

YOLO 레이블링 가이드

개요

윈도우즈 환경에서 레이블링 툴을 설치하고, 레이블링 하는 방법을 설명한다.

강아지와 고양이 영상의 데이터로 예를 들어 설명한다.

그리고 학습에 사용되는 코드는 <https://github.com/dhrim/keras-yolo3>의 것을 기준으로 한다.

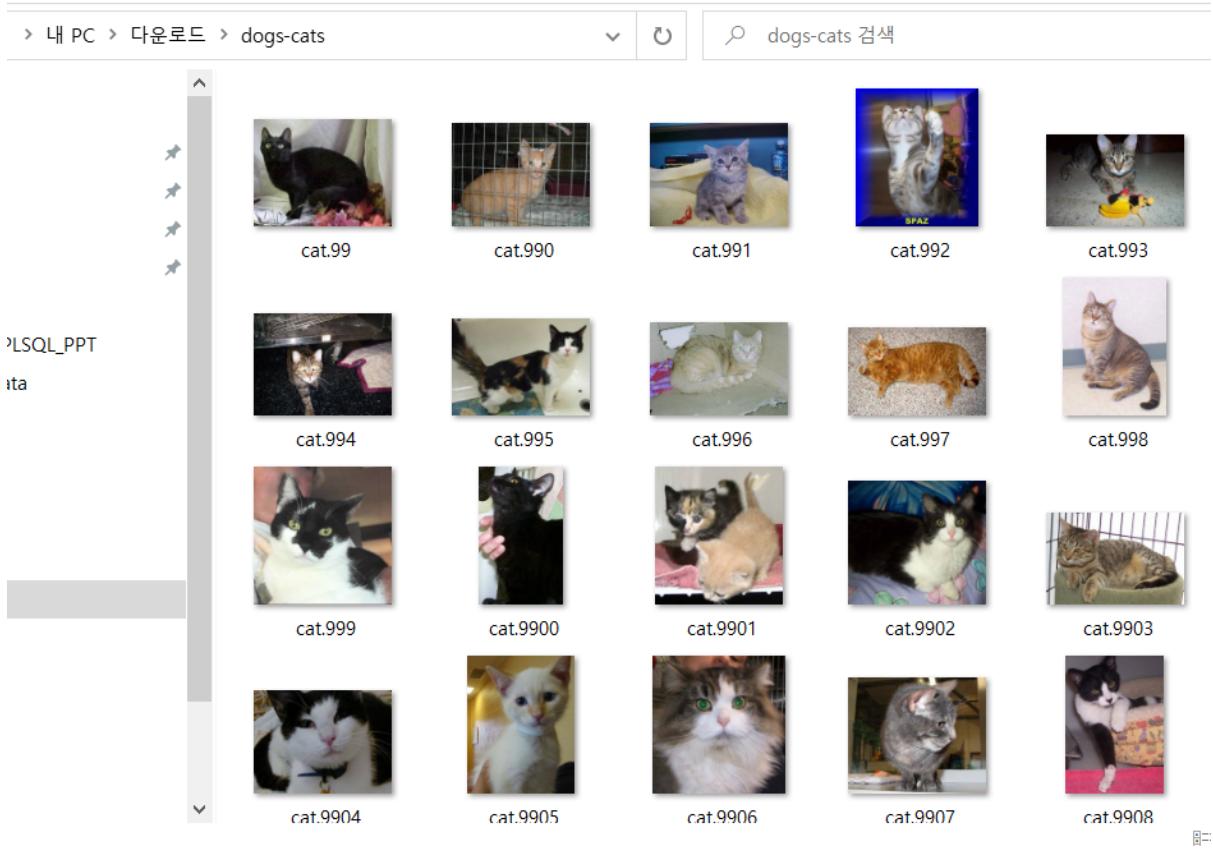
대상 데이터 다운로드

웹브라우저에 다음 url을 입력하고 zip파일을 다운 받는다.

https://github.com/dhrim/hongik_2020/raw/master/material/deep_learning/dogs-cats.zip

압축을 푼다.

풀린 위치는 내PC > 다운로드 > dogs-cats이다.



