

Edge Detection Using Different Algorithms in Raspberry Pi

Jignesh K. Tank¹ Prof. Vandana Patel²

¹M.E Student

¹L. D. College of Engineering, Ahmedabad

Abstract— Industry is using edge detection technique to automate the process for increasing efficiency by eliminating defects or malfunctioning of any system. Edge detection system is composed of multi-disciplinary subcomponent via embedded systems and software algorithm. The Embedded system is developed using many different architectures and design philosophy. The accuracy, speed and scalability is the factor which decides this design. In this paper, I will focus on different software algorithms, which were developed to detect the edge. This will include configuration of development IDE and some python programs to detect edge from camera.

I. INTRODUCTION

A software algorithm in embedded system used for image processing is critical, as this will decide the accuracy, speed & performance of entire system. This embedded system should have easy interface, scalability & support for end libraries for image processing. For this paper, Raspberry Pi is used as basic embedded system. Configuration and usage of Raspberry Pi is mainly highlighted in this paper. Some algorithms for edge detection is used and result has been presented.

II. EDGE TERMINOLOGY

There exist many different terminology of edge in image processing viz.

- 1) Relationship a pixel has with its neighbors.
- 2) Boundary between an object and the background.
- 3) Significant variation in the gray level or color of pixel in some direction

So basically, from all above definition, edge is change in gray level in image.

III. TECHNIQUES FOR EDGE DETECTION

Edge has largest rate of change of the gray levels in an image and its really constant in other areas. Gradient/Derivative operators can used to detect the edge. If image is 2D and is denoted $A(x,y)$, the gradient can found by $dA(x,y) = (dA/dx, dA/dy)$

Edge can be detected by detecting the points, which lie on it. It can be done by

- 1) Detecting the local Maxima or minima of the first derivative.

OR

- 2) Detecting the zero-crossing of the second derivative.

Derivative method is widely used to detect the edge.

IV. DIFFERENT ALGORITHMS FOR EDGE DETECTION

A. Canny edge detection [1]

- 1) Sobel edge detection [2]

- 2) Fuzzy Logic [3]
- 3) Segmentation [4]
- 4) Laplace
- 5) Image Smoothing

Any of the above method can be used to detect the image

V. STEPS FOR EDGE DETECTION

Canny has given the following steps for edge detection

- 1) Noise filtration and Gaussian filter is used for this purpose
- 2) For detecting the intensity gradient of the image, Canny algorithm uses filters to detect horizontal, vertical and diagonal edges in the blurred image.
- 3) Non-maximum suppression is applied by removing the pixels that are not considered to be part of an edge and hence, only thin lines (candidate edges) will remain.
- 4) Hysteresis: The final step. Canny does use two thresholds (upper and lower):
 - i. If a pixel gradient is higher than the upper threshold, the pixel is accepted as an edge
 - ii. If a pixel gradient value is below the lower threshold, then it is rejected.
 - iii. If the pixel gradient is between the two thresholds, then it will be accepted only if it is connected to a pixel that is above the upper threshold.

VI. SYSTEM SPECIFICATION AND CONFIGURATION

- 1) Easy interface to camera (USB)
- 2) Large Memory
- 3) Floating Point Operation
- 4) Reliable Operating system
- 5) Speed
- 6) Preferably, capable of installing Webserver
- 7) Directly connect to display

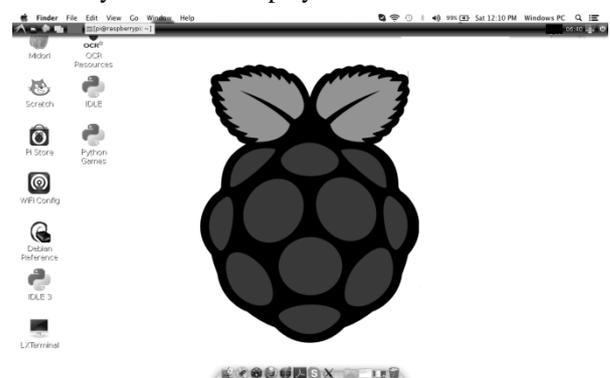


Fig. 1: Graphical interface of Raspberry Pi

A. Installation of OS on Raspberry Pi [6]

There are many different OS, which will be supported by raspberry pi. We have selected the original OS called

Raspbian “wheezy”. The Raspberry Pi has SD card slot, in which, any SD card having OS image can work.

