tensorflow / **tensorflow**

Branch: master ▾ tensorflow / tensorflow / contrib / **pi_examples** /

Create new file | Find file | History

caisq committed with **tensorflower-gardener** Merge changes from github. ... Latest commit e2e3a94 24 days ago

..

camera	Fix undefined reference to libtensorflow-core.a (#12778)	a month ago
label_image	Fix undefined reference to libtensorflow-core.a (#12778)	a month ago
.gitignore	Added camera example for Raspberry Pi (#2949)	a year ago
README.md	Update README.md	5 months ago

README.md

TensorFlow Raspberry Pi Examples

This folder contains examples of how to build applications for the Raspberry Pi using TensorFlow.

Building the Examples

- Follow the Raspberry Pi section of the instructions at tensorflow/contrib/makefile to compile a static library containing the core TensorFlow code.
- Install libjpeg, so we can load image files:

```
sudo apt-get install -y libjpeg-dev
```

- To download the example model you'll need, run these commands:

```
curl https://storage.googleapis.com/download.tensorflow.org/models/inception_dec_2015_stripped.zip \
-o /tmp/inception_dec_2015_stripped.zip
unzip /tmp/inception_dec_2015_stripped.zip \
-d tensorflow/contrib/pi_examples/label_image/data/
```

- From the root of the TensorFlow source tree, run `make -f tensorflow/contrib/pi_examples/label_image/Makefile` to build a basic example.

Usage

Run `tensorflow/contrib/pi_examples/label_image/gen/bin/label_image` to try out image labeling with the default Grace Hopper image. You should several lines of output, with "Military Uniform" shown as the top result, something like this:

```
I tensorflow/contrib/pi_examples/label_image/label_image.cc:384] Running model succeeded!
I tensorflow/contrib/pi_examples/label_image/label_image.cc:284] military uniform (866): 0.624293
I tensorflow/contrib/pi_examples/label_image/label_image.cc:284] suit (794): 0.0473981
I tensorflow/contrib/pi_examples/label_image/label_image.cc:284] academic gown (896): 0.0280926
I tensorflow/contrib/pi_examples/label_image/label_image.cc:284] bolo tie (940): 0.0156956
I tensorflow/contrib/pi_examples/label_image/label_image.cc:284] bearskin (849): 0.0143348
```

Once you've verified that is working, you can supply your own images with `--image=your_image.jpg`, or even with graphs you've trained yourself with the TensorFlow for Poets tutorial using `--graph=your_graph.pb --input=Mul:0 --output=final_result:0`.

Camera Example

Once you have the simple example running, you can try out a more complex version that reads frames from a camera attached to the Pi. You'll need to install and set up your camera module first. The example uses Video4Linux, so you'll need to install that first. Here's some commands I found necessary to get that set up, and I found more information at this blog post: http://www.richardmudhar.com/blog/2015/02/raspberry-pi-camera-and-motion-out-of-the-box-sparrowcam/

```
sudo bash -c "echo 'bcm2835-v4l2' >> /etc/modules"
sudo apt-get install libv4l-dev
```

Once that's working, run the following commands to build and run the camera example:

```
make -f tensorflow/contrib/pi_examples/camera/Makefile
tensorflow/contrib/pi_examples/camera/gen/bin/camera
```

You should see it looping over camera frames as they come in, and printing the top labels to the command line. This is a great starting point for all sorts of fun image recognition applications, especially when you combine it with a custom model you've built using something like the TensorFlow for Poets tutorial.

The example is designed to work with the Flite speech synthesis tool, so that your Pi can speak any labels that have a high enough score. To enable this, just install the Flite package and then pipe the output of the binary you've built, like this:

```
sudo apt-get install flite
tensorflow/contrib/pi_examples/camera/gen/bin/camera | xargs -n 1 flite -t
```