레이블링 프로그램 설치

windows

<https://github.com/tzutalin/lablelImg>에서 windows_v1.8.1.zip 파일을 다운로드

	libs	use enum for label file formats	20 days ago
	requirements	Update README.md and create a folder f...	2 years ago
	resources	Update zh translation and format code a l...	2 months ago
	tests	By default, use py3 qt5 to build	15 months ago
	.gitignore	Update Gitignore / .DS_Store Removal	last month
	.travis.yml	removed python2 build from travis.yml a...	3 months ago
	CONTRIBUTING.rst	Create new files for pypackage	3 years ago
	HISTORY.rst	typo	13 months ago
	LICENSE	Create new files for pypackage	3 years ago
	MANIFEST.in	Add missing files	3 years ago
	Makefile	Update README.md and setup.py	15 months ago
	README.rst	updated anaconda installation instructions	20 days ago
	__init__.py	Create new files for pypackage	3 years ago
	combobox.py	Fixed import error for PyQt4	7 months ago

python2 python3
 annotations tools
 imagenet deep-learning
 detection recognition
 image-classification

Readme

MIT License

Releases 23

Binary v1.8.1 Latest
 on 3 Dec 2018

+ 22 releases

Packages

No packages published

압축을 푼다.



Mac

```
1$ sudo pip3 install labelImg
2
3$ which labelImg
4/usr/local/bin/labelImg
5
6# 실행
7$ /usr/local/bin/labelImg
```

클래스 이름 파일 작성

windows_v1.8.1/data/ 폴더의 predefined_classes.txt 파일을 내용을 다음으로 작성

```
1dog
2cat
```

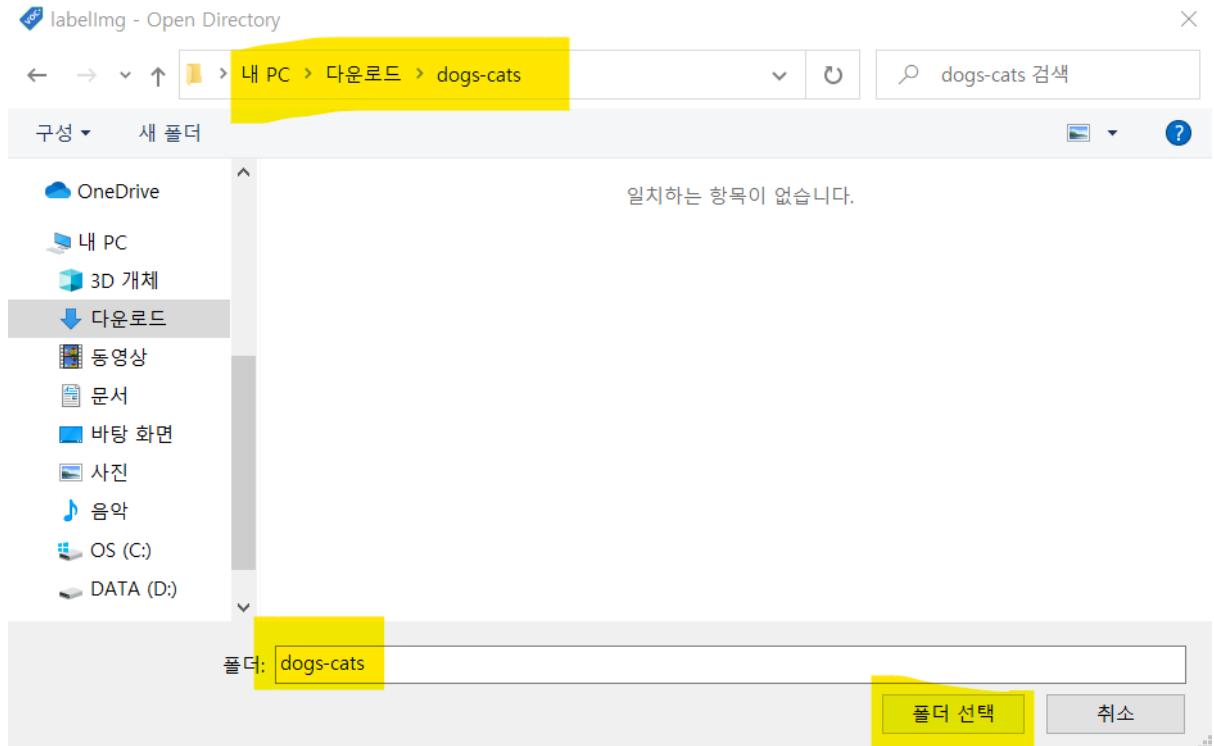
프로그램 실행

windows_v1.8.1/ 밑의 labelimg.exe를 실행

좌측의 'Open Dir'을 클릭하고



선택 화면에서 dogs-cats 폴더를 선택한다.



