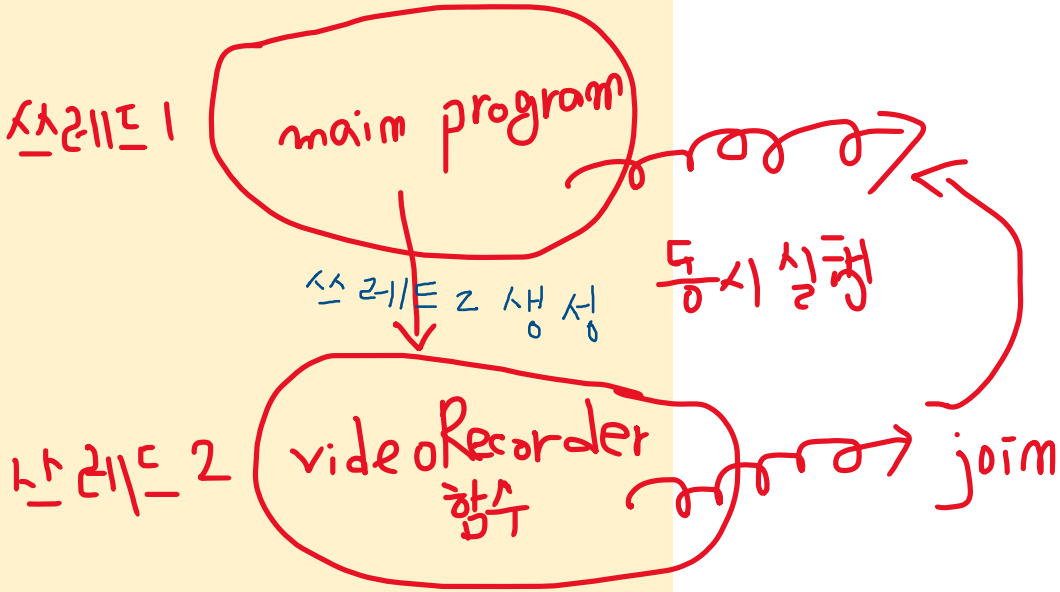
- 소스 코드 : https://github.com/damiafuentes/DJITelloPy/blob/master/examples/manual-control-opencv.py

```
while True:
    img = frame_read.frame
    cv2.imshow("drone", img)

    key = cv2.waitKey(1) & 0xff
    if key == ord('q'):
        break
    elif key == ord('t'):
        tello.takeoff()
    elif key == ord('f'):
        tello.move_forward(30)
    elif key == ord('b'):
        tello.move_back(30)
    elif key == ord('l'):
        tello.move_left(30)
    elif key == ord('r'):
        tello.move_right(30)
    elif key == ord('c'):
        tello.rotate_clockwise(30)
    elif key == ord('r'):
        tello.rotate_counter_clockwise(30)
    elif key == ord('u'):
        tello.move_up(30)
    elif key == ord('d'):
        tello.move_down(30)

tello.streamoff()
cv2.destroyWindow('drone')
cv2.destroyAllWindows()
```

드론이 촬영/저장 하면서 동시에 비행하기 : 쓰레드(Thread)

