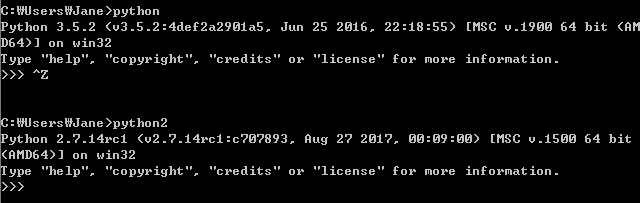
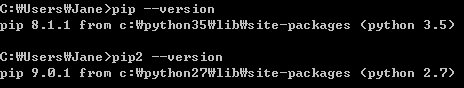
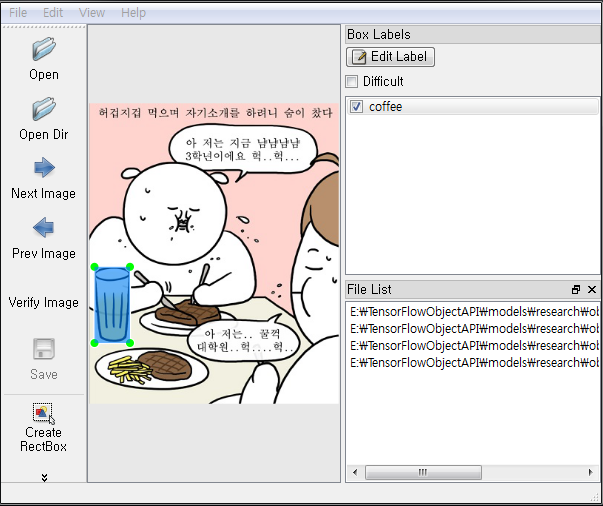
1. **이미지에 레이블 추가하여 VOC데이터 생성하기**

* Tesnorflow object detection은 dataset형식이 TFRecord여야합니다. 기존의 Pascal Voc데이터는 코드를 통해 TFRecord형식으로 바꿀 수 있습니다.
* Voc형식( 이미지labeling 정보를 .xml 파일에 저장합니다. )으로 데이터를 저장하기 위해 Image Labeling오픈소스를 사용할 수 있습니다. Image Labeling 마우스로 레이블 위치를 표시하면 자동으로 .xml파일로 저장됩니다. ( 여러 가지 소스들이 많은데 저는labelImg오픈소스를 선택하였습니다. <https://github.com/tzutalin/labelImg> 참고 )
* labelImg는 python2+Qt4에서 사용할 수 있고 Tensorflow는 python3에서 사용할 수 있기 때문에 python2 와 pip2 를 python3과 pip3과 중복되지 않게 주의!



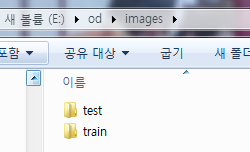


* labelImg를 실행하면 이미지중 object를 box로 표시하고 label를 추가하면 .xml형식인 voc데이터로 저장할 수 있습니다.

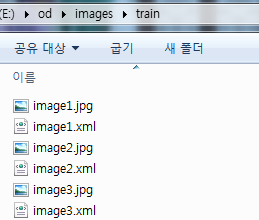
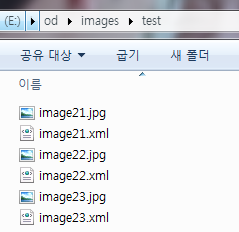




* 새로운 폴더 od를 만들고, 그 안에 또 새로운 폴더 images를 만듭니다. Images 폴더 안에 train와 test폴더를 새로 만듭니다.

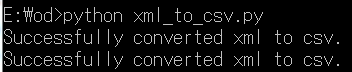


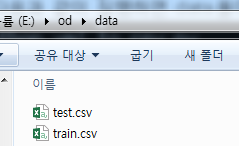
* train폴더에 트레이닝 할 이미지와 xml파일을 넣고 test폴더에 테스트 할 이미지와 xml파일을 넣으면 voc데이터 만들기 성공!!

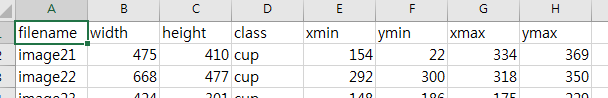
 

1. **xml파일을 csv로 변환하기**

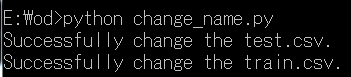
* od폴더 안에 data폴더를 만듭니다. data폴더에 코드를 통해 전환한 데이터를 저장합니다. od폴더안에 xml\_to\_csv.py와 change\_name.py코드를 복사합니다. cmd에서 다음과 같이 실행하면 data폴더에서csv파일을 확인할 수 있습니다.

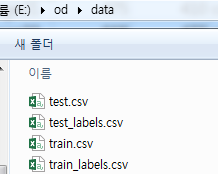


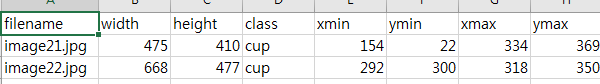




* 생성된 csv의 filename을 보면 .jpg가 생략되었습니다. cmd에서 다음과 같이 실행하면 data폴더에서 .jpg를 추가한 파일이름을 확인할 수 있습니다.







1. **Tfrecord로 변환하기**

* od폴더안에 test\_generate\_tfrecord.py와 train\_generate\_tfrecord.py코드를 복사합니다. cmd에서 다음과 같이 실행하면 data폴더에서 tfrecord 변환된 것을 확인할 수 있습니다.

python train\_generate\_tfrecord.py --csv\_input=data/train\_labels.csv --output\_path=data/train.record

python test\_generate\_tfrecord.py --csv\_input=data/test\_labels.csv --output\_path=data/test.record