



LINUXCON

NORTH AMERICA 2012

Image Processing on Pandaboard using OpenCV and Kinect

8-29-12

Jayneil Dalal

About me

Jayneil Dalal is a FOSS advocate and loves to explore different open source technologies. His areas of interest include OpenCV, Python, Android, Linux, Human Computer Interaction, BeagleBoard, PandaBoard, Arduino and other open source hardware platforms. In his spare time he likes to make video tutorials on various different open source technologies which can be found on YouTube. He has also been selected to give a talk at Drodicon 2012 in Berlin and various other international conferences. He is also a regular writer for Linux For You Magazine, Linux Pro Magazine, Full Circle Magazine. He is a big time Arsenal fan.

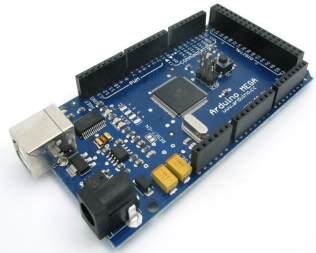


What you will learn?

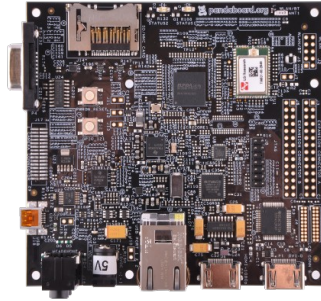


- Basics of Image Processing
- Pandaboard overview
- Installing Ubuntu on Pandaboard
- OpenCV overview
- Installing OpenCV on Pandaboard
- Image Processing using OpenCV
- Kinect overview
- Installing support for Kinect on Pandaboard
- Arduino
- Troubleshooting

Basic Idea



+



+



+



=

EUREKA!!!

Overview video



Below is a short video that I have made on a few cool hacks made using Kinect:

<http://db.tt/RxcPLTkK>

Motivation



- There have been so many awesome Kinect hacks performed in the world. But most of them use a computer/laptop which is not at all portable!
- So, why not start making the application on Pandaboard itself?
- More easier to commercialize the system because of the small form factor.
- Much cheaper!

What is Image Processing?

- In electrical engineering and computer science, image processing is any form of signal processing for which the input is an image, such as a photograph or video frame; the output of image processing may be either an image or, a set of characteristics or parameters related to the image. Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal-processing techniques to it. In other words it is basically the transformation of data from a still or video camera into either a decision or a new representation. All such transformations are done for achieving some particular goal. The input data may be a live video feed, the decision may be that a face has been detected, and a new representation may be conversion of a color image into a gray scale image.

Few Applications



- Automotive industry
- Security
- Medical
- Robotics

OpenCV



- OpenCV grew out of an Intel Research initiative to advance CPU-intensive applications.
- The intent behind OpenCV was to provide a platform that a student can readily utilize for developing applications, instead of reinventing the basic functions from scratch.

Why use OpenCV?



- It is free.
- It is fast.
- Good documentation in the form of documentation, tutorials, user groups, forums etc.
- Lot of pre-built functions and algorithms to get a head start.
- There is an active development on interfaces for other languages like Ruby, Python, Matlab etc.
- Pretty fast.

License



- OpenCV is released under a BSD license, it is free for both academic and commercial use.
- There are only a few algorithms inside it that may have been patented and those are not free.

What OS does OpenCV support?

