Raspberry Pi Camera

Written by Mark Webster and ??

Summary: Learn to capture and display still images and videos with the Raspberry Pi

Introduction

A common use for single board computers involves capturing images or videos. For example, a home security system, or a time lapse camera to monitor 3D printers, or a blind spot camera on a car, or a baby monitor.

The Raspberry Pi has a ribbon connector for a raspberry pi camera. This is the easiest and fastest image capture.A specialized case or 3D printed structure will be needed to hold the camera.

The Raspberry Pi Zero also has a camera

The Raspberry Pi camera can be for regular light or low light (NoIR, no IR filter so some infrared comes though). There are macro lense cameras, fish-eye wide angle, and zoom.

(Night vision camera)

An older Raspberry Pi with mount for the camera.

USB cameras can also be used.

(mini USB camera)

Note: before using the camera the camera interface must be enabled using the raspi-config preferences setting interface, either command line or GUI.

Still Images

From the terminal command line, images can captured using the raspistill command.
For example:

```
raspistill -hf -vf -o myPhoto.jpg
```

Will take a default 5 megapixel image (8 meg on the V2 version of the camera), flip it horizontally (-hf) and flip it vertically (-vf) and store the image in a JPEG compressed file called “myPhoto.jpg”

There are various options available for raspistill
--nopreview   (don’t display a preview window)
--preview <'x,y,w,h'> which sets the x,y position, width and height of the preview window
--fullscreen (set the preview window to full screen)
--e (file encoding. PNG, JPG, BMP… The program may guess the encoding type from the file name extension)

Many options exist such as setting contrast, night mode, special effects like solarize, image size, file type, etc. Some can be found in the PDF
https://www.raspberrypi.org/app/uploads/2013/07/RaspiCam-Documentation.pdf

The “man” option may also provide the documentation you need, in more cryptic format.

Raspistill can be called within a python program when doing timelapse or special effects, or when triggering a photo from a button or remote interface like wifi.

Still images can be manipulated from the command line with imagemagic

Timelapse: (from the raspberry pi organization documentation)

The raspistill application has a built in time-lapse mode, using the --timelapse (or -tl) command line switch.

The value that follows the -tl switch is the time between shots in milliseconds (2000 ms in this example). The -t number is the total time to capture images (30000 ms in this example)

```
raspistill -t 30000 -tl 2000 -o image%04d.jpg
```

Note the %04d in the output filename: this indicates the point in the filename where you want a frame count number to appear. So, for example, the command above will produce a capture every two seconds (2000ms), over a total period of 30 seconds (30000ms), named image0001.jpg, image0002.jpg, and so on, through to image0015.jpg.

From the file manager, select the image and then right click to view image.

To display images on the Raspberry Pi from the command line, many utilities exist. The default viewer on the Raspberry Pi is gpicview, but some use EOG (eye of Gnome)

```
gpicview myStillpic.jpg
```

Exercise 1: Raspistill Command Line

Open a terminal window and take a picture using raspistill. Take a selfie and store it with your name. Capture an image using PNG and BMP formats. Compare the sizes. Create an image with the -ifx sketch   special effect.
For example:

```
raspistill -hf -vf -t 5 -o myPicture.jpg
```

Take a still image with a 5 second delay before capture. Flip the image horizontally and vertically. Save it in the file myPicture.jpg with the JPEG compression encoding.

Exercise 2: USB camera

https://www.raspberrypi.org/documentation/usage/webcams/

To use a usb webcamera instead of the raspberry pi camera interface, connect the USB camera to a USB port on the raspberry pi. Make sure the fswebcam software is installed.

```
sudo apt-get install fswebcam
```

Then capture an image with the command line. To

```
fswebcam myPicture.jpg
```

The fswebcam software has various options such as: -r 1280x720 --no-banner
-r defines the resolutions (default is what the webcam can provide), and --no-banner leaves off the timestamp banner at the bottom of the picture.

More options for fswebcam are at:
http://manpages.ubuntu.com/manpages/bionic/man1/fswebcam.1.html

Exercise 3: Raspistill Python3

Within a Python3 program, call the command line raspistill.

```
#execute a command line call from within python program
import subprocess

cmd = "raspstill -vf -o yourFile.jpeg"
subprocess.call(cmd, shell=True)
```

Exercise 4: PiCamera

Make sure the picamera module is installed. If not, with an internet connection from the command line type:

```
sudo apt-get update
sudo apt-get install python-picamera python3-picamera
```

```
#python program to take a picture after the camera warms up
from time import sleep
from picamera import PiCamera

camera = PiCamera()
camera.resolution = (1024, 768)
camera.start_preview()
# Camera warm-up time
sleep(2)
camera.capture('foo.jpg')
```

Exercise 5: Timelapse within Python

```
from time import sleep
from picamera import PiCamera

camera = PiCamera()
camera.start_preview()
sleep(2)
```

```
for i in range(5):
   sleep(1)
   camera.capture('image{0:04d}.jpg'.format(i))
```

Video

The command line raspivid will capture video to a file.
https://www.raspberrypi.org/documentation/usage/camera/raspicam/raspivid.md

Many of the raspistill options apply to raspivid, such as -hf for horizontal flip.