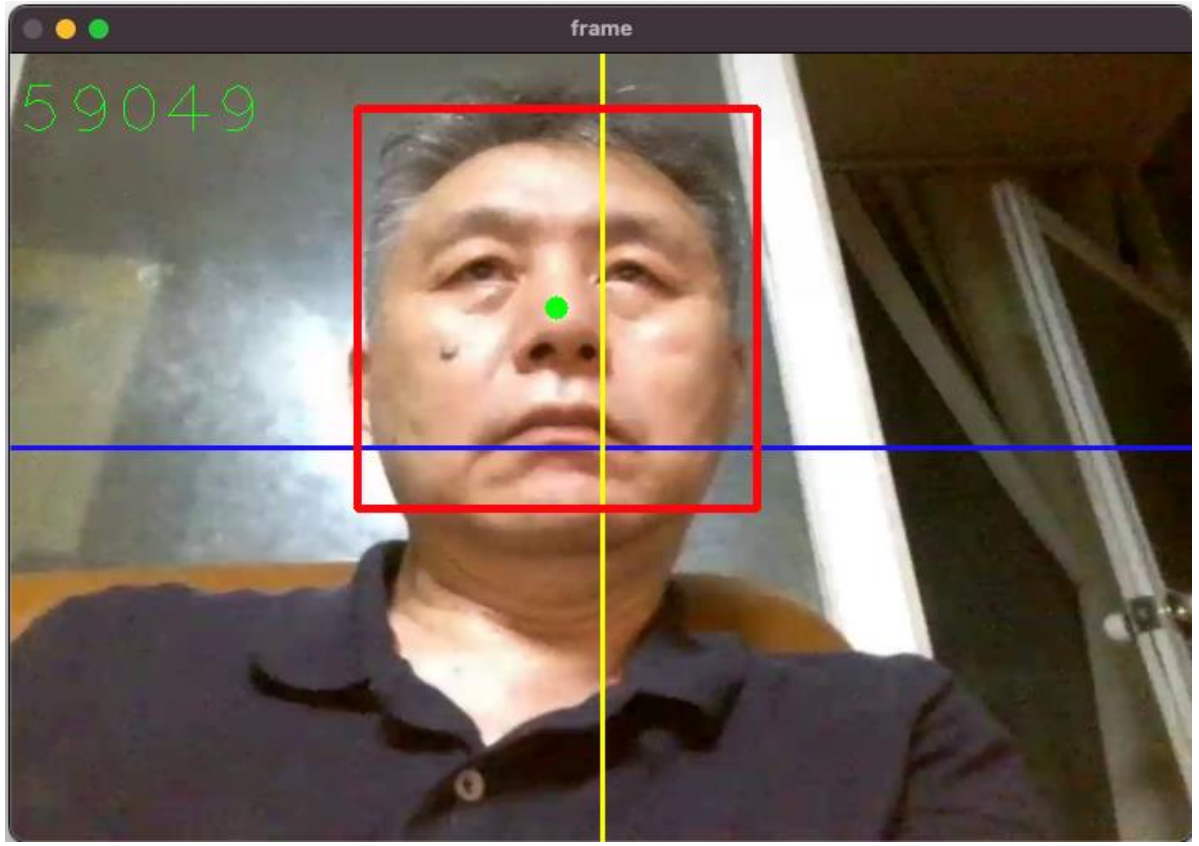
```
import time, cv2
from threading import Thread
from djitellopy import Tello

tello = Tello()
tello.connect()
battery_level = tello.get_battery()
print(battery_level)

keepRecording = True

tello.streamon()
frame_read = tello.get_frame_read()
```