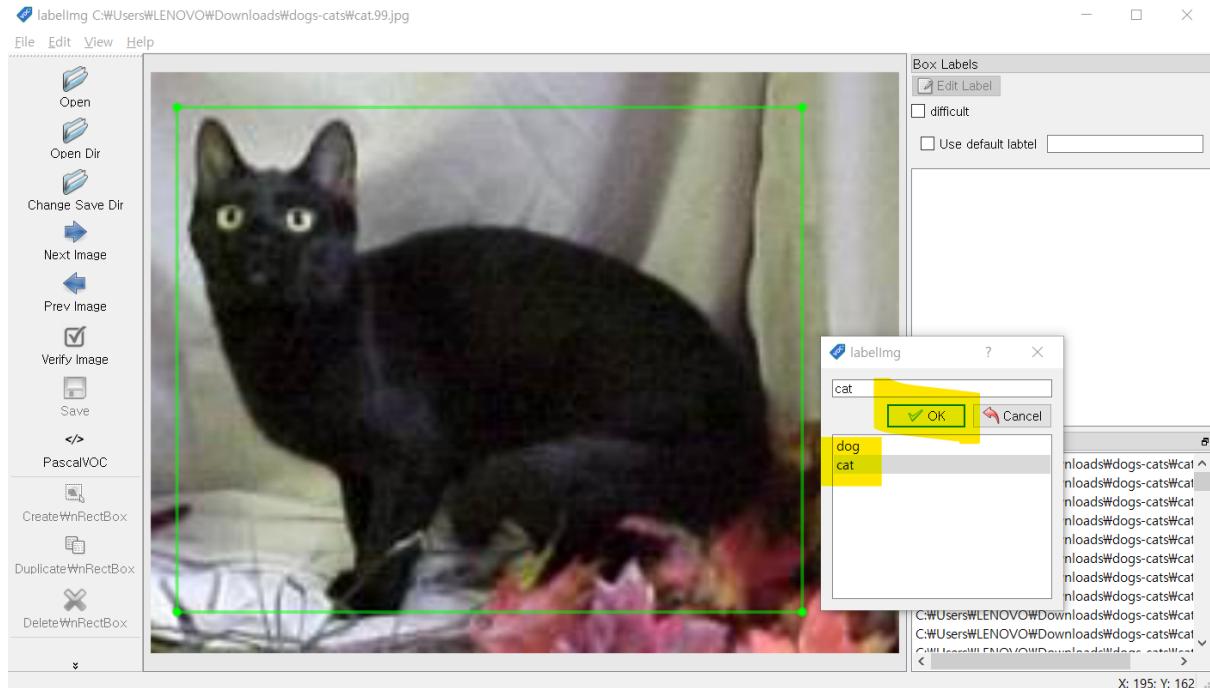
폴더내의 첫번째 이미지가 화면에 보인다.