B. Display configuration

Raspberry Pi has HDMI output, so any display, which is HDMI, can be connected directly to Raspberry Pi. I was not having any HDMI output display so need to figure out some other method. I have used SSH [7] for connecting MacBook as shown in Fig. 1.

So the Raspberry Pi is connected to Macbook Pro without any HDMI display.

C. Integrating Development IDE in Raspberry Pi [8]

In order to develop the software for edge detection, it is required to install integrated development environment (IDE) in the raspberry Pi. I have used Geany for the developing Python software. For installing this, following is procedure.

- 1) In menu → accessory → LXTerminal, it will open terminal in Raspberry Pi graphical interface
- 2) Type `sudo apt-get install Geany`
- 3) That's it. Open the Geany from menu → Programming → Geany. This will give graphical IDE for developing Python programs.

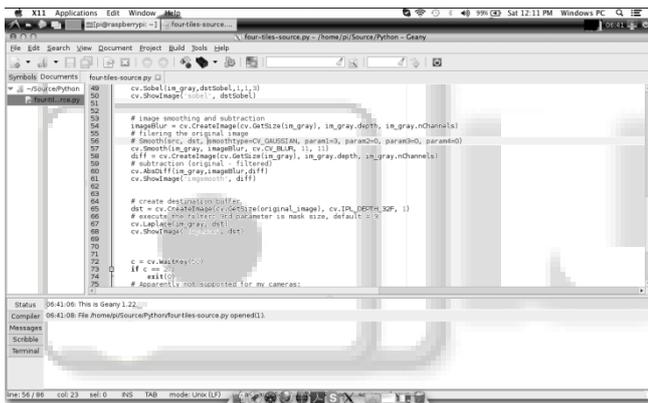


Fig. 2: Geany IDE in Raspberry Pi

D. OpenCV Installation [9]

Installing OpenCV in Raspberry Pi requires many different dependencies to be installed first. Following procedure was used to install OpenCV in Raspberry Pi

a) In menu → accessory → LXTerminal, it will open terminal in Raspberry Pi graphical interface

- 1) `sudo apt-get install build-essential`
- 2) `sudo apt-get install cmake`
- 3) `sudo apt-get install pkg-config`
- 4) `sudo apt-get install libpng12-0 libpng12-dev libpng++-dev libpng3`
- 5) `sudo apt-get install libpnglite-dev libpngwriter0-dev libpngwriter0c2`
- 6) `sudo apt-get install zlib1g-dbg zlib1g zlib1g-dev`
- 7) `sudo apt-get install pngtools libtiff4-dev libtiff4 libtiffxx0c2 libtiff-tools`
- 8) `sudo apt-get install libjpeg8 libjpeg8-dev libjpeg8-dbg libjpeg-progs`
- 9) `sudo apt-get install ffmpeg libavcodec-dev libavcodec52 libavformat52 libavformat-dev`

- 10) `sudo apt-get install libgstreamer0.10-0-dbg libgstreamer0.10-0 libgstreamer0.10-dev`
- 11) `sudo apt-get install libxine1-ffmpeg libxine-dev libxine1-bin`
- 12) `sudo apt-get install libunicap2 libunicap2-dev`
- 13) `sudo apt-get install libdc1394-22-dev libdc1394-22 libdc1394-utils`
- 14) `sudo apt-get install swig`
- 15) `sudo apt-get install libv4l-0 libv4l-dev`
- 16) `sudo apt-get install python-numpy`
- 17) `sudo apt-get install libpython2.6 python-dev python2.6-dev`
- 18) `sudo apt-get install libgtk2.0-dev pkg-config`