Raspivid encodes into H264 but many video players cannot handle raw h264 so it must be wrapped in an mp4 container.

```
sudo apt-get install -y gpac
```

Will install the MP4Box utility which will put the output into an MP4 container.

Example command line statements to use MP4Box:

```
raspivid -t 10 -fps 25 -o pivideo.h264
MP4Box -add pivideo.h264 pivideo.mp4
rm pivideo.h264
```

To play video on the Raspberry Pi at the command line use the omxplayer command. For example:

```
omxplayer myVideo.mp4
```

This will play the example.mp4 in full screen. Hit Ctrl + C to exit.

The newest Raspbian has VLC video viewer installed for a GUI video player.

Exercise 6: Raspivid

Record the default 10 seconds of video to a file. The extension must be h264 since it is h264 encoding.

```
raspivid -fps 25 -o myVideo.h264
```

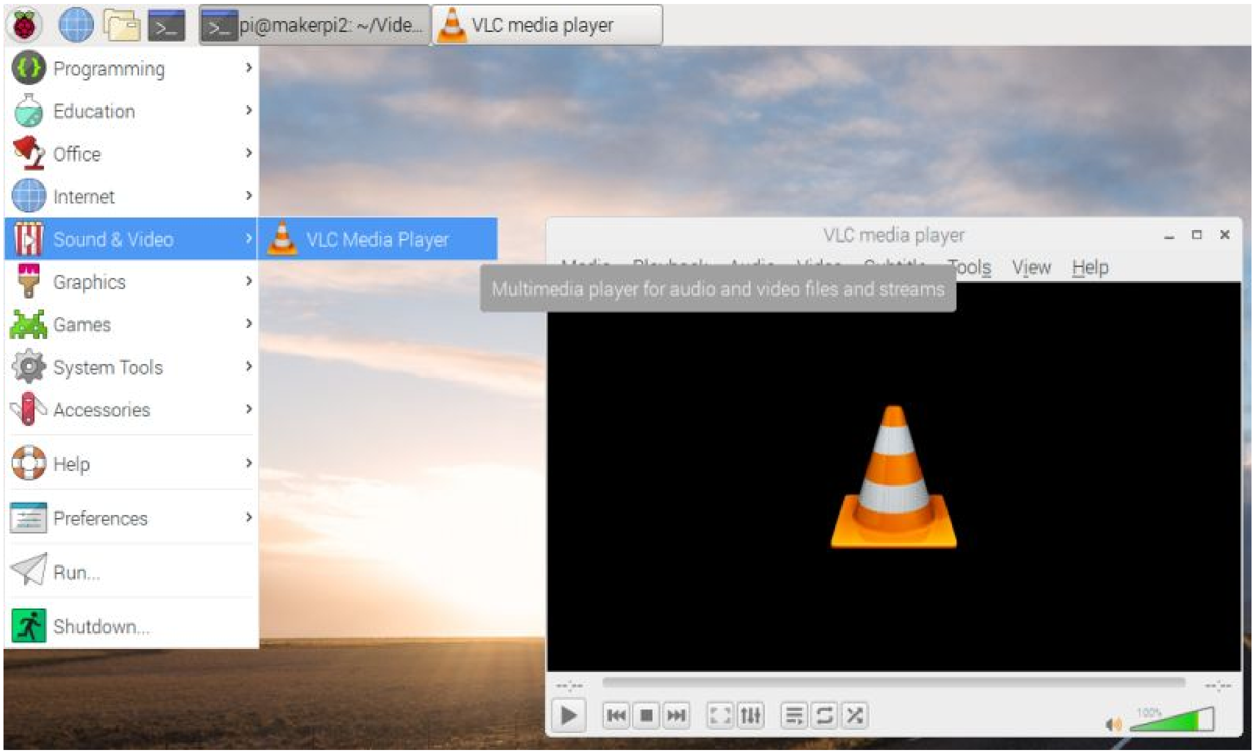
The h264 file is raw video and most video players can’t deal with it properly, and won’t play the right frame rate. The raw video must be put inside a container. The MP4Box software will put the raw h264 inside an mp4 container. Use the -add command to put the video inside an mp4 container.

```
MP4Box -add myVideo.h264 myVideo.mp4
```

The resulting mp4 file can be played from the command line with

```
Omxplayer myVideo.mp4
```

Or can be played with the GUI application VLC.



Exercise 7: Python video

Write a python3 program and run it from the command line.

```
import picamera

camera = picamera.PiCamera()
camera.resolution = (640, 480)
camera.start_recording('my_video.h264')
camera.wait_recording(10)
camera.stop_recording()
```

The resulting video can be played with the omxplayer but a better approach is to use the MP4Box software to put it inside an MP4 container. (See exercise 5)