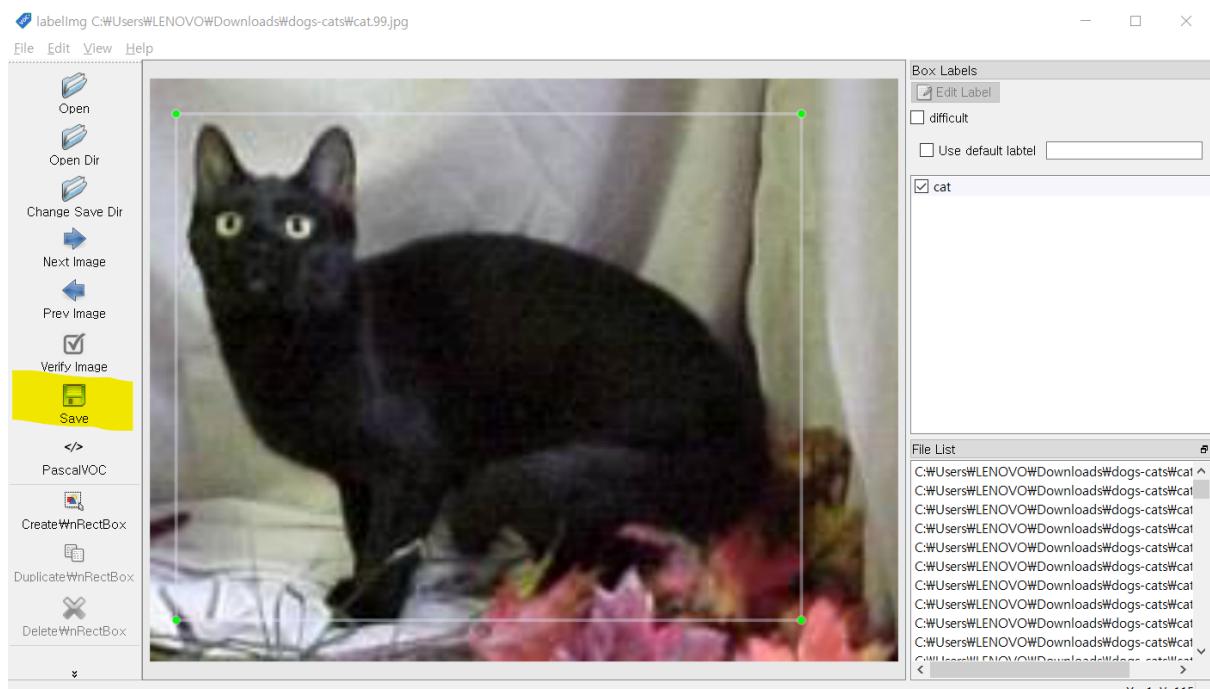
키보드로 w를 클릭하고, 마우스로 상자를 드래그앤드랍해서 상자를 그린다.

그리고 팔업창에서 레이블을 선택하고 'OK' 버튼을 클릭.

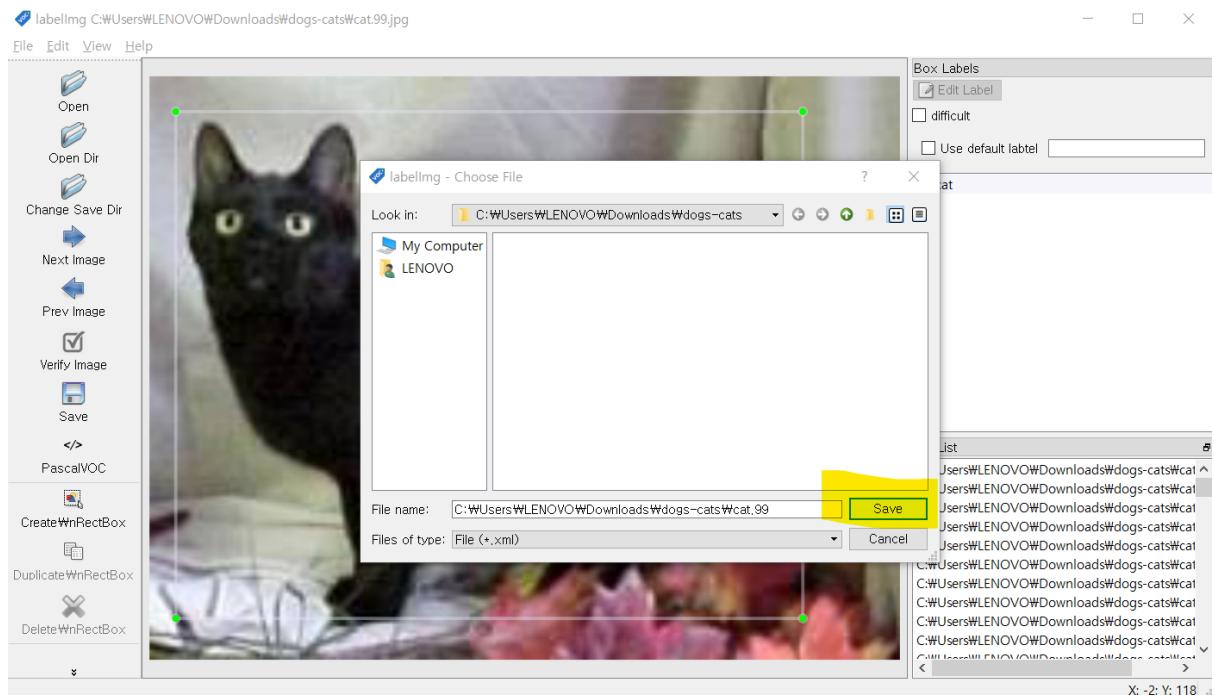
판업창에 보여지는 레이블은 앞에서 작성한 windows_v1.8.1/data/ 밑의 predefined_classes.txt 내용이다.



