Leeds Guide to OpenCV

Starting Out In OpenCV

- If you'd like to get on with an overview of how to program in OpenCV rather than seeing what it's capable of doing, I highly recommend this page.
- If you're keen to get your hands on a 'Hello World' program, see here and here.
- If you'd prefer a worked example, click 'IplImage' on the left.

The sample programs that come with OpenCV help you to get a feel for the speed and power of the library. In Windows they're installed as executables and will work straight away (see below for what each one is/does), and in linux they are usually readily compiled. There is a script (build_all.sh) in case they don't work, and it's only if you're unlucky that linux will give you any trouble (missing libraries, errors, warnings, etc.)
They are located wherever OpenCV was extracted to, under /samples/c/, along with some in python (samples/python/) if you prefer van Rossum's approach. I haven't tried using OpenCV in python myself, but seeing as it's interpreted I doubt you'll have as much trouble (if any) getting them to work.

If you're using the C versions in linux, have a go at making the samples and see what happens. Simply cd to the folder and type:

```
sh ./build_all.sh
```

If you get errors then you'll need to put in a couple of extra lines first. Remember to substitute your directory in place of /home/opencv/, just like on the installation page.

```
export PKG_CONFIG_PATH="/home/opencv
/lib/pkgconfig"
export LD_LIBRARY_PATH="/home/opencv/lib"
sh ./build_all.sh
```

With any luck they'll start working straight away. If they don't then I recommend you consult the wiki.

This is what they do - starred items require a camera:

- ***camshiftdemo***
  Select areas of the window with the mouse. The histogram records the colours and an ellipse attempts to cover the area most closely matched.

- ***contours***
  This doesn't seem to work on my linux box, but in Windows it gives an example of contour fitting at various levels on a hard-coded image.

- ***convexhull***
  Generates a set of random points and constructs their convex hull.

- ***delaunay***
  Demonstration of Delaunay triangulation on randomly plotted points. Another that doesn't seem to work on linux.

- ***demhist***
  Demonstrates the changes caused to a histogram by altering the brightness and contrast of an image.

- ***distrans***
  Distance transform demo on 'stuff.jpg'.

- ***drawing***
  Demonstrates OpenCV's drawing abilities.

- ***edge***
  Uses a Canny edge detector to find the outline of various fruits.

- ***(*)facedetect***
  Usage: facedetect --cascade="../../data/haarcascades/" [optional file].

- ***ffilldemo***
  Fills fruit with different amounts of colour depending on the thresholds selected.

- ***fitellipse***
  Yet another that doesn't work on my linux box but does on Windows. Fits ellipses to the items in 'stuff.jpg'.

- ***kalman***

Performs Kalman filtering on a point rotating around the window.

- ***kmeans***
  Performs the K-means algorithm on randomly generated points. Press space bar to generate new sets.

- ****laplace***
  Calculates Laplacian of an image.

- ****lkdemo***
  Press 'r' to find corners in the camera image (using cvGoodFeaturesToTrack) and plot the points it finds on the viewer. 'c' clears the points and 'n' removes the image, showing only the points.
  Using the iterative Lucas-Kanade method in pyramids, it also performs optical tracking...all at 25fps.

- ***minarea***
  Fits a circle and a square of minimum area to a set of randomly generated points.

- ***morphology***
  A demonstration of the speed with which you can perform opening, closing, erosion and dilation.

- ****motempl***
  Shows how an image changes over time using motion segmentation, and plots the direction in which the change is happening using a motion gradient calculation.

- ***pyramid_segmentation***
  Performs segmentation on the fruit image using two adjustable threshold settings. (I'm sure there's a pun to be made about segmenting an orange...)

- ***squares***
  The last one, and again it doesn't work very well on my linux box. Press space to cycle through various images and adjust the square-fitting threshold using the slider at the top.

Owner: Drew Morgan | Last updated: Wednesday 13 September 2006