

Drone Object Classification

Yonsei Univ, Intel, G-Market

http://www.anaconda.com/download/



[Products](#) [Solutions](#) [Resources](#) [Partners](#) [Company](#)

 [Free Download](#)

[Sign In](#)

Distribution

Register to get everything you need to get started on your workstation

- ✓ Distribution installation on Windows, MacOS, or Linux
- ✓ Easily search and install thousands of data science, machine learning, and AI packages
- ✓ Manage packages and environments from a desktop application or work from the command line
- ✓ Deploy across hardware and software platforms

Commercial use at a company of more than 200 employees requires a Business or Enterprise license. [See Pricing](#)

Provide email to download Distribution

Don't miss out! Get access to: Cloud Notebooks, Anaconda Assistant, easy application deployment, learning resources, and updates from Anaconda.

Email Address:

- ☐ I agree to receive communication from Anaconda regarding relevant content, products, and services. I understand that I can revoke this consent [here](#) at any time.

By continuing, I agree to Anaconda's [Privacy Policy](#) and [Terms of Service](#).

[Submit](#) >

[Skip registration](#)

Hey! 🐼 Welcome back. Can I help you find something today?



Download Now


For installation assistance, refer to [Troubleshooting](#).


Download Distribution by choosing the proper installer for your machine.


 [Download](#)



 Home

 Environments

 Learning

 Community

NEW

Anaconda Toolbox
Supercharged local notebooks.
Click the Toolbox tile to Install.

[Read the Docs](#)

[Documentation](#)

[Anaconda Blog](#)



All applications ▾

on

base (root) ▾

Channels



PyCharm Professional

The Python IDE for data science. It combines the interactivity of Jupyter notebooks with intelligent Python coding assistance, Anaconda support, and scientific libraries.

Install



Anaconda Toolbox

4.0.15

Anaconda Assistant

JupyterLab supercharged with a suite of Anaconda extensions, starting with the Anaconda Assistant AI chatbot.

Launch



Anaconda Cloud Notebooks

Cloud-hosted notebook service from Anaconda. Launch a preconfigured environment with hundreds of packages and store project files with persistent cloud storage.

Launch



CMD.exe Prompt

0.1.1

Run a cmd.exe terminal with your current environment from Navigator activated

Launch



JupyterLab

4.0.11

An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Launch




Notebook


7.0.8

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Launch

+ 신규

▼  내 드라이브

- ▶  Backup

► Classroom

▼ Colab Notebooks

▼ data

```

▶ bin

```

► train


► val

► data2

►  fpga

▶  HelloFax

►  Lecture

-  nump

- ▶  smart_gate

내 드라이브 > Colab Notebooks > data ▾

✕ 1개 선택됨 👤 ⬇️ 📁 🗑️ 🔗 ⋮

 단축키 업데이트 2024년 8월 1일 목요일부터 첫 글자를 탐색할 수 있도록 Drive 단축키가 업데이트됩니다. 자세히 알아보기

단축키

더불어

 val

train

bin

파일

 result.txt

[illegible] mobilenet_model....


The diagram illustrates the MobileNetV2 architecture, showing the flow of data through various layers. The input is a 224x224x3 image, which is processed by an InputLayer to produce a 224x224x3 output. This output is then processed by a Functional layer to produce a 7x7x1280 output. The output is then processed by a GlobalAveragePooling2D layer to produce a 1x1x1280 output. Finally, the output is processed by a Dense layer to produce a 512 output.


```

graph TD
    Input["mobilenetv2_1.00_224_input  
input: [(None, 224, 224, 3)]"] --> InputLayer["InputLayer  
output: [(None, 224, 224, 3)]"]
    InputLayer --> Functional["mobilenetv2_1.00_224  
Functional  
input: (None, 224, 224, 3)  
output: (None, 7, 7, 1280)"]
    Functional --> GlobalAvgPool["global_average_pooling2d  
GlobalAveragePooling2D  
input: (None, 7, 7, 1280)  
output: (None, 1280)"]
    GlobalAvgPool --> Dense["dense  
Dense  
input: (None, 1280)  
output: (None, 512)"]
  