- Linux
- Windows
- Mac
- Android

OpenCV 2.4.2 installation

Please Refer the OpenCV 2.4.2 installation guide that I have made which can be downloaded from the link below:

<http://db.tt/HBDrUs7n>

Configure OpenCV with codeblocks

Please Refer the OpenCV 2.4.2 installation guide that I have made which can be downloaded from the link below:

<http://db.tt/HBDrUs7n>

Codeblocks IDE can be downloaded from the link below:

<http://www.codeblocks.org/>

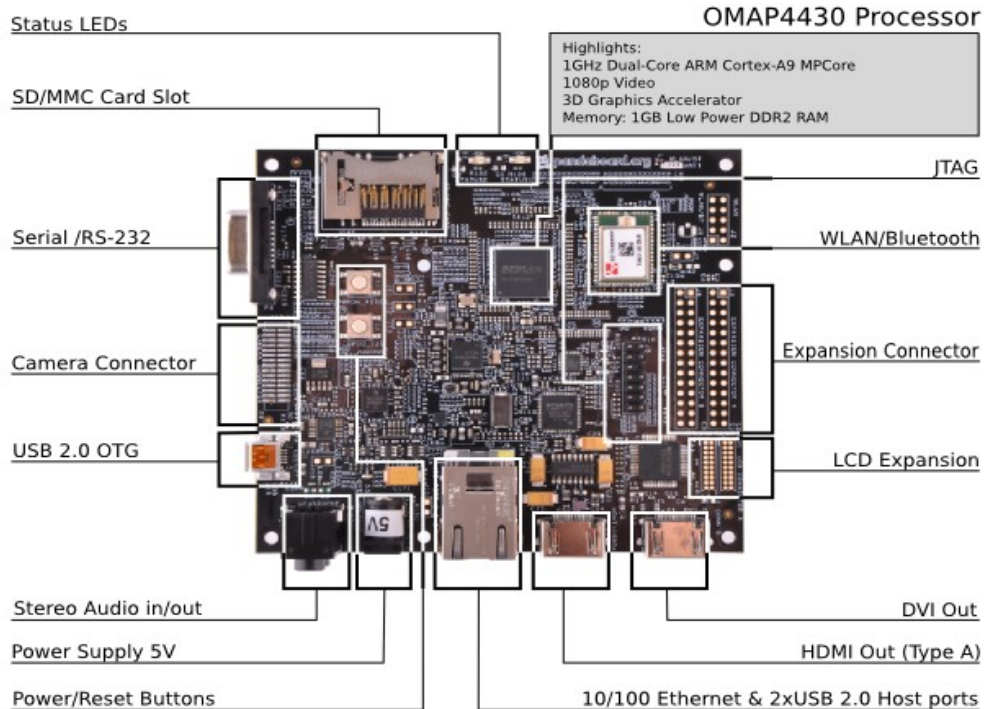
OpenCV 2.4.2 un-installation

Please Refer the OpenCV 2.4.2 installation guide that I have made which can be downloaded from the link below:

<http://db.tt/HBDrUs7n>

At the end of the above guide, the un-installation process will be explained.

Pandaboard



Board Dimensions: W:4.0" (101.6 mm) X H: 4.5" (114.3 mm)

Pricing



- Pandaboard: \$153
- Pandaboard ES: \$162

Pandaboard Overview

- OMAP4 (Cortex-A9) CPU based open development platform.
- OMAP4430 Application processor
- 1GB low-power DDR2
- Display HDMI v1.3 Connector (Type A) to drive HD displays, DVI-D Connector,
- 3.5" audio in/out and HDMI Audio out
- Full size SD/MMC card
- Built in 802.11 & Bluetooth v2.1+EDR
- On board 10/100 Ethernet
- Expansion: 1xUSB OTG, 2xUSB HS host ports, General purpose expansion header

Where to Buy?



<http://pandaboard.org/content/buy>

Few dealers that I recommend: SVTronics, Tigal.

What OS does Pandaboard support?



- Debain
- Ubuntu
- Android
- Fedora
- Sabayon
- Xubuntu
- Angstrom

Pandaboard accessories

- HDMI/DVI cable to see the output on a monitor.
- 8Gb/16Gb SDHC or micro sd card(then use it using a converter).
- A 5V power supply
- USB hub
- Serial to USB Cable
- Keyboard
- Mouse
- Laptop cooler(so that the pandaboard does not heat up)

Installing Ubuntu on Pandaboard

Please Refer the 'Installing Ubuntu 12.04 on Pandaboard' installation guide that I have made which can be downloaded from the link below:

<http://db.tt/EmDq0Vyo>

Performance Tuning

Speeding up the Pandaboard

There are a few tweaks that we can perform to further increase the performance of Ubuntu on Pandaboard. First we need to disable swap which is nothing but virtual RAM. So, edit out the swap file line found in `/etc/fstab` i.e. put a '#' in front of the line which creates a swapfile.