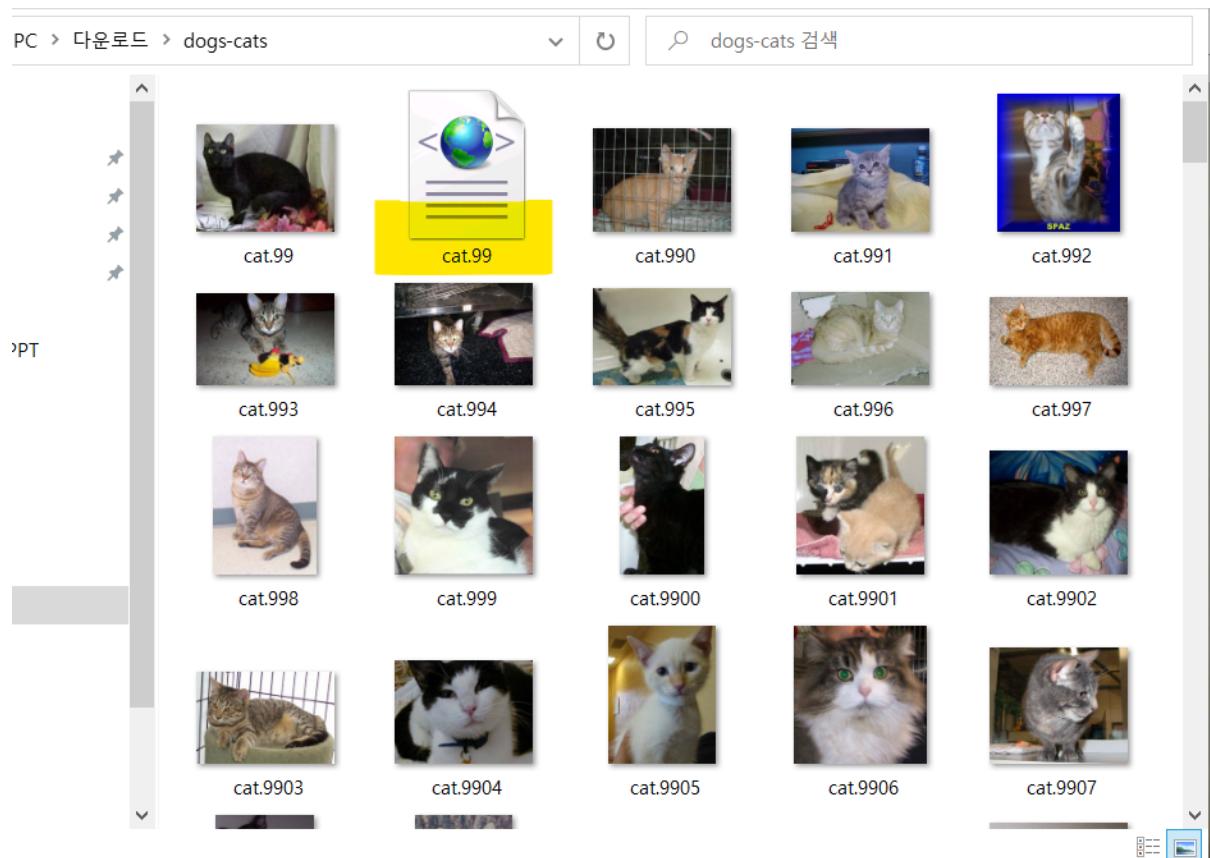
좌측의 'Save'를 클릭.



