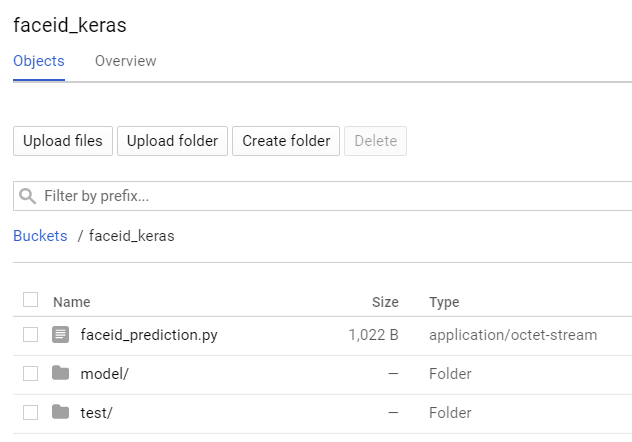
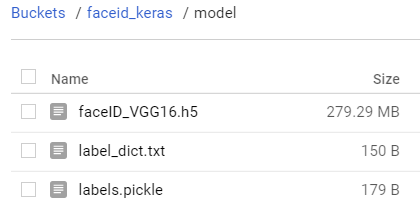
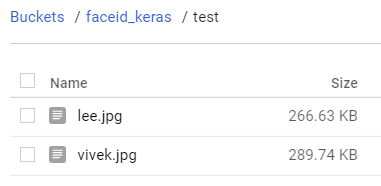
# Running Keras Model on Google Cloud Instance