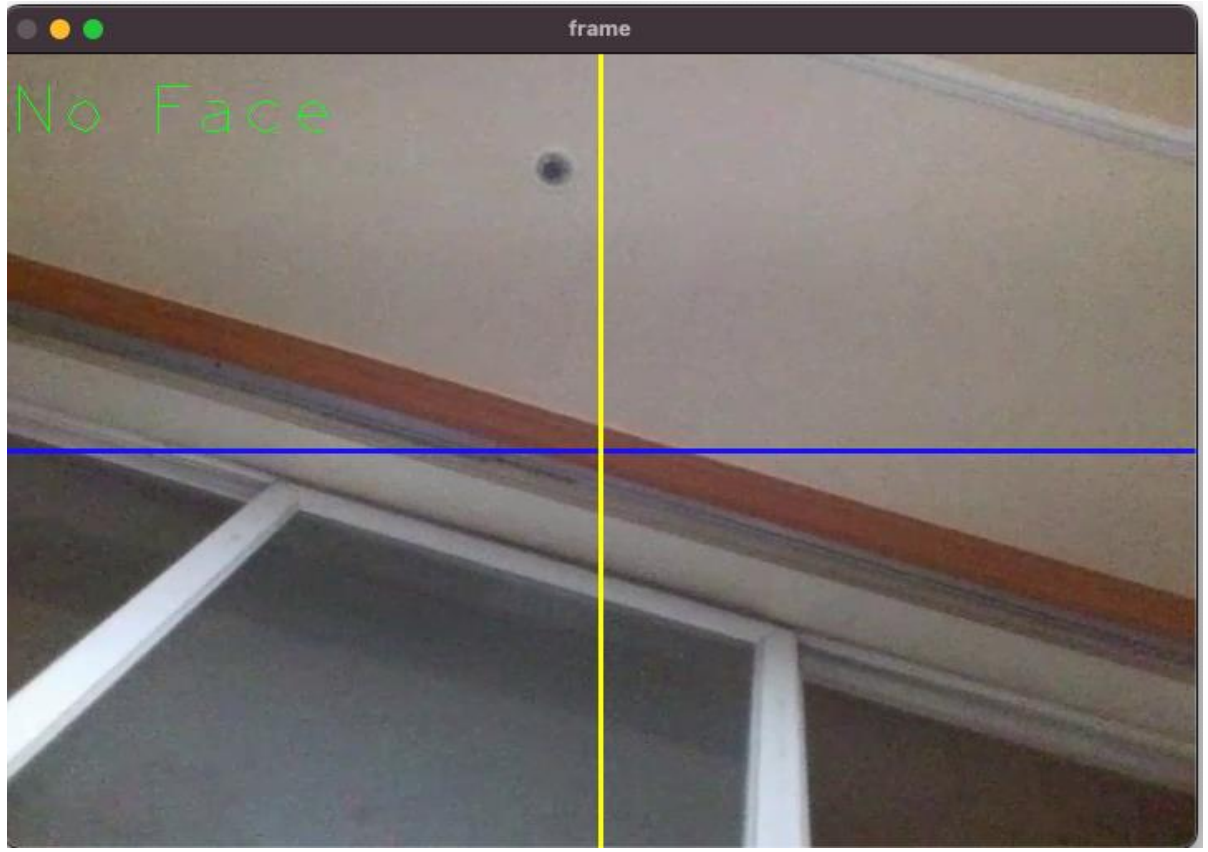
```
def videoRecorder():
    height, width, _ = frame_read.frame.shape

    # create a VideoWrite object, recoring to ./video.avi
    video = cv2.VideoWriter('video3.avi', cv2.VideoWriter_fourcc(*'XVID'), 30, (width, height))

    while keepRecording:
        tello_video_image = frame_read.frame
        tello_video_image = cv2.resize(tello_video_image, (360 * 2, 240 * 2))

        # PC에 Tello 촬영 동영상 저장
        video.write(frame_read.frame)

        if tello_video_image is not None:
            cv2.imshow('video', tello_video_image)
            time.sleep(1 / 30)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            cv2.destroyWindow('video')
            cv2.destroyAllWindows()

# we need to run the recorder in a seperate thread
recorder = Thread(target=videoRecorder)
recorder.start()

tello.takeoff()
tello.move_up(30)
tello.rotate_counter_clockwise(360)
tello.land()

keepRecording = False
recorder.join()
```

