IoT: Capturing Photos and Analyzing The Image with TensorFlow on a Raspberry Pi

(/users/9304/tspann.html)

Timothy Spann (/users/9304/tspann.html) created · Jan 28 at 04:50 PM

Short Description:

IoT: Capturing Photos and Analyzing The Image with TensorFlow on a Raspberry Pi Running TensorFlow for RPI

Article

Preparing a Raspberry PI to Run TensorFlow Image Recognition

I can easily have a Python script that polls my webcam (use official Raspberry Pi webcam) , calls TensorFlow and then sends the results to NiFi via MQTT.

You need to install Python MQTT Library (https://pypi.python.org/pypi/paho-mqtt/1.1 (https://pypi.python.org/pypi/paho-mqtt/1.1))

For setting up Python, Raspberry PI with Camera, see https://dzone.com/articles/picamera-ingest-real-time (https://dzone.com/articles/picamera-ingest-real-time)

Raspberry Pi 3 B+ preparation

Buy a good quality 16 GIG SD Card and from OSX, Run SD Formatter to Overwrite Format the device at FAT, download here: https://www.sdcard.org/downloads/formatter_4 (https://www.sdcard.org/downloads/formatter_4)/. Download the BerryBoot image from here (http://www.berryterminal.com/doku.php/berryboot). Unzip it and then copy it to your complete SD card.

For examples of RPi TensorFlow You Can Run:

https://github.com/tensorflow/tensorflow/tree/master/tensorflow/contrib/pi_examples (https://github.com/tensorflow/tensorflow/tree/master/tensorflow/contrib/pi_examples)/

You need to build tensorflow for pi, which took me over 4 hours.

See:

https://github.com/tensorflow/tensorflow/tree/master/tensorflow/contrib/makefile (https://github.com/tensorflow/tensorflow/tree/master/tensorflow/contrib/makefile)

https://github.com/tensorflow/tensorflow/tree/master/tensorflow/contrib/pi_examples (https://github.com/tensorflow/tensorflow/tree/master/tensorflow/contrib/pi_examples)/

Process:

```
wget https://github.com/tensorflow/tensorflow/archive/master.zipapt-get install -y libjpeg-devcd tensorflow-mastertensorflow/contrib/makefile/download_dependencies.shsudo apt-get install -y autoconf automake libtool gcc-4.8
g++-4.8cd tensorflow/contrib/makefile/downloads/protobuf/./autogen.sh./configuremakesudo make installsudo ldconfig  # refresh shared library cachecd ../../../../..make -f tensorflow/contrib/makefile/Makefile HOST_OS=PI TARGET=PI \ OPTFLAGS="-Os -mfpu=neon-vfpv4 -funsafe-math-optimizations -ftree-vectorize" CXX=g++-4.8curl https://storage.googleapis.com/download.tensorflow.org/models/inception_dec_2015_stripped.zip \-o /tmp/inception_dec_2015_stripped.zipunzip /tmp/inception_dec_2015_stripped.zip \-d tensorflow/contrib/pi_examples/label_image/data/make -f tensorflow/contrib/pi_examples/label_image/Makefile
```

```
root@raspberrypi:/opt/demo/tensorflow-master#
tensorflow/contrib/pi_examples/label_image/gen/bin/label_image2017-01-28 01:46:48: I
tensorflow/contrib/pi_examples/label_image/label_image.cc:144]
Loaded JPEG: 512x600x32017-01-28 01:46:50: W
tensorflow/core/framework/op_def_util.cc:332] Op
BatchNormWithGlobalNormalization is deprecated. It will cease to work in
GraphDef version 9. Use tf.nn.batch_normalization().2017-01-28 01:46:52: I
tensorflow/contrib/pi_examples/label_image/label_image.cc:378] Running model
succeeded!2017-01-28 01:46:52: I
tensorflow/contrib/pi_examples/label_image/label_image.cc:272] military uniform
```

It took over 4 hours to build. But only 4 seconds to run and gave good results for analyzing a picture of Computer Legend Grace Hopper.

```
root@raspberrypi:/opt/demo/tensorflow-master#
tensorflow/contrib/pi_examples/label_image/gen/bin/label_image --help2017-01-28 01:51:26: E
tensorflow/contrib/pi_examples/label_image/label_image.cc:337]usage: tensorflow/contrib/pi_examples/label_image/gen/bin/label_i
mageFlags:  --image="tensorflow/contrib/pi_examples/label_image/data/grace_hopper.jpg"  string  image to be processed  --
graph="tensorflow/contrib/pi_examples/label_image/data/tensorflow_inception_stripped.pb"  string  graph
to be executed  --labels="tensorflow/contrib/pi_examples/label_image/data/imagenet_comp_graph_label_strings.txt"  string  name
of file containing labels  --input_width=299    int32  resize image to this
width in pixels  --input_height=299    int32  resize image to this height in pixels  --input_mean=128    int32  scale pixel valu
es to
this mean  --input_std=128    int32  scale pixel values to this std deviation  --input_layer="Mul"    string  name of input laye
```

RELATED ARTICLES

Using An ASUS TinkerBoard with TensorFlow and Python to Ingest Data to Hadoop with Apache NiFi (/articles/103863/using-an-asus-tinkerboard-with-tensorflow-and-pyth.html)

How to write topology with the new kafka spout client in storm (/articles/87597/how-to-write-topology-with-the-new-kafka-spout-cli.html)

How to convert the HDP Sandbox into a Vagrant Box on Mac OS X (/articles/45043/how-to-convert-the-hdp-sandbox-into-a-vagrant-box.html)

How to auto create topics in Ranger enabled Kafka cluster (/articles/91546/how-to-auto-create-topics-in-ranger-enabled-kafka.html)

How to manage built in database under Cloudbreak (/articles/108675/how-to-manage-built-in-database-under-cloudbreak.html)

Sample Java code to run Sqoop Command using google SSHXCUTE framework (/articles/72959/sample-java-code-to-run-sqoop-command-using-google.html)

Hive View in HDP 2.5 and Ambari 2.4 returns 'userhome' error (/articles/83143/hive-view-in-hdp-25-and-ambari-24-returns-userhome.html)

Protocol-level debugging of the Phoenix Query Server (/articles/93080/protocol-level-debugging-of-the-phoenix-query-serv.html)

Partner Demo Kit (/articles/75341/partner-demo-kit-1.html)

How to run service checks for various services using Ambari Rest APIs. (/articles/73626/how-to-run-service-checks-for-various-services-usi.html)