```

mobilenetv2_1.00_224_input	input:	[(None, 224, 224, 3)]
InputLayer	output:	[(None, 224, 224, 3)]
↓		
mobilenetv2_1.00_224	input:	(None, 224, 224, 3)
Functional	output:	(None, 7, 7, 1280)
↓		
global_average_pooling2d	input:	(None, 7, 7, 1280)
GlobalAveragePooling2D	output:	(None, 1280)
↓		
dense	input:	(None, 1280)
Dense	output:	(None, 512)

연결 앱

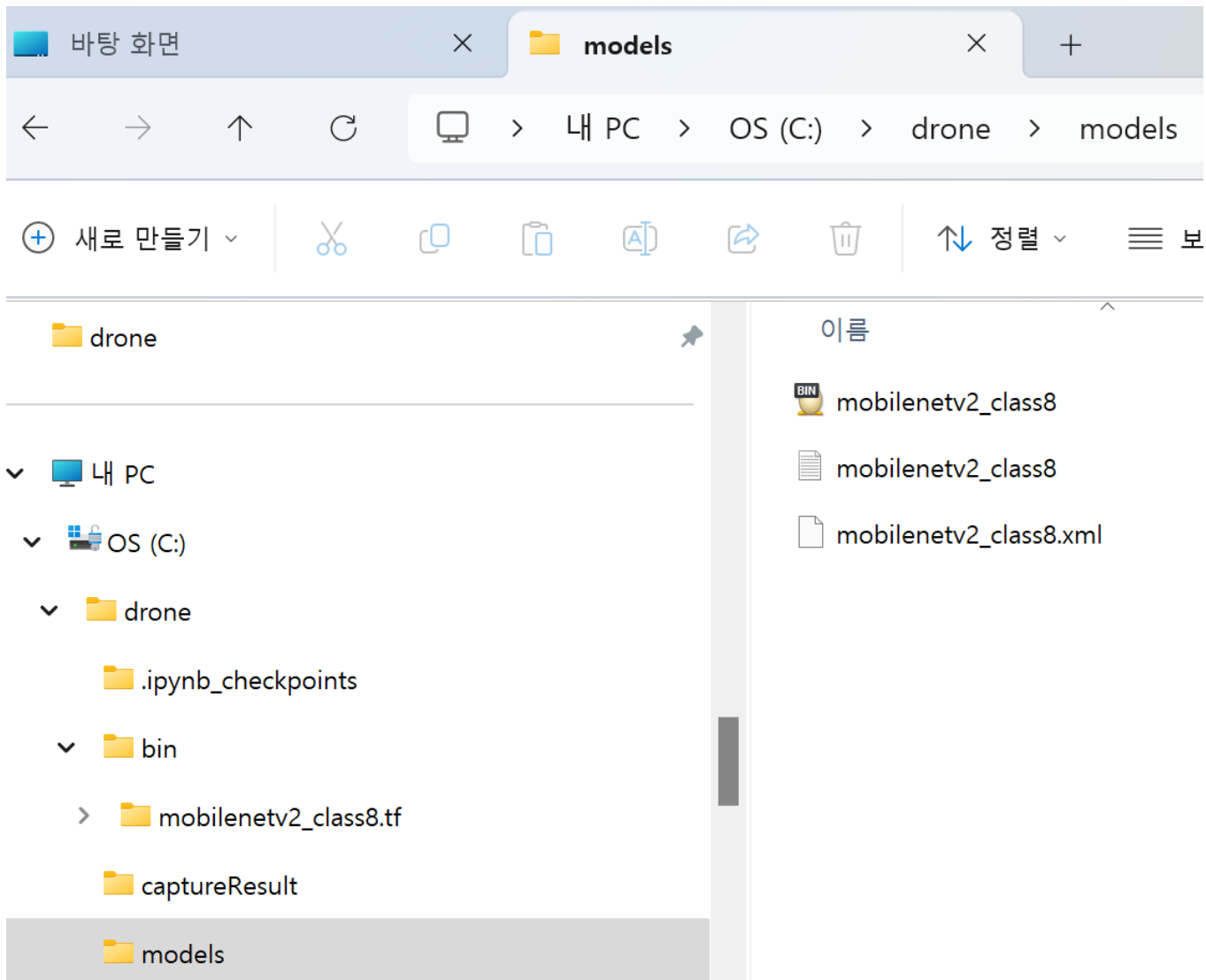
 다운로드

 이름 바꾸기
$$D^+ \quad \frac{F}{O} \frac{O}{F}$$

📁 정리

① 폴더 정보

휴지통으로 이동



drone_control.py

drone_control은 드론이 학습시킨 모델로 설정한 사물을 인식하면 script 대로 drone을 제어하는 코드.

추론할 사물 이름 넣어주기 <중요!!!>

C:\drone\modelss 에서 mobilenet_class8v2.txt 파일을 수정합니다

현재 mobilenet_class8v2.txt에는 강의에 사용한 이름 8개가 들어 있습니다. 대회에서 사용하는 이름으로 추후 변경해야 합니다. *알파벳 순서로 넣는 것이 매우* 중요함.

drone_control.py 사용법

target 설정하기

1. drone_control.py 에 들어가서 ###드론이 원하는 사물 발견하면 script대로 움직이기###를 찾습니다. (234번째 line)
2. 드론이 어떤 사물을 발견했을 때 움직이게 하고 싶은지 결정하고 그 사물을 target 1, 2, 3에 적어줍니다. 이 때 이름은 앞 강좌에던 사물 이름 그대로 사용합니다.