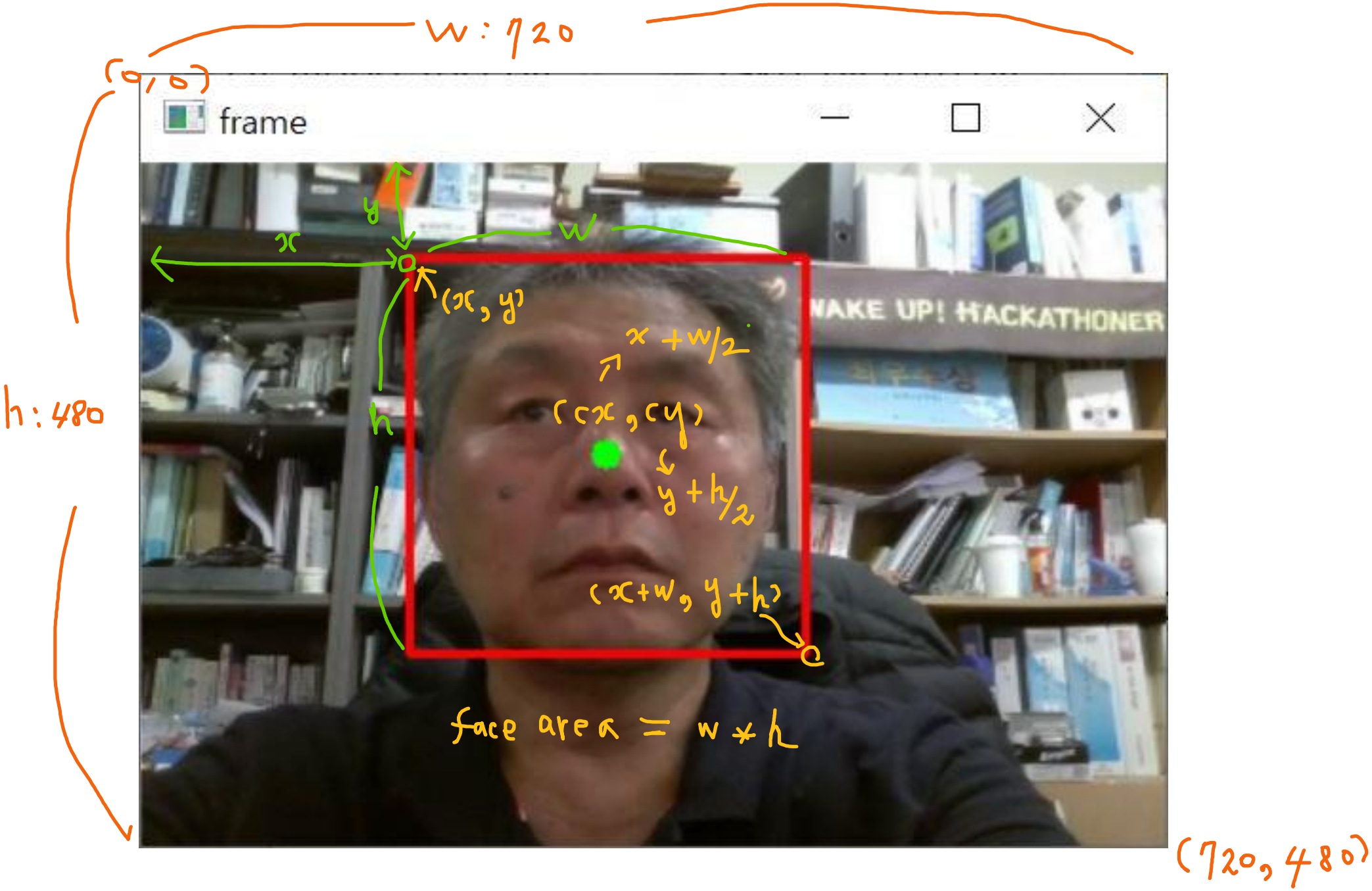


- 소스 코드 다운로드 : https://drive.google.com/file/d/1jZD-DjCZK8cVg1r-20hhr4-RcYzW9-5-/view?usp=sharing

