

[Ideas](#)[Learning](#)[Events](#)[Shop](#)[FEEDBACK](#) [LOG OUT](#)[ON OUR RADAR](#)[AI](#)[BUSINESS](#)[DATA](#)[DESIGN](#)[ECONOMY](#)[OPERATIONS](#)[SECUR](#)[SEE ALL](#)

AI

FOLLOW THIS TOPIC

# TensorFlow on Android

Take deep learning mobile.

By Justin Francis, October 13, 2016

*To learn how to build and train your first TensorFlow graph from the ground up, check out [Aaron Schumacher's Oriole Tutorial: Hello, TensorFlow!](#)*

If you followed my [previous post](#), you learned how to install GPU-accelerated TensorFlow and create your own image classifier on a Linux



Smartphone. (source: Pixabay).

computer. Honestly, though, the process of classifying individual images is time consuming on a laptop: you have to download the image you want to classify and enter a lot of code into the terminal just to classify your image.

Thankfully (though it's not well publicized) you can run the Inception classifier—or your own image classifier—live on your camera-equipped phone. You simply need to point your camera at what you're trying to classify and TensorFlow will tell you what it thinks it is. You can also use TensorFlow on iOS and Raspberry Pi, but for this tutorial, I will be using an Android device.

I'll also step you through how I learned to get my custom classifier working on my Android device—getting the custom graph to work was a lot of work and was not documented anywhere. A lot of searching in TensorFlow's

GitHub forums was necessary; I hope I can spare you some of that trouble.

## Download Android SDK & NDK

You can download Android SDK using the terminal and then extract it into your TensorFlow directory.

```
$ wget https://dl.google.com/android/android-sdk_r24.4.1-linux.tgz  
$ tar xvzf android-sdk_r24.4.1-linux.tgz -C ~/tensorflow
```

Then we need to download some additional build tools for the SDK; I closed the terminal after it started downloading for Android 5.1.1.

```
$ cd ~/tensorflow/android-sdk-linux  
$ tools/android update sdk --no-ui
```

### Get O'Reilly's AI newsletter

OREILLY  
[Artificial Intelligence](#) Newsletter

1. Found in translation  
Google has replaced its old statistical machine translation method with a new one based on deep learning.

2. The simple economics of machine intelligence  
Harvard Business Review takes a look at whether, from an economic standpoint, machine intelligence lives up to its hype.

3. Introducing model-based thinking into AI systems  
In this episode of the O'Reilly Data Show Podcast, Vikash Mansinghka discusses recent developments in probabilistic programming.

Email Address [Subscribe](#)

We protect your privacy.

Next you can download the Android NDK and extract using:

```
$ wget https://dl.google.com/android/repository/android-ndk-r12b-linux-x86_64.zip  
$ unzip android-ndk-r12b-linux-x86_64.zip -d ~/tensorflow
```

## Download Inception

```
$ cd ~/tensorflow  
$ wget https://storage.googleapis.com/download.tensorflow.org/models/inception5h.zip -O /tmp/inception5h.zip  
$ unzip /tmp/inception5h.zip -d tensorflow/examples/android/assets/
```

## Modify WORKSPACE File

In order to build our app using the Android tools, we will need to modify our workspace

file.

```
$ gedit ~/tensorflow/WORKSPACE
```

You can copy the code below and overwrite the similar lines in your WORKSPACE file.

```
android_sdk_repository(  
    name = "androidsdk",  
    api_level = 24,  
    build_tools_version = "24.0.3",  
    path = "android-sdk-linux")
```

```
android_ndk_repository(  
    name="androidndk",  
    path="android-ndk-r12b",  
    api_level=21)
```

## Enable USB debugging and adb

In order to use adb, you have to put your phone into developer mode and enable USB debugging. To do this, make sure the phone *isn't* connected to a computer via USB and:

- Go to Setting - General - About Phone
- Go to Software info and touch "Build number" seven times in a row
- This will start a counter and will tell you when you are in Developer Mode
- Go to Settings - General - Developer Options
- Enable USB Debugging

Every Android phone is different, but I found my Android G4 had to be in PTP mode to use adb. You will also have to confirm the debugging connection via the phone after you plug your cell phone into your computer. When a screen appears on your phone saying "Allow

"USB debugging," make sure you select the box "Always allow from this computer" and press OK.