8/3/2018 Mooyoung Lee

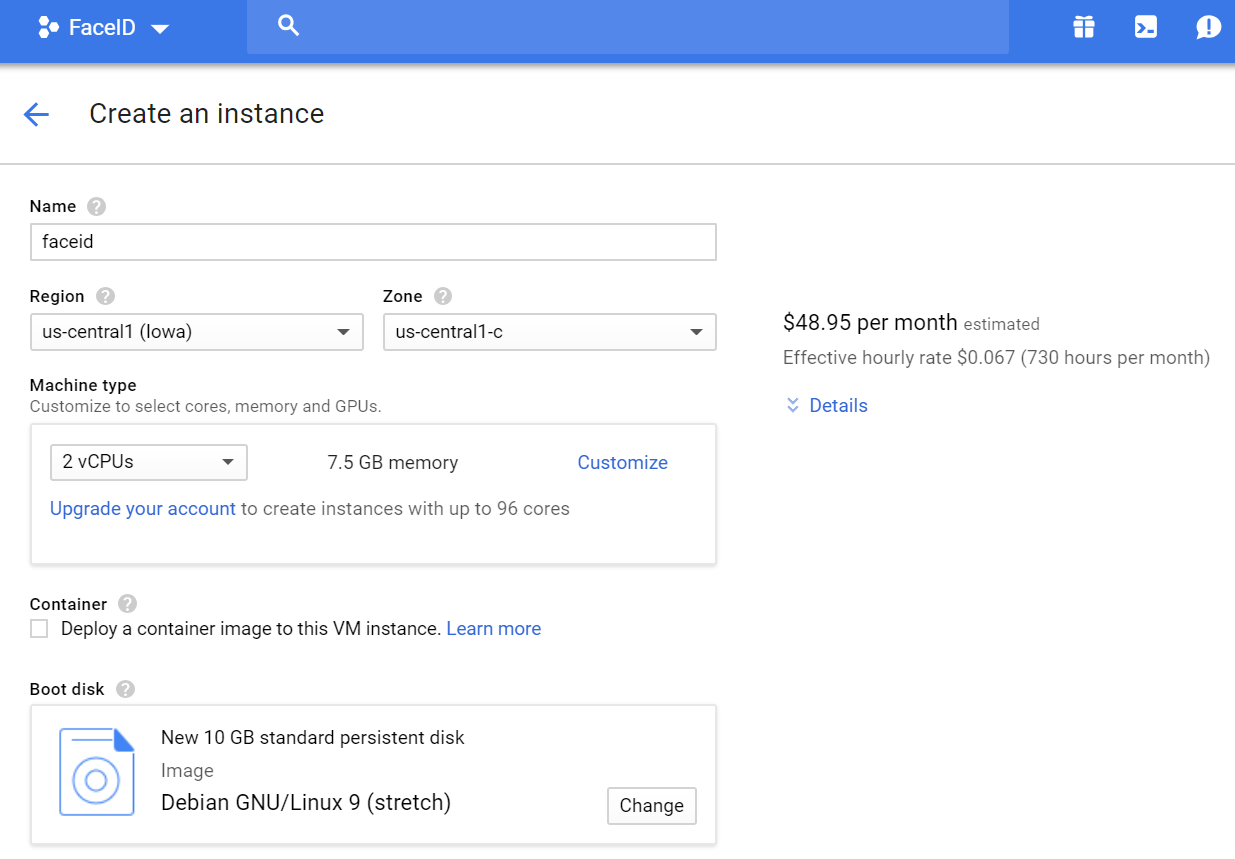
Bucket setup



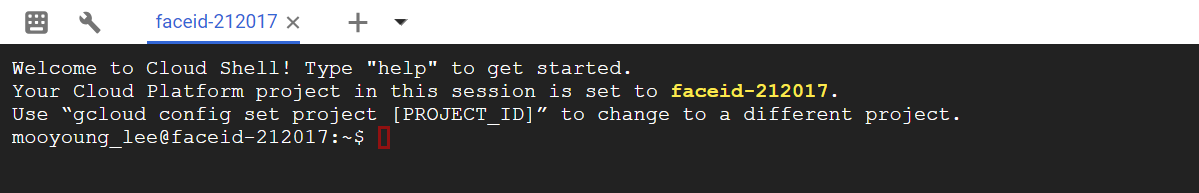




Create a compute engine instance



Connect to the instance



folder permission tips (run if needed)

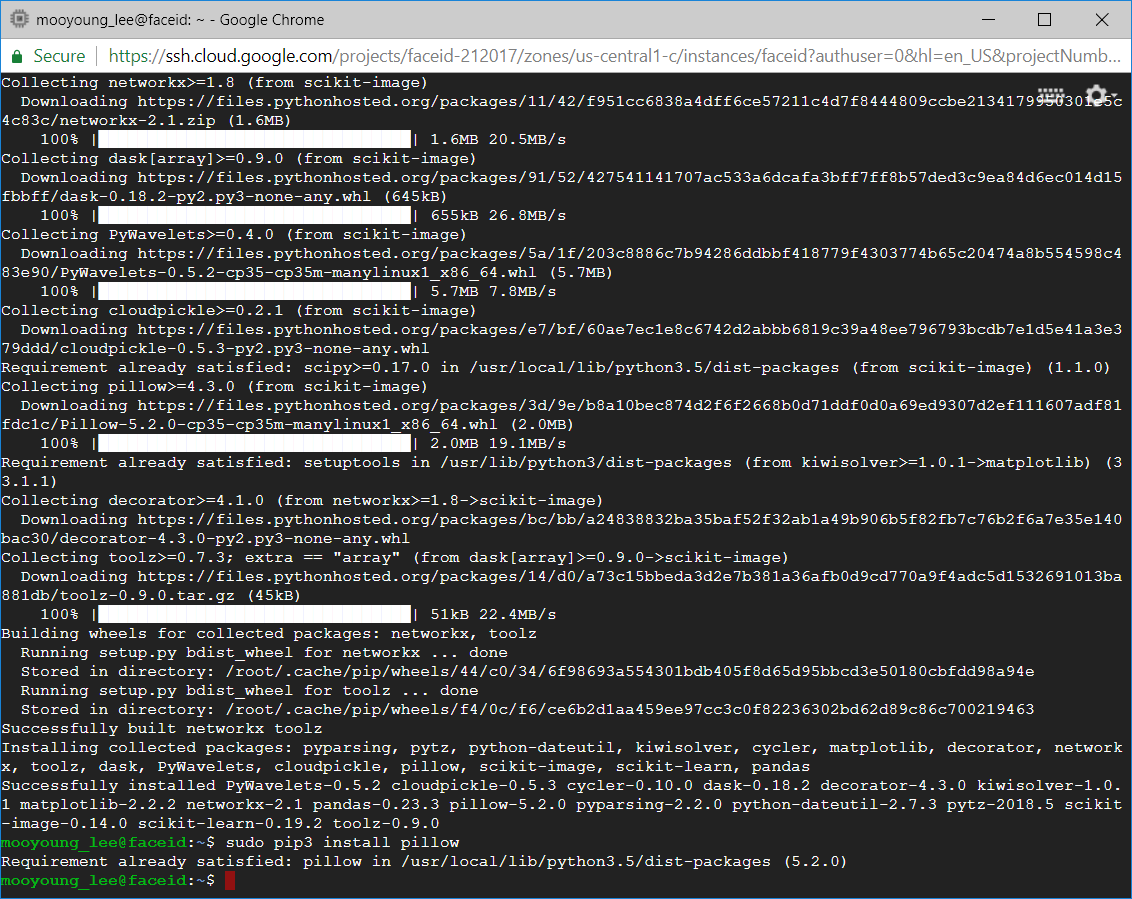
|  |
| --- |
| delete folder/files:  > rm -rf xxx  read/write permission:  > sudo chmod -R ugo+rw id  bucket public-readability  > gsutil -m acl -r set public-read gs://faceid\_keras |