팝업창에서 'Save'를 클릭



이미지가 있는 폴더에 cat.99.xml 파일이 생성되었다.



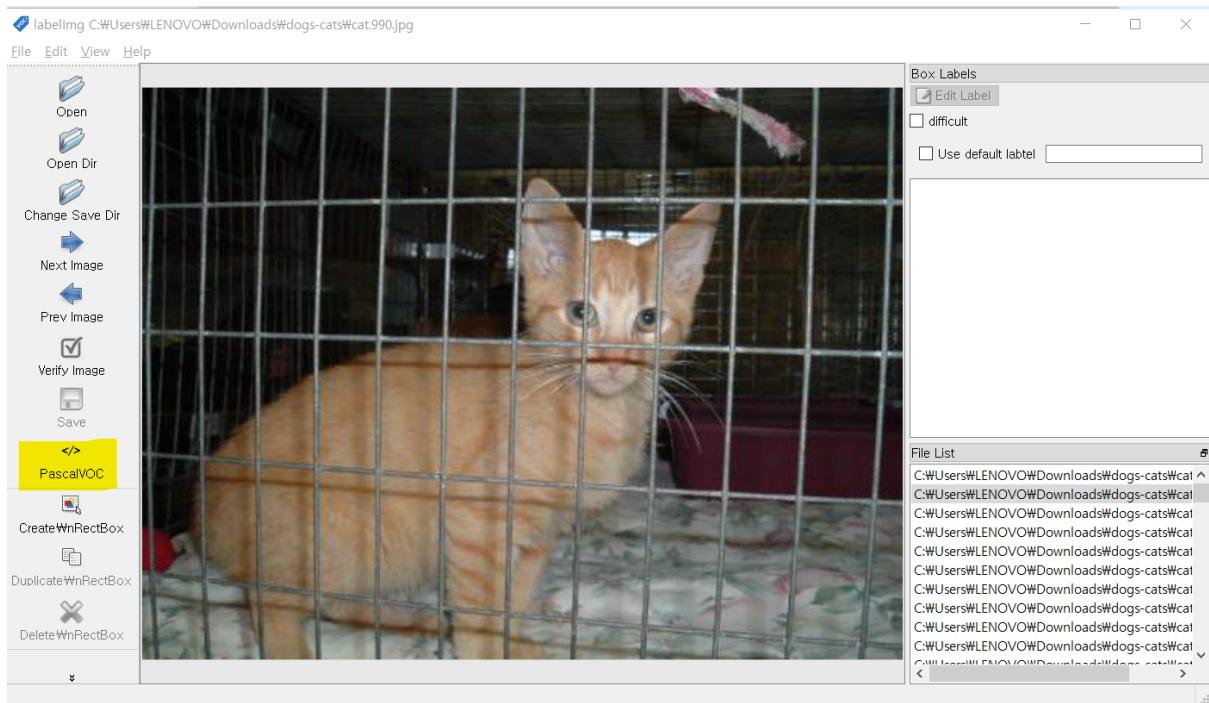
다음은 cat.99.xml 파일의 내용.

이미지의 사이즈(255x175)가 보이고, 상자의 위치((7,10), (195,162))가 보이고, 레이블 'cat'이 보인다.

```
<?xml version="1.0"?>
<annotation>
- <folder>dogs-cats</folder>
  <filename>cat.99.jpg</filename>
  <path>C:\Users\LENOVO\Downloads\dogs-cats\cat.99.jpg</path>
- <source>
  <database>Unknown</database>
</source>
- <size>
  <width>225</width>
  <height>175</height>
  <depth>3</depth>
</size>
<segmented>0</segmented>
- <object>
  <name>cat</name>
  <pose>Unspecified</pose>
  <truncated>0</truncated>
  <difficult>0</difficult>
- <bndbox>
  <xmin>7</xmin>
  <ymin>10</ymin>
  <xmax>195</xmax>
  <ymax>162</ymax>
</bndbox>
</object>
</annotation>
```

