

Make:

[Subscribe](#)[Now](#)[Give a Gift](#)

Find your DIY supplies in the [Maker Shed](#) [Kits, Books, Components, 3D Printers,](#)
[Arduino, Raspberry Pi, More!](#)

MAKE: PROJECTS

Build Your Own Face-Recognition System with Intel Edison

Teach Intel's Edison how to spot faces using OpenCV

By [Stephanie Moyerman](#) Category: [Computers & Mobile](#), [Electronics](#)



PARTS

[Intel Edison](#)
[microcontroller](#)

[Intel IoT developer kit for Edison](#)

[OpenCV facial-recognition algorithm](#)

TOOLS

The Intel demo team at CES posing with some of the awards the Edison won.

Computer vision is a processor-demanding task, but thanks to a dual-core Atom processor, the Intel Edison handles it with ease. The Edison ships with a highly custom Linux image, but you'll only need to add a few software packages and custom code to get OpenCV — a wildly popular approach to computer vision — operational and recognizing human faces in photos.

1. Flash the Edison with the Latest Firmware

Follow the flashing instructions on the Intel documentation page at makezine.com/go/flashing-edison update your Edison with the latest image.

Then run the Edison configuration script:

```
configure_edison --setup
```

And follow the setup prompts to configure a hostname and root password and to set up Wi-Fi access.

2. SSH into the Edison

On Windows, download and install Putty, an SSH client. Then point Putty to your Edison.

On OSX or Linux, open a terminal and type:

```
ssh root@edison.local
```

NOTE: If you changed the hostname, replace `edison` in this address with the new name you created.

3. Install the latest IoT Developer Kit Libraries

Type in the following commands, and note this is actually one long line with spaces between `intel-iotdk` and the URL, and on both sides of the `>` character:

```
echo "src intel-iotdk
```

```
http://iotdk.intel.com/repos  
/1.1/intelgalactic" > /etc/opkg/intel-  
iotdk.conf
```

Update the package repository, then upgrade all the packages:

```
opkg update
```

```
opkg upgrade
```

4. Add an Unofficial Package Repository

Access to every package is not available without adding repository locations to the *opkg/base-feeds.conf* file. By doing this, you'll add an enormous number of compiled applications, saving you the hassle of compiling from source.

NOTE: Unofficial repositories are quite common across most Linux distributions.

Add the following lines to *base-feeds.conf*:

```
echo "src/gz all http://repo.opkg.net  
/edison/repo/all
```

```
src/gz edison http://repo.opkg.net/edison  
/repo/edison
```

```
src/gz core2-32 http://repo.opkg.net/edison  
/repo/core2-32" >> /etc/opkg/base-feeds.conf
```

Update the repository index again, since you just added new package locations:

```
opkg update
```

Next, install NumPy, OpenCV, and OpenCV-Python.

```
opkg install python-numpy opencv python-
```

```
opencv nano
```

That's it! All the necessary packages are installed. Time to start hacking code!

NOTE: Installing the basic text editor nano is not necessary, but is suggested unless you're comfortable using vi.

5. Programming with Python and OpenCV

Launch nano and specify a filename to use. Then import the 3 required Python libraries:

```
nano ~/FaceDetection.py
```

```
import numpy
```

```
import cv2
```

```
import urllib
```

Download and place our sample photo in the Edison's web server directory with the new filename, *in.jpg*.

```
print("Downloading Images and Necessary  
Files")
```

```
urllib.urlretrieve(http://cdn.makezine.com  
/make/43/Intel_CES_Team.png, '/usr/lib  
/edison_config_tools/public/in.jpg')
```

Next, download the XML file that defines the parameters for the OpenCV facial-recognition algorithm. This file is also saved to the public directory of the Edison's web server as *haarcascade_frontal face_alt.xml*.

```
urllib.urlretrieve('https://raw.githubusercontent.com  
/Itseez/opencv/master/data/haarcascades  
/haarcascade_frontalface_alt.xml',  
'/usr/lib/edison_config_tools/public  
/haarcascade_frontalface_alt.xml')
```

Import the photo using OpenCV and convert it to grayscale for use in the facial-recognition process:

```
img = cv2.imread('/usr
/lib/edison_config_tools/public/in.jpg')

gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
```

Using the OpenCV libraries, create the facial-recognition algorithm and process the grayscale image:

```
faceCascade =

cv2.CascadeClassifier('haarcascade_frontalface_alt.xml')

faces =

faceCascade.detectMultiScale(gray,scaleFactor=1.1,minNeighbors=5,

minSize=(30, 30), flags =

cv2.cv.CV_HAAR_SCALE_IMAGE)
```

The `faces` variable now contains an array of rectangular coordinates that surround each face that OpenCV found in the image. These coordinates are then used to draw a box around each face in the original color image, which you'll save as a new file:

```
for (x,y,w,h) in faces:

    cv2.rectangle(img,(x,y),(x+w,y+h),

(255,0,0),2)

cv2.imwrite('in_facefound.png',img)
```

Finally, save the text file by pressing Ctrl-X on your keyboard. When prompted to save the file, type Y and Enter.

6. Web page Setup

Download a simple HTML file which will display the pre- and post-processed images on the Edison's onboard web server.

```
wget http://cdn.makezine.com/make/43  
/OpenCV.html
```

Change directories to the web server's public directory:

```
cd /usr/lib/edison_config_tools/public
```

7. Viewing the Images

Now head on over to <http://edison.local/OpenCV.html> to view the Before and After images, with a box around each detected face!

Going Further

Now that OpenCV and Python are configured on your Edison, be sure to see the official documentation for great example code and ideas at makezine.com/go/opencv-python-tutorials. OpenCV can detect all kinds of shapes, analyze video, and much more.



Follow this story



STEPHANIE MOYERMAN

is a research scientist with the Smart Device Innovation Team in Intel's New Devices Group.

Comments Community**1** Login ▾

♥ Recommend

Sort by Best ▾

Join the discussion...

**John O'Reilly** • a day ago

I am stuck on step 4. When I type `opkg install python-numpy opencv python-opencv nano` into the terminal I get unknown package. Any help or insight would be highly appreciated. Also a quick question is it not advisable to enlarge the edison's partition due to the size of OpenCV?

^ | ▾ • Reply • Share ›

**Bill Ryder** • 24 days ago

I agree with Gaaaaare, is it possible to publish an errata page or make available a zip file with corrected code / examples? I closer to a NOOB than I would like to admit and I'm locked at step #5. Thanks, Bill

^ | ▾ • Reply • Share ›

**pradeep gaddam** • a month ago

Very detailed. Thank you Pradeep Reddy Gaddam

^ | ▾ • Reply • Share ›

**Peter Ma** • 4 months ago

Nice one!

^ | ▾ • Reply • Share ›

**Dave Hunt** • 4 months ago

OpenCV seems to think there's 6 faces in the above image.. ;)

^ | ▾ • Reply • Share ›

**Tod E. Kurt** • 4 months ago

The link in step (1) is giving me a 404 not found. The link text is correct but the URL is <http://makezine.com/projects/m...>

^ | ▾ • Reply • Share ›

**Caleb Kraft** Mod ➔ Tod E. Kurt • 4 months ago<https://communities.intel.com/...>

that is where it leads

^ | ▾ • Reply • Share ›

**Alex Godfrey** • 4 months ago

I finally got the python script to work. However, I found some errors in the published code. The first `urllib.request` is missing