얼굴 추적 드론의 원리를 생각해 봅시다



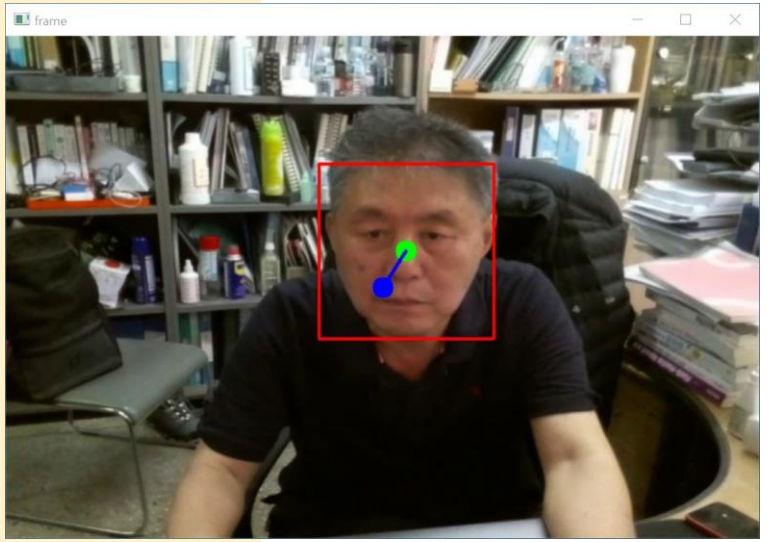


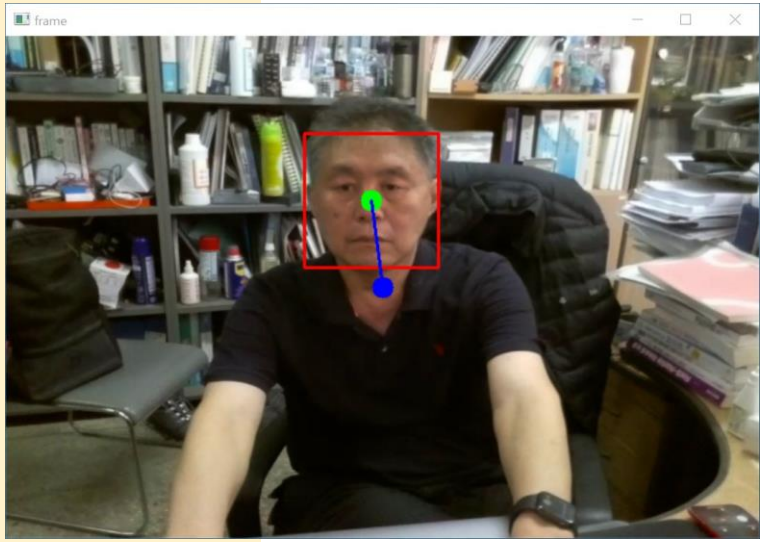

w : 720
(0, 0)
frame
y
x
w
(x, y)
x + w/2
(cx, cy)
y + h/2
h
h : 480
(x+w, y+h)
face area = w * h
(720, 480)

얼굴 인식하기 : Cascade Classifier

```
import cv2
from djitellopy import tello

tello = tello.Tello()
tello.connect()
battery_level = tello.get_battery()
print(battery_level)
tello.streamon()
```