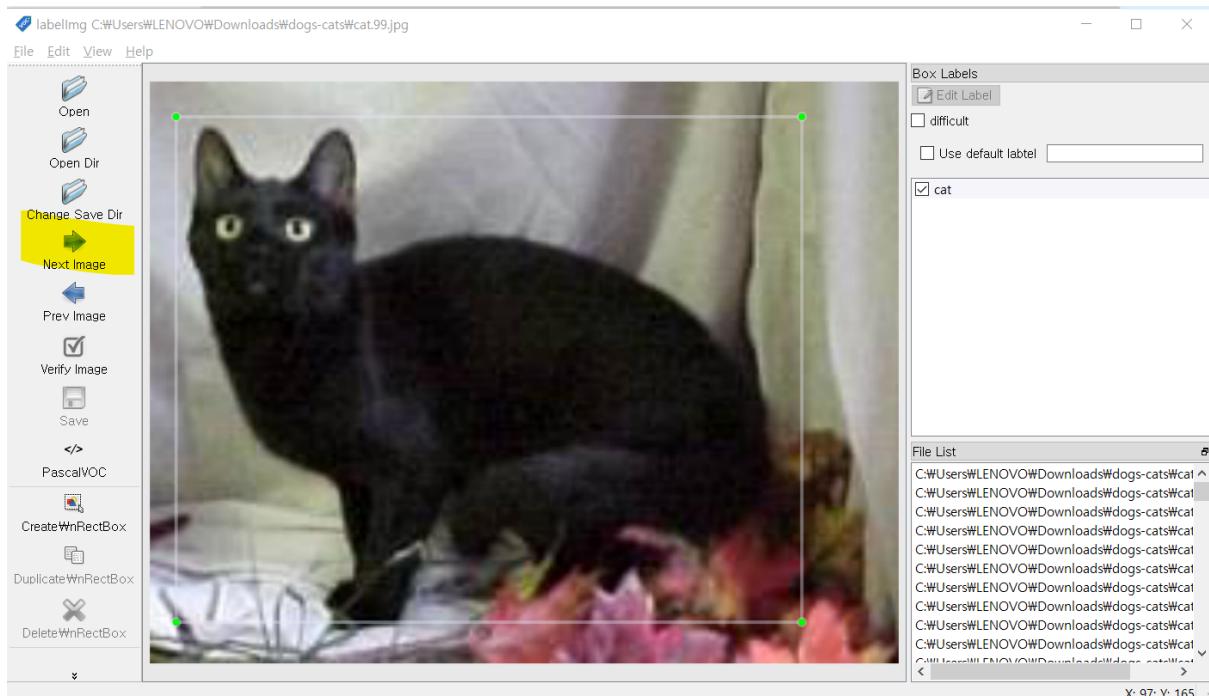
저장되는 포맷은 2가지이다. VOC와 YOLO. 좌측에 PascalVOC가 보이는 상태로 해야 xml로 저장된다.

PascalVOC인지 확인한다.