3. 타겟을 하나 또는 두 개만 하고 싶을 때는 맨 앞에 #을 붙여주어 주석처리를 해줍니다. 이 때 첫번째, 두번째, 세번째 타겟을 주석처리 함에 따라 그 아래 해당하는 부분도 함께 주석처리를 해주어야 합니다. drone_control.py 에는 타겟과 모두 주석처리해 준 상태입니다.

label list 수정하기

4. 83번째 line으로 올라가서 label list에 내가 정한 target 이름을 적어주고 그 개수만큼 제거하여 총 8개로 맞추어 준다. 이 부분은 순서 상관없이 그냥 아무거나 지우시고 원하는 target 을 넣어준다.

```
self.picLabels = ['aeroplane', 'bicycle', 'bird', 'boat', 'bottle', 'bus', 'car', 'cat']
```

scripts 작성하기

5. 드론이 사물을 발견했을 때 움직이는 motion은 scripts에 txt 파일로 저장합니다. 이 때 스크립트 이름은 꼭 사물과 동일한 이름으로 해주어야 합니다.

6. 위의 스크립트를 작성할 때 직접 txt 파일에 drone_control.py 를 실행한 후 reset 버튼 아래 스크립트를 적는 칸에 명령어들을 적고 제목을 사물 이름으로 한 후 save 버튼을 눌러도 저장이 됩니다.

drone_control.py 에서 사물이름과 스크립트 작성이 완료된다면 잘 작동됩니다.

Tips

drone_control.py 를 실행하고 GUI 가 뜨면 드론 카메라에 사물을 비춰줍니다. 그럼 83번째 line 에 적힌 사물이라면 GUI 오른쪽 이미지가 방금 비춰진 사물로 update 됩니다. 사물을 인지하면 script 대로 추가 action 을 준 target 이었다면, 사진이 update 되면서 script 대로 추가 action 이 일어납니다. 혹시 여기서 추가 action 이 일어나지 않고 '이미 실행중인 오더가 있습니다.' 라고 뜬다면, GUI 의 RESET 버튼을 눌러주면, 방금 update 된 사진이 다시 원래의 이미지로 돌아가면서 다시 인식할 수 있습니다.