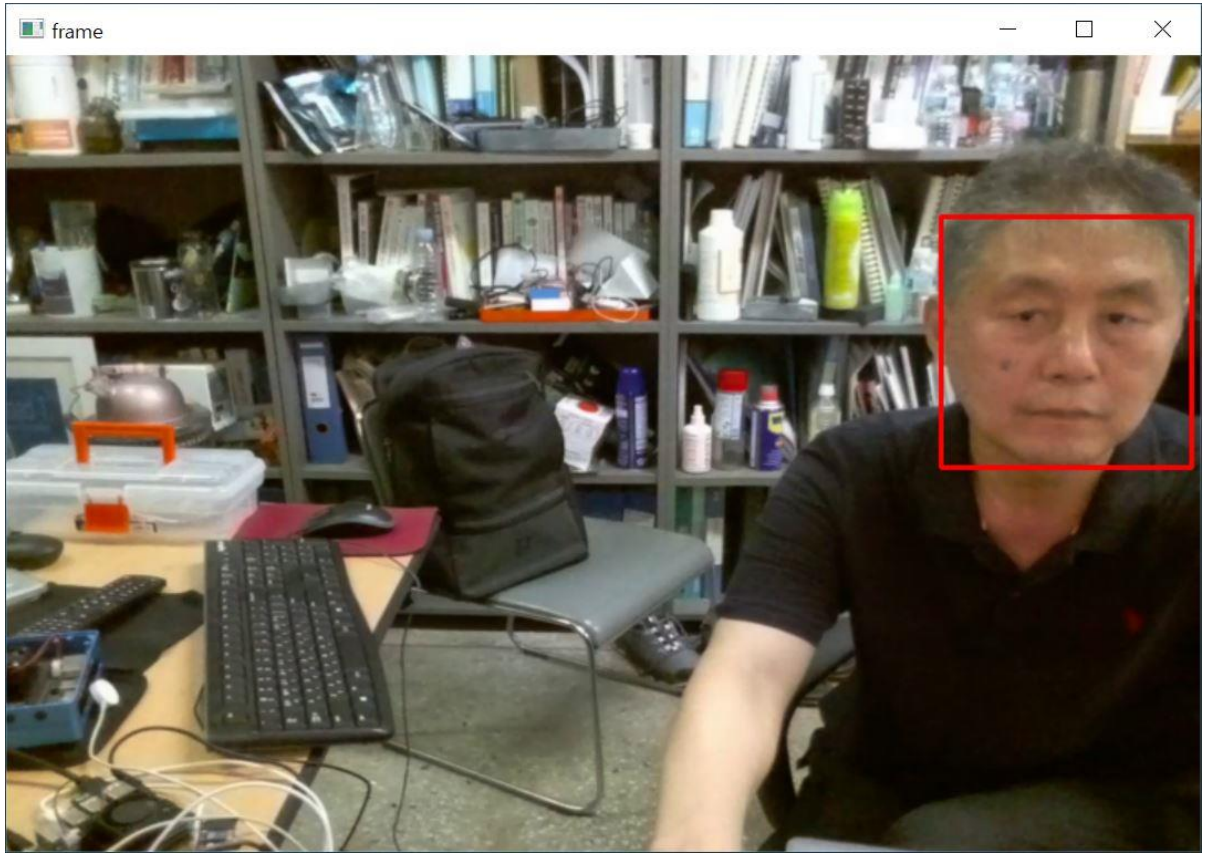
```
while True:
    img = tello.get_frame_read().frame
    img = cv2.resize(img, (360*2, 240*2))
    faceCascade = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
    imgGray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = faceCascade.detectMultiScale(imgGray, 1.3, 5)

    if len(faces) == 1:
        print("face found")
        print('faces', faces)
    else:
        print("face not found")

    # 전체 화면의 중심점(center)
    tcx = 360
    tcy = 240
    cv2.circle(img, (tcx, tcy), 10, (255, 0, 0), cv2.FILLED)

    # 인식된 얼굴 사각형 박스 그리기
     for (x, y, w, h) in faces:
        cv2.rectangle(img, (x, y), (x + w, y + h), (0, 0, 255), 2)

        cx = x + w // 2
        cy = y + h // 2
        area = w * h
        cv2.circle(img, (cx, cy), 10, (0, 255, 0), cv2.FILLED)
        print('area =', area)

        # 전체 이미지 중심과 얼굴 이지미 중심 간의 선 그리기기
          cv2.line(img, (cx, cy), (tcx, tcy), (255, 0, 0), 2)

    cv2.imshow('frame',img)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

tello.streamoff()
cv2.destroyAllWindows()
```