To test if this works, we can install adb, plug in our phone, and enter:

```
$ sudo apt-get install android-tools-adb  
$ adb devices
```

You should get a readout similar to this:

```
justin@wagon-helm:~$ adb devices  
List of devices attached  
LGH8121aa151ee    device  
justin@wagon-helm:~$
```

## Build the APK

```
$ cd ~/tensorflow
```

```
$ bazel build //tensorflow/examples/android:tensorflow_demo
```

## Install the APK

This is the only step I feel like I could not objectively test. Every android device is different; I suggest upgrading your device to Android 6.0 if you are having any issues. On my friend's Moto G, I had to remove the -g from this command:

```
$ adb install -r -g bazel-bin/tensorflow/examples/android  
/tensorflow_demo.apk
```

You can now have fun with TensorFlow and the Inception classifier on your android device. I find the best part is the humorous classifications it sometimes gets wrong. Keep

in mind the Inception classifier only knows 1,000 images used from the Imagenet challenge.

## Using a custom classifier

To use a custom graph from our own classifier, we have to optimize the graph file for mobile use and put it into the assets directory.

First we must build the graph optimizer:

```
$ cd ~/tensorflow  
$ bazel build tensorflow/python/tools:optimize_for_inference
```

Then optimize the graph:

```
$ bazel-bin/tensorflow/python/tools/optimize_for_inference \  
--input=tf_files/retrained_graph.pb \  
--output=retrained_graph_optimized.pb
```

```
--output=tensorflow/examples/android/assets/retrained_graph.pb  
--input_names=Mul \  
--output_names=final_result
```

## Copy labels into assets folder

```
$ cp ~/tensorflow/tf_files/retrained_labels.txt ~/tensorflow/tensorflow  
/examples/android/assets/
```

## Edit TensorflowImageListener.java

```
$ gedit ~/tensorflow/tensorflow/examples/android/src/org/tensorflow  
/demo/TensorFlowImageListener.java
```

We need to edit this file to use our custom graph: replace the following lines with these variables and save (if you want to revert back to

the older file you can find a copy [here](#)):

```
private static final int INPUT_SIZE = 299;
private static final int IMAGE_MEAN = 128;
private static final float IMAGE_STD = 128;
private static final String INPUT_NAME = "Mul:0";
private static final String OUTPUT_NAME = "final_result:0";

private static final String MODEL_FILE = "file:///android_asset
/retrained_graph.pb";
private static final String LABEL_FILE = "file:///android_asset
/retrained_labels.txt";
```

## Rebuild the APK

```
$ cd ~/tensorflow
$ bazel build //tensorflow/examples/android:tensorflow_demo
```

## Reinstall the APK

```
$ adb install -r -g bazel-bin/tensorflow/examples/android  
/tensorflow_demo.apk
```

And there we have it! You can now use your own custom classifier on your Android Device. There are endless ideas that researchers and hobbyists alike could experiment with using this technology. One idea I had was to try to learn how to classify nutrient deficiencies or other botanical ailments in plants by classifying images of unhealthy leaves. If you would like to learn how to further compress the graph file and how use a classifier on iPhone, you can follow [Pete Warden's tutorial](#).

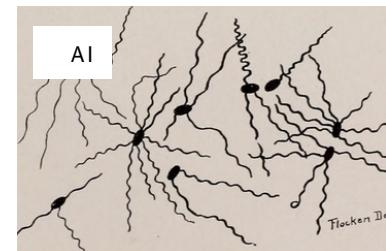
I hope I've inspired you and can't wait to see what you come up with! Feel free to tag me on Twitter [@wagonhelm](#) or [#TensorFlow](#) and share what you created.

Article image: Smartphone. (source: Pixabay).



## Justin Francis

Justin lives on the west coast of Canada and works on a small farm focused on permaculture ethics and design. In the past, he was the founder and educator at a non-profit Community Cooperative bicycle shop. For the last two years, he lived on a sailboat exploring and experiencing the Georgia Strait full-time but is now primarily focused on studying machine learning.



### Highlights from the O'Reilly AI Conference in New York 2016

By Mac Slocum

Watch highlights covering artificial intelligence, machine learning, intelligence engineering,

### How AI is propelling driverless cars, the future of surface transport

By Shahin Farshchi

Shahin Farshchi examines role artificial intelligence will play in driverless cars.

### Untapped opportunities in AI

By Beau Cronin

Some of AI's viable approaches lie outside the organizational boundaries of Google and other large Internet companies.

### Small brains, big data

By Jeremy Freeman

How neuroscience is benefiting from distributed computing, and how computing might learn from neuroscience.

and more. From the  
O'Reilly AI Conference in  
New York 2016.

**ABOUT US**

Our Company

Work with Us

Customer Service

Contact Us

**SITE MAP**

Ideas

Learning

Topics

All

© 2017 O'Reilly Media, Inc. All trademarks and registered trademarks appearing on oreilly.com are the property of their respective owners.

[Terms of Service](#) • [Privacy Policy](#) • [Editorial Independence](#)