Now, we will use the tmpfs filesystem which use RAM and swap and hence provides much faster response and access compared to accessing from the SD card. This is achieved by editing `/etc/fstab` and add the following lines:

```
-----CODE-----  
tmpfs /tmp      tmpfs nodev,nosuid      0 0  
tmpfs /var/log  tmpfs nodev,nosuid      0 0  
-----CODE-----
```

Performance Tuning

Installing Xubuntu

Once you install Ubuntu 12.04 and start using it on Pandaboard, you will feel that it is quite slow. So, we need a lightweight desktop environment unlike Xfce. Now Xubuntu is also a derivative of Ubuntu but with Xfce as the desktop environment and some other applications. So, now we will see how to install Xubuntu on top of Ubuntu. Open the terminal and type the following code:

```
-----CODE-----  
sudo apt-get update  
sudo apt-get install xubuntu-desktop  
-----CODE-----
```

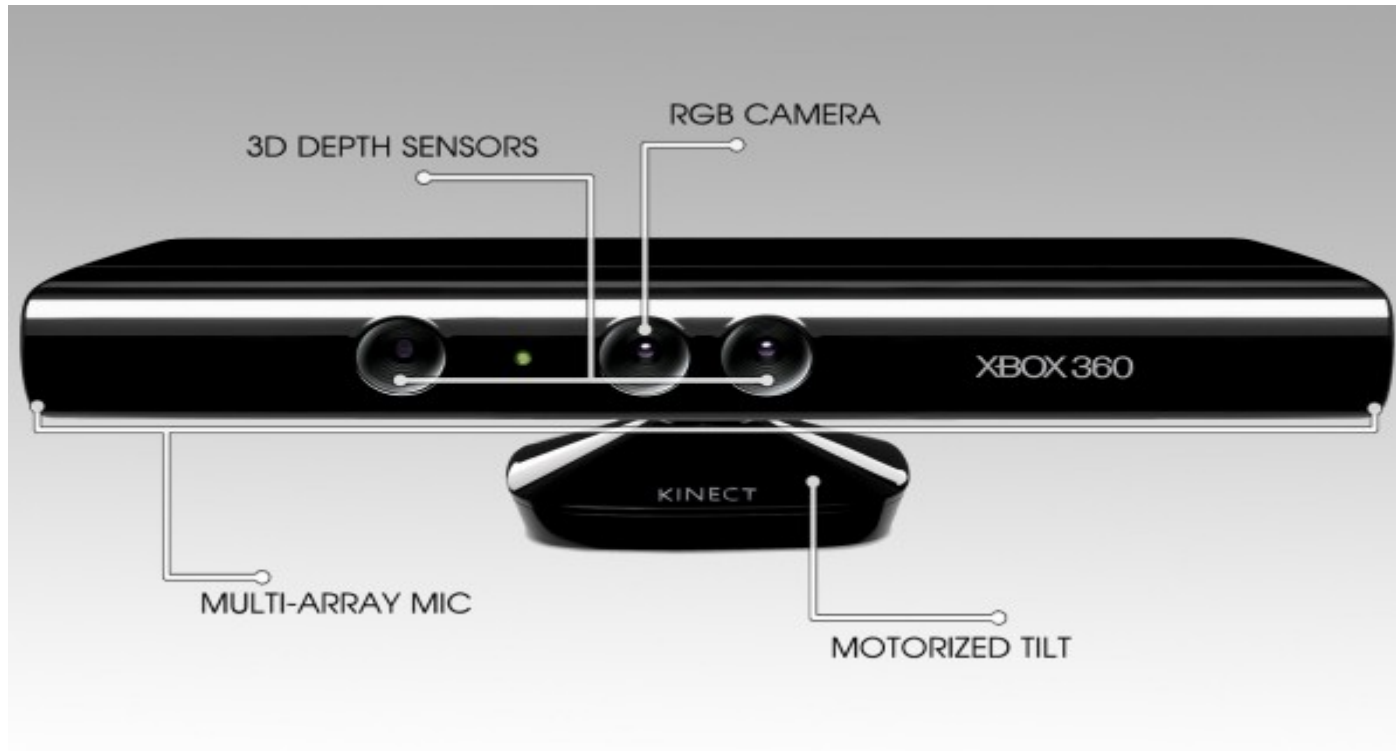

Performance Tuning



Use an external Solid State Drive(SSD)

Currently, sd card is used to boot the operating system on pandaboard. Hence the speed is slow. A better option would be to boot the operating on pandaboard from an external SSD.

Kinect



Kinect Overview

- a VGA 640x480 color camera (CMOS) with a Bayer color filter
 - a IR 640x480 camera (CMOS)*
 - an IR projector
 - a motor, an assortment of control chips and 4 microphones
 - a fan!
- (* it seems the outputted depth-map is 640x480 but the IR camera sensor size is closer too 1600x1200)

Install OpenNI and Kinect driver in Ubuntu running on Pandaboard

Please Refer the 'Installing OpenNI+Kinect driver in Ubuntu 12.04 running on Pandaboard' installation guide that I have made which can be downloaded from the link below:

<http://db.tt/1rJpon2h>

Arduino



Arduino

To make any changes in the external environment we need a hardware circuit with a 'brain'. This role is can be performed by any micro controller or micro processor. Various different companies such as Texas Instruments, NXP, Maxim, Atmel etc. make micro controllers. A micro controller cannot be used directly in a circuit in its original form. Following things have to be done before:

- A full development circuit board has to be made providing access to different pins of the micro controller. This process is not easy as it sounds. It involves designing the circuit, soldering. All the changes once made cannot be reverted.
- A program loader has to be made to load the code from the computer into the micro controller.

So the above procedures can be quite cumbersome. As a result, Arduino is used due to the following reasons:

- Open Source. So all the schematics are available and one can design the same board on their own.
- Concept of breadboard prototyping which encourages the use of hook up wires and breadboard to make a circuit which can be easily modified later. This saves all the pain of soldering.
- Lot of tutorials, tons of codes, forums etc. to help out beginners.
- The design of the board and IDE is so simple that one does not have to be an engineer to use it. Even a 6th grade student can use it.
- Pretty cheap compared to other available development boards.

There are various different versions of Arduino to choose from such as Uno, Mega, LilyPad etc. ATmega1280 has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the micro-controller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

Demos on Pandaboard



I will now show the following demos on the Pandaboard:

- Using Kinect as a normal camera
- Getting the depth data from Kinect in OpenCV
- Live Edge detection using OpenCV and Kinect
- Face detection using OpenCV and Kinect
- Using Arduino along with Kinect and OpenCV

Learning Resources

OpenCV:

- Refer the documentation and tutorial guides provided:
[/OpenCV-2.4.2/doc](#)
- Books
 - Learning OpenCV: Computer Vision with the OpenCV Library
 - OpenCV 2 Computer Vision Application Programming Cookbook

Pandaboard:

- http://omappedia.org/wiki/Main_Page
- <http://pandaboard.org/>

Arduinio:

- <http://arduino.cc/playground/>

Going Beyond



- PCL(Point Cloud Library)
<http://pointclouds.org/>
- ROS(Robotic Operating System)
<http://pointclouds.org/>

Contact

- Shoot me an email at:
jayneil.dalal@gmail.com



THANK YOU