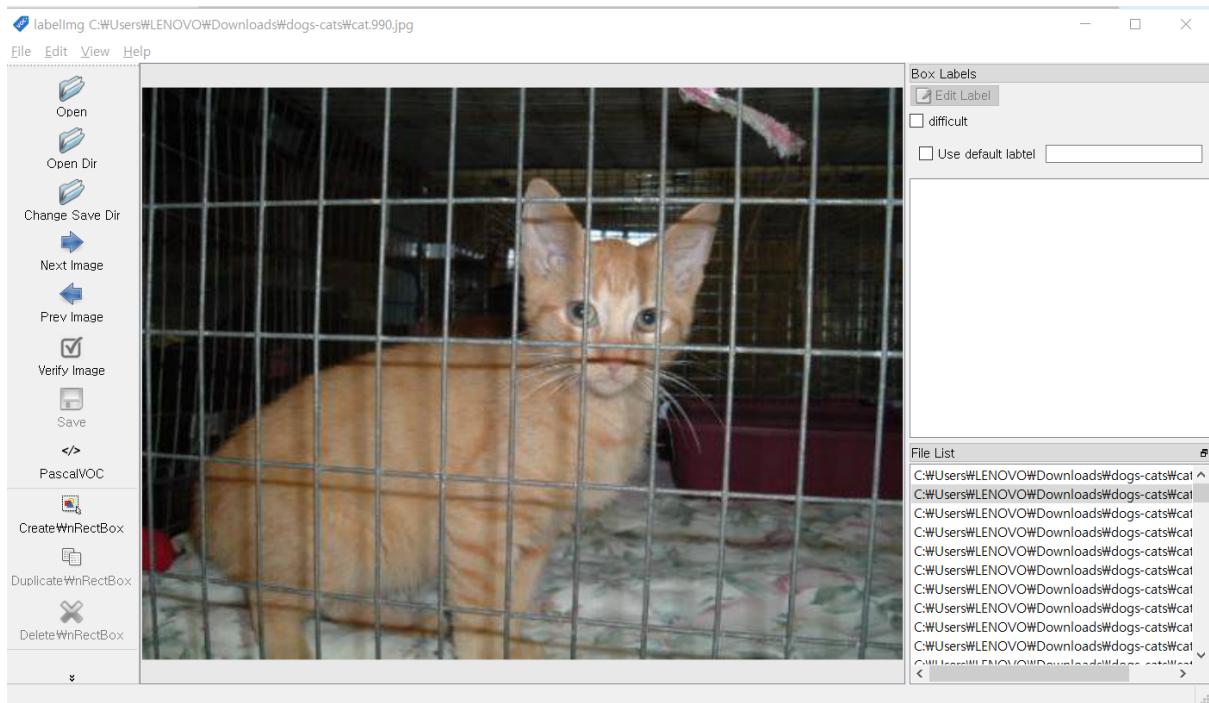


만약 아니면 그 부분을 클릭하면 변한다.

프로그램의 좌측에서 'Net Image'를 클릭하면 다음 이미지가 보여진다.



같은 방법으로 모든 파일에 대하여 레이블링 한다.



레이블링 데이터 변환

PascalVOC 포맷으로 저장된 xml파일을 keras-yolo3에서 사용하려면 keras-yolo3의 포맷으로 레이블링 데이터를 변환해야 한다.

다음 코드의 2개 부분을 수정하고 코드를 실행시킨다.

- CLASS_NAMES : 레이블링한 대상 이름들
- TARGET_FOLDER_NAME : xml 파일들이 있는 폴더 위치

```
s%cd /content/keras-YOLOv3-model-set

import os
import glob

CLASS_NAMES = [ "cat", "dog" ]
TAGET_FOLDER_NAME = "labels"

label_file_names = []
for file_name in glob.glob(TAGET_FOLDER_NAME+'/*.xml'):
    label_file_names.append(file_name)

print(len(label_file_names))
print(label_file_names)

import xml.etree.ElementTree as ET
```

```

all_record = []

for file_name in label_file_names:

    with open(file_name) as in_file:

        tree = ET.parse(in_file)
        root = tree.getroot()

        a_record = [ file_name.replace("xml", "jpg") ]

        for obj in root.iter('object'):
            difficult = obj.find('difficult').text
            cls = obj.find('name').text
            if cls not in CLASS_NAMES or int(difficult)==1:
                continue
            cls_id = str(CLASS_NAMES.index(cls))
            xmlbox = obj.find('bndbox')
            xmin = xmlbox.find('xmin').text
            ymin = xmlbox.find('ymin').text
            xmax = xmlbox.find('xmax').text
            ymax = xmlbox.find('ymax').text

            a_record.append(",".join([xmin,ymin,xmax,ymax,cls_id]))

    all_record.append(a_record)

with open("labels.txt", "w") as f:
    for a_record in all_record:
        print(a_record)
        f.write(" ".join([str(i) for i in a_record]))
        f.write("\n")

print("labels.txt created.")

```

다음 내용의 labels.txt 파일이 생성된다.

```

dogs_cats_yolo_labeled/cat.993.jpg 77,1,336,217,0
dogs_cats_yolo_labeled/cat.9994.jpg 73,2,366,297,0
dogs_cats_yolo_labeled/cat.996.jpg 63,24,365,231,0
dogs_cats_yolo_labeled/cat.9929.jpg 70,57,245,321,0
dogs_cats_yolo_labeled/cat.9945.jpg 55,1,158,127,0
dogs_cats_yolo_labeled/cat.9960.jpg 170,49,278,151,0 106,57,171,151,0 10,153,15,234,0

```