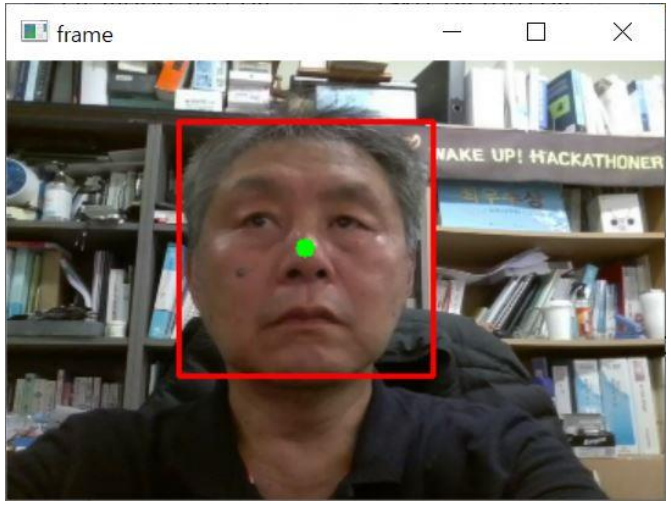
얼굴 인식하기 : Cascade Classifier

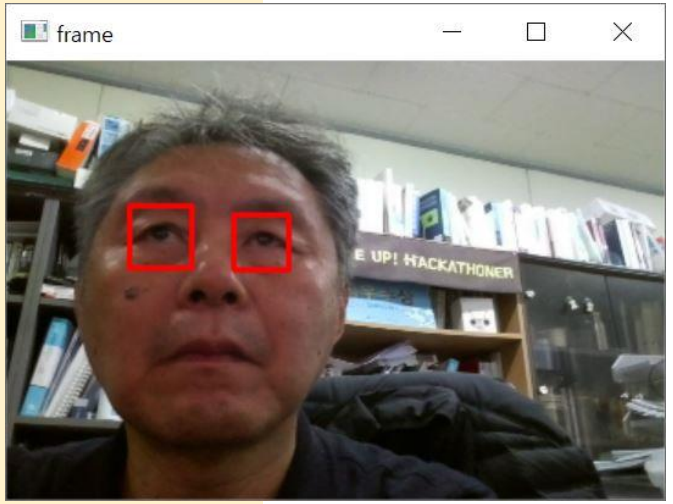
```
import cv2
from djitellopy import tello

tello = tello.Tello()
tello.connect()
battery_level = tello.get_battery()
print(battery_level)
tello.streamon()

while True:
    img = tello.get_frame_read().frame
    img = cv2.resize(img, (360*2, 240*2))
    faceCascade = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")

    imgGray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = faceCascade.detectMultiScale(imgGray, 1.3, 5)

    for (x, y, w, h) in faces:
        cv2.rectangle(img, (x, y), (x + w, y + h), (0, 0, 255), 2)

    cv2.imshow('frame',img)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

tello.streamoff()
cv2.destroyAllWindows()
```

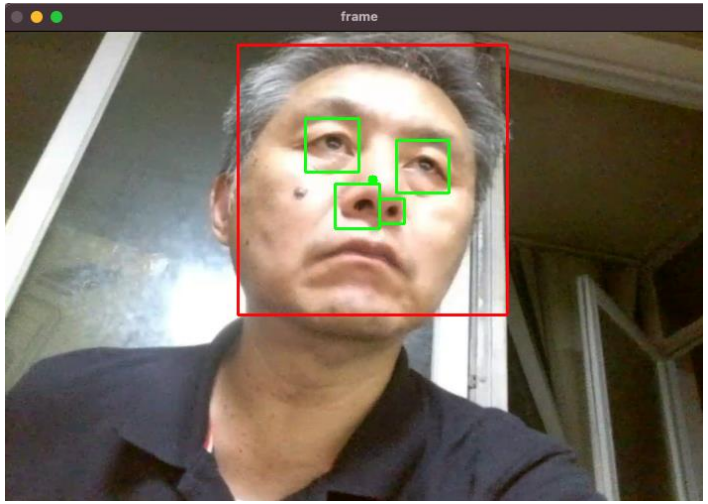


얼굴 인식하기 : Cascade Classifier

```
import cv2
from djitellopy import tello

tello = tello.Tello()
tello.connect()
battery_level = tello.get_battery()
print(battery_level)
tello.streamon()

while True:
    img = tello.get_frame_read().frame
    img = cv2.resize(img, (360, 240))
    faceCascade = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
    imgGray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = faceCascade.detectMultiScale(imgGray, 1.2, 8)

    for (x, y, w, h) in faces:
        cv2.rectangle(img, (x, y), (x + w, y + h), (0, 0, 255), 2)
        cx = x + w // 2
        cy = y + h // 2
        area = w * h
        cv2.circle(img, (cx, cy), 5, (0, 255, 0), cv2.FILLED)
        print('area =', area)

    cv2.imshow('frame',img)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cv2.destroyAllWindows()
```







* 소스 코드 : https://drive.google.com/file/d/1jZLuuovY-tWJdwy1LYD4knpCQkvAjzxk/view?usp=sharing
* haarcascade_frontalface_default.xml 파일 : https://drive.google.com/file/d/1j_Hjo0N6v0HtL1F55ekFBbQ6yxKT3SrQ/view?usp=sharing