IMAGE AND VIDEO PROCESSING ON RASPBERRY PI PLATFORM

Ph.D. Mustafa Karhan

Çankırı Karatekin University,
Electronics and Automation Department, Çankırı, Turkey
mustafakarhan@gmail.com

Timisoara, Romania
July, 2019
Computer vision at the intersection of multiple scientific fields
Single Board Computers (SBCs)

A single board computer is a fully functional computer system built around a single printed circuit board. An SBC has a microprocessor(s), memory, input/output, and other features required of a minimally functioning computer.

A few popular SBCs for Embedded Computer Vision

- Raspberry Pi
- Beaglebone
- Beagleboard
- Orange Pi
- Banana Pi
- Intel Edison
- Intel Galileo

Differences Between SBCs and Regular Computers

<table>
<thead>
<tr>
<th>Single Board Computer</th>
<th>Regular Computer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not modular</td>
<td>Modular</td>
</tr>
<tr>
<td>Components cannot be upgraded or replaced</td>
<td>Components can be upgraded or replaced</td>
</tr>
<tr>
<td>A System On Chip</td>
<td>Not a System On Chip</td>
</tr>
<tr>
<td>Has a small form factor</td>
<td>Has a large form factor</td>
</tr>
<tr>
<td>Is portable</td>
<td>Is mostly non-portable or semi-portable</td>
</tr>
<tr>
<td>Consumes less power</td>
<td>Consumes more power</td>
</tr>
<tr>
<td>Cheaper than a regular computer</td>
<td>Costs more than a SBC</td>
</tr>
</tbody>
</table>
Raspberry Pi

The Raspberry Pi is a credit-card-sized single-board mini computer developed in the UK. Raspberry Pi provide opportunities for open source software based projects. So it is the preferred in open electronics.

This popular development board is used as educational, commercial and academic. The aim behind developing Raspberry Pi was to promote the teaching of basic computer science in schools and developing countries by providing a low-cost computing platform.

https://www.raspberrypi.org

# Raspberry Pi New Versions

![Raspberry Pi 4 Model B](image)

<table>
<thead>
<tr>
<th></th>
<th>Raspberry Pi 3 B+</th>
<th>Raspberry Pi 4 Model B</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Processor</strong></td>
<td>Broadcom BCM2837B0, Quad-core Cortex-A53 64-bit SoC @ 1.4GHz</td>
<td>Broadcom 2711, Quad-core Cortex-A72 64-bit SoC @ 1.5GHz</td>
</tr>
<tr>
<td><strong>Memory</strong></td>
<td>1GB LPDDR2 SDRAM</td>
<td>1GB, 2GB or 4GB LPDDR4 SDRAM</td>
</tr>
<tr>
<td><strong>Connectivity</strong></td>
<td>2.4GHz / 5.0GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE, 4 x USB 2.0 ports, Gigabit Ethernet over USB2.0 (max. 300Mbps)</td>
<td>2.4GHz / 5.0GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 5.0, BLE, 2 x USB 2.0 / 2 x USB 3.0 ports, True Gigabit Ethernet</td>
</tr>
<tr>
<td><strong>Video</strong></td>
<td>1 x full size HDMI</td>
<td>2 x micro HDMI, 4K video</td>
</tr>
<tr>
<td><strong>Multimedia</strong></td>
<td>H264, MPEG-4 decode (1080p30), H264 encode (1080p30), OpenGL ES 1.1, 2.0 graphics</td>
<td>H265 decode (4k@60), H264 decode (1080p60), H264 encode(1080p30), OpenGL ES 1.1, 2.0, 3.0 graphics</td>
</tr>
<tr>
<td><strong>Input Power</strong></td>
<td>5V/2.5A DC via micro USB connector</td>
<td>5V/3A DC via USB type C connector</td>
</tr>
</tbody>
</table>
### Raspberry Pi Camera, USB Camera and Digital Microscope for Computer Vision