Install python3

|  |
| --- |
| > sudo apt-get update  > sudo su  > sudo apt-get install -y libprotobuf-dev libleveldb-dev libsnappy-dev libhdf5-serial-dev protobuf-compiler libopencv-dev  sudo apt-get install make  $ sudo apt-get install -y build-essential checkinstall  $ sudo apt-get install -y libreadline-gplv2-dev libncursesw5-dev libssl-dev libsqlite3-dev tk-dev libgdbm-dev libc6-dev libbz2-dev  wget <https://www.python.org/ftp/python/3.6.3/Python-3.6.3.tgz>  tar -xvf Python-3.6.3.tgz  cd Python-3.6.3  ./configure  sudo apt-get install zlib1g-dev  sudo make  sudo make install  python3 -V  source :< <https://stackoverflow.com/questions/47273260/google-cloud-compute-engine-change-to-python-3-6?rq=1>> |
|  |

> sudo pip3 install tensorflow

> sudo pip3 install tensorflow keras matplotlib scikit-image scikit-learn pandas



Reference : < https://www.learnopencv.com/installing-deep-learning-frameworks-on-ubuntu-with-cuda-support/>

# Connect bucket and the instance

> mkdir id

> sudo gcsfuse faceid\_keras id

Install gcsfuse (if gcsfuse not working)

1. Add the gcsfuse distribution URL as a package source and import its public key:

export GCSFUSE\_REPO=gcsfuse-`lsb\_release -c -s`

echo "deb http://packages.cloud.google.com/apt $GCSFUSE\_REPO main" | sudo tee /etc/apt/sources.list.d/gcsfuse.list

curl https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -

1. Update the list of packages available and install gcsfuse.

sudo apt-get update

sudo apt-get install gcsfuse

Prediction w/ python model

> Python3

'faceid\_prediction.py'

|  |
| --- |
| from PIL import Image, ImageOps  image = Image.open('id/test/lee.jpg')  import numpy as np  from keras.models import load\_model  from keras import optimizers  import pandas as pd  import json  # test file prep  size = (224,224)  img = ImageOps.fit(image, size, Image.ANTIALIAS)  img = np.reshape(img,[1,224,224,3])  # load label  with open('id/model/label\_dict.txt', 'rb') as f:  old\_labels = json.load(f)  labels = {v:k for k, v in old\_labels.items()}  # load model  model = load\_model('id/model/faceID\_VGG16.h5')  # compile model  model.compile(loss='categorical\_crossentropy',optimizer=optimizers.RMSprop(lr=1e-4), metrics=['acc'])  # top-5 prediction  Num\_Prediction = 5 # Number of people most matching  label = pd.DataFrame(list(labels.items()), columns = ['ID','Name'])  label['Probability'] = model.predict(img)[0]  label.sort\_values(by=['Probability'], axis = 0, ascending = False, inplace = True)  label.reset\_index(drop=True, inplace=True)  print(label.iloc[0:Num\_Prediction,:]) |

Model predictions w/ 'lee.jpg' and 'vivek.jpg'

