Morugamu Open CV Project

Stuff to enter in the console will be denoted like this:

```
Sudo apt-get ‘example thing’
```

So, this will be a guide to do all of this in **Linux Ubuntu 16.04 (This most likely also works for Mac OS since it is also Unix)**
(You might be able to do it in windows, Daniel Flohr has more info on that)

1. CREATE A FRESH FOLDER (**NOT IN DOWNLOADS OR DESKTOP**).
   Create this new folder in, for example, the Documents folder. Name it, “Morugamu_CV” exactly in that casing and spelling.
2. Inside that “Morugamu_CV” Folder clone the tensorflow “models” folder:
   https://github.com/tensorflow/models
3. Install Python **3.6.6:**
   https://www.python.org/downloads/release/python-366/
4. Install Anaconda for **PYTHON 3.6 VERSION** for Linux (or preferred platform):
   https://www.anaconda.com/download/#linux
5. Download pip using this command in linux:
   `sudo apt-get install python-pip`
6. Then follow this guide to install tensorflow:
   https://towardsdatascience.com/real-time-object-detection-api-using-tensorflow-and-opencv-47b505d745c4
7. Let us go back to the tensorflow “models” folder inside the “Morugamu_CV” folder in your Documents. Navigate to models/research/ and in this folder you will paste the “fish-detection” folder in.
8. `cd` into the Morugamu_CV/models/research directory and run this following command line thingy:

```
PIPELINE_CONFIG_PATH=fish_detection/models/model/ssd_mobilenet_v1_pets.config
MODEL_DIR=fish_detection/models/model/
NUM_TRAIN_STEPS=50000
NUM_EVAL_STEPS=2000
python fish_detection/model_main.py \
    --pipeline_config_path=${PIPELINE_CONFIG_PATH} \
    --model_dir=${MODEL_DIR} \
    --num_train_steps=${NUM_TRAIN_STEPS} \
    --num_eval_steps=${NUM_EVAL_STEPS} \
    --alsologtostderr
```

9. After debugging all the issues/errors that come with running this line in the command prompt, your terminal may spit out warning, but if there are no specific errors, that means you might be training the model. This is as far as we got, I don’t know what else to do after this. Here are some helpful things: [down below]

Don’t remember what this is for, but seems useful:

```
INPUT_TYPE=image_tensor
PIPELINE_CONFIG_PATH= fish_detection/models/model/pipeline.config
TRAINED_CKPT_PREFIX={path to model.ckpt}
EXPORT_DIR={path to folder that will be used for export}
python object_detection/export_inference_graph.py \
    --input_type=${INPUT_TYPE} \
    --pipeline_config_path=${PIPELINE_CONFIG_PATH} \
    --trained_checkpoint_prefix=${TRAINED_CKPT_PREFIX} \
    --output_directory=${EXPORT_DIR}
```

Articles:

Raccoon Detector (what we based our entire project off of):

https://towardsdatascience.com/how-to-train-your-own-object-detector-with-tensorflows-object-detector-api-bec72ecfe1d9

Softwares:

1. **ImageMagick** for Linux is really helpful for batch image renaming or resizing. We recommend 600px landscape width for training image sets.
2. ^^ **XnView for Windows** ^^
3. **LabelImg: Annotating Images to train a model. Exports to xml files.**