b) Get the OpenCV from the server on Raspberry Pi using following command

- 1) `wget http://sourceforge.net/projects/opencvlibrary/files/opencv-unix/2.3.1/OpenCV-2.3.1a.tar.bz2/download`

c) Make, compile and install the OpenCV using following command

- 1) `cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local -D BUILD_PYTHON_SUPPORT=ON -D BUILD_EXAMPLES=ON ..`
- 2) `make`
- 3) `sudo make install`

d) This compilation and installation will take more than 12Hours.

VII. EDGE DETECTION WORKING RESULT

I have used python for the scripting. Please the below result images in which edge is detected using USB camera connected with Raspberry Pi.

Different algorithms were used to detect the edge. I have used following algorithms:

- 1) Canny
- 2) Sobel
- 3) Laplace
- 4) Image Smoothing

Different algorithms were written in Python using Geany as IDE. In order to compare the results, python script was

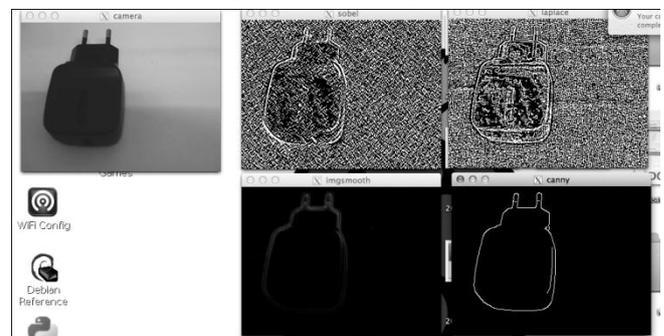


Fig. 3: Result comparison for different edge detection algorithms

Written to show the real camera image and edge detection using different algorithms. All screens were shown at a time on screen for comparison.

VIII. CONCLUSION

Raspberry Pi OS was installed successfully on Raspberry Pi Kit. The visual graphical interface was generated using SSH. Python program for developed for generating the different algorithms for edge detection. It was found out that Canny detection is giving more accurate result than other edge detection technique.

ACKNOWLEDGEMENT

I would like to thanks Prof. V. V. Patel (HOD, A.I. Department, L.D. College of Engineering, Ahmedabad) for his guidance and support during my M.E course work. I would also like to thanks Prof. Vandana Patel, my guide and entire staff of A.I. Department, L.D. College of Engineering, Ahmedabad for their great help and support.

REFERENCES

- [1] John Canny, A Computational Approach to Edge Detection: IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. PAMI-8, NO. 6, NOVEMBER 1986
- [2] Wenshuo Gao, An Improved Sobel Edge Detection: 978-1-4244-5540-9/10/\$26.00 ©2010 IEEE
Aborisade, D.O, Novel Fuzzy logic Based Edge Detection Technique: International Journal of Advanced Science and Technology Vol. 29, April, 2011
- [3] Y.Ramadevi, SEGMENTATION AND OBJECT RECOGNITION USING EDGE DETECTION TECHNIQUES at International Journal of Computer Science & Information Technology (IJCSIT), Vol 2, No 6, December 2010
- [4] R. Owens, "Lecture 6", Computer Vision IT412, 10/29/1997.
http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT6/node2.html
- [5] Ehsan Nadernejad, Edge Detection Techniques: Evaluations and Comparisons: Applied Mathematical Sciences, Vol. 2, 2008, no. 31, 1507 – 1520 Edge Detection Techniques: Evaluations and Comparisons
- [6] <http://www.raspberrypi.org/>
- [7] http://en.wikipedia.org/wiki/Secure_Shell
- [8] <http://learn.adafruit.com/adafruit-raspberry-pi-lesson-1-preparing-and-sd-card-for-your-raspberry-pi/overview>
- [9] <http://mitchtech.net/raspberry-pi-opencv/>