GUI 오른쪽 8개 사물 사진 변경하기

- GUI 를 실행하면 오른쪽에 8개 사물 사진이 뜹니다. 이 사진을 내가 훈련시킨 사물의 사진으로 변경하고 싶을 때는 아래의 대로 적용해야 합니다. 먼저 8개 사물 사진은 res폴더에 1. ~ 8. 까지 사진 8장을 볼 수 있다. 여기서 원하는 사진여 넣어준다. 이 때 사진의 크기는 100 * 100 으로 해주어야 나중에 GUI 에 잘 나타남. 사진을 넣은 후 이름을 변경해주어야 하는데, 왼쪽부터 아래로 갈수록 1번부터 8번입니다. 즉 비행기가 1번, 자전거가 2번, 사람이 8번 순으로 GUI 에 사진이 나타남.
- 비행기 대신 추가하려는 사물이 꽃이라면, 꽃의 사진을 res 폴더 밑에 넣어줍니다. 1. ~ 8. 의 사진 외에 앞에 '.' 이 붙은 '.1. ~ .8.' 의 사진들을 추가로 더 확인할 수 있습니다. 앞에 '.' 이 하나 더 붙은 파일들이 실제 GUI 에서 볼 수 있는 사진입니다. 따라서 원래의 '.1.' 을 지우고, 꽃의 사진 이름을 '.1.' 로 변경해주어 최종적으로 GUI 의 사진을 바꿀 수 있다. 사진이 변경되지 않으면 기존의 .1~.8을 지우고 시도해야 함.

드론 인터페이스 실행하기

```
python3 drone_control.py True
```


Microsoft Windows [Version 10.0.22631.3593]
(c) Microsoft Corporation. All rights reserved.

C:\Users\easyh>conda create -n drone_env python=3.11

Channels:

- defaults

Platform: win-64

Collecting package metadata (repodata.json): done

Solving environment: done

C:\Users\easyh>conda activate drone_env

(drone_env) C:\Users\easyh>

(drone_env) C:\Users\easyh>pip install -q "openvino>=2023.1.0"

(drone_env) C:\Users\easyh>pip install openvino-dev

Collecting openvino-dev

Using cached openvino_dev-2024.1.0-15008-py3-none-any.whl.metadata (16 kB)

Collecting defusedxml>=0.7.1 (from openvino-dev)

Downloading defusedxml-0.7.1-py2.py3-none-any.whl.metadata (32 kB)

Collecting networkx<=3.1.0 (from openvino-dev)

Downloading networkx-3.1-py3-none-any.whl.metadata (5.3 kB)

```
cd C:/drone
```

```
conda create -n drone python=3.11
```

```
conda activate drone
```

```
pip install -q "openvino>=2023.1.0"
```

```
pip install openvino-dev
```

```
pip install PyQt5
```

```
pip install opencv-python
```

```
pip install tello
```

```
python drone_control True
```



TELLO-FD0772

개방형

연결하는 중

취소

Yonsei-Intel Drone Hackathon



intel



Gmarket



0 분 0 초 0

0 초 0



0 분 0 초 0

0 초 0



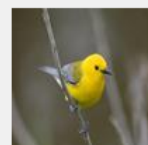
0 분 0 초 0

0 초 0



0 분 0 초 0

0 초 0



0 분 0 초 0

0 초 0



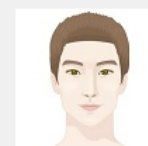
0 분 0 초 0

0 초 0



0 분 0 초 0

0 초 0



0 분 0 초 0

0 초 0

RESET



0 분 0 초 0

0 초 0

0

IP
updated
:
192.168.
10.2

START

PAUSE

STOP

Save

Load

Drone info

Setting

My IP Address 192.168.10.2

Height(Up,Down) 20 CM CCW,CW 45 °

☒ Move(↑,↓,←,→) 20 CM

☐ Forward,Back(↑,↓) 20 CM Left,Right(←,→) 20 CM

Mode1

Mode2

CCW



Take Off

Up

Connect



Land



Down

CW

```
font = QtGui.QFont()
font.setPointSize(10)
self.droneinfo = QtWidgets.QWidget()
self.droneinfo.setObjectName("droneinfo")
self.droneinfo.setFont(font)
```

드론 디자인

[Qt Designer and Python: Build Your GUI Applications Faster – Real Python](http://realpython.com/qt-designer-python/)
<http://realpython.com/qt-designer-python/>