<table>
<thead>
<tr>
<th>Camera Module v1</th>
<th>Camera Module v2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Net price</strong></td>
<td>$25</td>
</tr>
<tr>
<td><strong>Size</strong></td>
<td>Around 25 x 24 x 9 mm</td>
</tr>
<tr>
<td><strong>Weight</strong></td>
<td>9g</td>
</tr>
<tr>
<td><strong>Still resolution</strong></td>
<td>8 Megapixels</td>
</tr>
<tr>
<td><strong>Video modes</strong></td>
<td>1080p30, 720p60 and 640x480p60/90</td>
</tr>
<tr>
<td><strong>Linux integration</strong></td>
<td>V4L2 driver available</td>
</tr>
<tr>
<td><strong>C programming API</strong></td>
<td>OpenMAX IL and others available</td>
</tr>
<tr>
<td><strong>Sensor</strong></td>
<td>OmnimVision OV5647</td>
</tr>
<tr>
<td><strong>Sensor resolution</strong></td>
<td>2592 x 1944 pixels</td>
</tr>
<tr>
<td><strong>Sensor image area</strong></td>
<td>3.76 x 2.74 mm</td>
</tr>
<tr>
<td><strong>Pixel size</strong></td>
<td>1.4 µm x 1.4 µm</td>
</tr>
<tr>
<td><strong>Optical size</strong></td>
<td>1/4&quot;</td>
</tr>
<tr>
<td><strong>Full-frame SLR lens equivalent</strong></td>
<td>35 mm</td>
</tr>
<tr>
<td><strong>S/N ratio</strong></td>
<td>36 dB</td>
</tr>
<tr>
<td><strong>Dynamic range</strong></td>
<td>67 dB @ 8x gain</td>
</tr>
<tr>
<td><strong>Sensitivity</strong></td>
<td>680 mV/ lux·sec</td>
</tr>
<tr>
<td><strong>Dark current</strong></td>
<td>16 mV/sec @ 60 °C</td>
</tr>
<tr>
<td><strong>Well capacity</strong></td>
<td>4.3 Ke-</td>
</tr>
<tr>
<td><strong>Fixed focus</strong></td>
<td>1 m to infinity</td>
</tr>
<tr>
<td><strong>Focal length</strong></td>
<td>3.60 mm +/- 0.01</td>
</tr>
<tr>
<td><strong>Horizontal field of view</strong></td>
<td>63.60 +/- 0.13 degrees</td>
</tr>
<tr>
<td><strong>Vertical field of view</strong></td>
<td>41.41 +/- 0.11 degrees</td>
</tr>
<tr>
<td><strong>Focal ratio (F-Stop)</strong></td>
<td>2.0</td>
</tr>
</tbody>
</table>
Operating Systems for Raspberry Pi


Raspbian is our official operating system for all models of the Raspberry Pi. Download it here, or use NOOBS, our easy installer for Raspbian and more.

Third Party Operating System Images

Third-party operating system images for Raspberry Pi are also available:

- Ubuntu MATE
- Ubuntu Core
- Ubuntu Server
- Windows 10 IoT Core
- OSMC
- LibreELEC
- PINE
- RISC OS
- Weather Station
- IchigoJam
- IchigoJam RPi
Linux Commands for Raspberry Pi

- **apt-get update**
- **apt-get upgrade**
- **apt-get install**
- **date**
- **nano galatasaray.py**
- **raspi-config**
- **reboot**
- **startx**
- **df**
- **pwd**
- **wget http://www.web.com/a.txt**
- **ls**
- **ls -l**
- **lsusb**
- **cat**
- **cd**
- **mkdir**
- **rmdir**
- **rm**
- **cp**
- **ifconfig**
- **ping**
- **mv**

**raspistill**

is the command line tool for capturing still photographs with the camera module.

```
raspistill -t 5000 -o hamza.jpg -w 300 -h 500
%%Take a capture after 5s resolution 300x500
```

**raspivid**

is the command line tool for capturing video with the camera module.

```
raspivid -t 10000 -o video.h264 -f 5
%%Record a 10s clip at a specified framerate (5fps)
```
Platforms for Image and Video Processing

MATLAB®

OpenCV

Simulink

Python

SimpleCV

e等
With MATLAB support package for Raspberry Pi, the Raspberry Pi is connected to a computer running MATLAB. Processing is done on the computer with MATLAB.

With Simulink support package for Raspberry Pi, you develop the algorithm in Simulink and deploy to the Raspberry Pi using automatic code generation. Processing is then done on the Raspberry Pi.
MATLAB / SIMULINK Hardware Support Packages for Raspberry Pi
MATLAB / SIMULINK
Hardware Support Packages for Raspberry Pi
clear rpi
rpi = raspi();
%Create a connection from the MATLAB software to the Raspberry Pi
cam = cameraboard(rpi,'Resolution','640x480');
%Create a connection, cam, from the MATLAB software to the camera board, and set the image resolution
img = snapshot(cam)
record(cam,'hamza.h264',30)
%Record a 30 seconds video.
stop(mycam)
%Stop the recording immediately.
Basic Example Simulink Model
Basic Example Simulink Model
OpenCV (Open Source Computer Vision) is a library of programming functions for computer vision.

OpenCV has interfaces for popular programming languages, such as C/C++, Python, and Java, and OpenCV runs on a variety of operating systems including Windows, Android, and Unix-like operating systems.
sudo apt-get update
sudo apt-get upgrade
sudo rpi-update
sudo reboot

sudo apt-get install build-essential cmake cmake-curses-gui pkg-config

sudo apt-get install \
  libjpeg-dev \
  libtiff5-dev \
  libjasper-dev \
  libpng12-dev \
  libavcodec-dev \
  libavformat-dev \
  libswscale-dev \
  libeigen3-dev \
  libxvidcore-dev \
  libx264-dev \
  libgtk2.0-dev

sudo apt-get install python2.7-dev python2-numpy
sudo apt-get install python3-dev python3-numpy

mkdir /home/pi/opencv
cd /home/pi/opencv

wget https://github.com/opencv/opencv/archive/3.2.0.zip -O opencv_source.zip
wget https://github.com/opencv/opencv_contrib/archive/3.2.0.zip -O opencv_contrib.zip
Install OpenCV on Raspberry Pi

cmake -D CMAKE_BUILD_TYPE=RELEASE \ 
    -D CMAKE_INSTALL_PREFIX=/usr/local \ 
    -D BUILD_WITH_DEBUG_INFO=OFF \ 
    -D BUILD_DOCS=OFF \ 
    -D BUILD_EXAMPLES=OFF \ 
    -D BUILD_TESTS=OFF \ 
    -D BUILD_opencv_ts=OFF \ 
    -D BUILD_PERF_TESTS=OFF \ 
    -D INSTALL_C_EXAMPLES=ON \ 
    -D INSTALL_PYTHON_EXAMPLES=ON \ 
    -D OPENCV_EXTRA_MODULES_PATH=../../opencv_contrib-3.2.0/modules \ 
    -D ENABLE_NEON=ON \ 
    -D WITH_LIBV4L=ON \ 

../

make -j4

sudo make install
sudo ldconfig

There is an installation process of approximately 3 hours.
Your disk must have at least 2GB of space to complete the installation.
import cv2
import numpy as np
import os

def main():
    imgOriginal = cv2.imread("image.jpg")

    if imgOriginal is None:
        print("Error!! There is no Image \n\n")
        os.system("pause")
        return

    imgGrayscale = cv2.cvtColor(imgOriginal, cv2.COLOR_BGR2GRAY)

    imgBlurred = cv2.GaussianBlur(imgGrayscale, (5, 5), 0)

    imgCanny = cv2.Canny(imgBlurred, 100, 200)

    cv2.imshow("imgOriginal", imgOriginal)
    cv2.imshow("imgCanny", imgCanny)

    cv2.waitKey()

    cv2.destroyAllWindows()

    return

if __name__ == "__main__":
    main()
import numpy as np
import cv2
from matplotlib import pyplot as plt

img = cv2.imread('/home/pi/mustafa/deneme/hagi.jpg',0)
kern = np.ones((5,5),np.uint8)
erosion = cv2.erode(img,kern,iterations = 2)
dilation = cv2.dilate(img,kern,iterations = 2)
opening = cv2.morphologyEx(img, cv2.MORPH_OPEN, kern)
titles=['Original','Erosion','Dilation','Opening']
output=[img,erosion,dilation,opening]

for i in xrange(4):
    plt.subplot(2,2,i+1),plt.imshow(output[i],cmap='gray')
    plt.title(titles[i]),plt.xticks([]),plt.yticks([])
plt.show()
Basic Example Code
SimpleCV

- SimpleCV is an open source framework for building computer vision applications.
- It is a collection of libraries and software that you can use to develop vision applications.
- SimpleCV is written in Python, and it's free to use. It runs on Mac, Windows, and Ubuntu Linux, and it's licensed under the BSD license.

```bash
sudo apt-get install ipython python-opencv python-scipy python-numpy python-setuptools python-pip

sudo pip install https://github.com/sightmachine/SimpleCV/zipball/master

mkdir ~/Code
cd ~/Code
git clone git://github.com/sightmachine/SimpleCV.git
cd SimpleCV
sudo pip install -r requirements.txt
sudo python setup.py develop
```
from SimpleCV import Camera
# Initialize the camera
cam = Camera()
# Capture and image and display it
cam.getImage().show()

from SimpleCV import Camera
cam = Camera()
cam.live()

%To get live video feed from the camera

from SimpleCV import Image
img = Image('starry_night.png')
hsv = img.toHSV()
print hsv.getPixel(25,25)
rgb = hsv.toRGB()
print rgb.getPixel(25,25)
Basic Example Code

from SimpleCV import Image
img = Image('karasevda.jpg')
imgBin = img.binarize()
imgBin.dilate().show()
imgBin.erode().show()
Applications of Image Processing using Raspberry Pi

✓ Object Detection
✓ Object Recognition
✓ Object Categorization
✓ Image Segmentation
✓ Image Acquisition
✓ Image Restoration
✓ Video Processing
✓ Pattern Recognition
✓ Medical Imaging
✓ Image Restoration
✓ Image Sharpening
✓ Robot Vision
✓ Microscopic Imaging and etc..
Implementations using Raspberry Pi

Image and Video Processing

Raspberry Pi

Academic and Commercial Studies
Smart Aging Test Platform and Water Tree Microscopic Image Analysis using Raspberry Pi

Smart Aging Test Platform

Water Treeing Image

Histograms of Oriented Gradients

Steps of MFCC for microscope images

HOG - Histograms of Oriented Gradients

CoHOG - Co-occurrence Histograms of Oriented Gradients
Contact Angle Measurement using Raspberry Pi

Video streaming

Snapshot image

Image to be processed

Calculation image processing
Contact Angle Measurement under HVDC (10kV) using Raspberry Pi

Video streaming  
Snapshot image  
Image to be processed  
Calculation image processing
Color Based Segmentation of Rusted Areas in Sandblasting Process Using Raspberry Pi

Sample to be sandblasted

Segmentation using K-means algorithm
References

• http://www.mathworks.com
• https://www.raspberrypi.org
• https://www.opencv.org
• http://www.simplecv.org
"Thanks for your attention"

"Mulțumesc pentru atenție "