Raspberry Pi Surveillance Camera
Visitors Monitoring with Raspberry Pi and Pi Camera

Step 1: First run the below command to **update the Ubuntu OS** on Raspberry Pi:

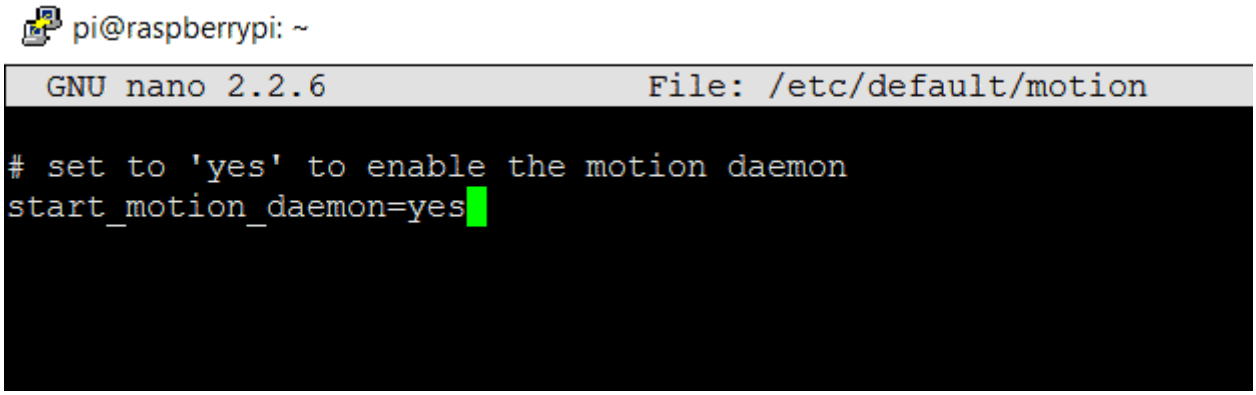
```
sudo apt-get update
```

Step 2: Then **install 'Motion' Library** by using below command:

```
sudo apt-get install motion
```

Step 3: Now **set Motion daemon to yes** by editing the file:/etc/default/motionso that it will be always running. Edit this file using 'gedit' editor with 'sudo' like given below:

- **sudo apt-get install geditsudo**
- **gedit /etc/default/motion**



Step 4: Now we need to **set the permission for the Target Directory** (/var/lib/motion/), in which Motion saves all the Video recordings and picture files. We need to set 'Motion' as owner of this directory by issuing below command:

```
sudo chown motion:motion /var/lib/motion/
```

This permission is necessary otherwise you will get below error, when you check Motion service Status using this command: *sudo service motion status*

```
pi@raspberrypi: /var/lib/motion
pi@raspberrypi:/var/lib/motion $ sudo service motion status
● motion.service - LSB: Start Motion detection
   Loaded: loaded (/etc/init.d/motion)
   Active: active (exited) since Sat 2017-02-04 06:12:49 UTC; 1min 7s ago
  Process: 1301 ExecStop=/etc/init.d/motion stop (code=exited, status=0/SUCCESS)
  Process: 1347 ExecStart=/etc/init.d/motion start (code=exited, status=0/SUCCESS)

Feb 04 06:13:08 raspberrypi motion[1367]: [1] [ERR] [ALL] put_picture: Can't write picture to file /var/lib/motion/0
                                          Thread is going to finish due to this fatal error:
Feb 04 06:13:08 raspberrypi motion[1367]: [1] [NTC] [ALL] motion_loop: Thread exiting
Feb 04 06:13:08 raspberrypi motion[1367]: [1] [NTC] [STR] stream_stop: Closing motion-stream listen socket & active
Feb 04 06:13:08 raspberrypi motion[1367]: [1] [NTC] [STR] stream_stop: Closed motion-stream listen socket & active m
Feb 04 06:13:08 raspberrypi motion[1367]: [1] [NTC] [VID] vid_close: Closing video device /dev/video0
Feb 04 06:13:09 raspberrypi motion[1367]: [0] [NTC] [ALL] main: Threads finished
Feb 04 06:13:09 raspberrypi motion[1367]: [0] [NTC] [STR] httpd_run: motion-httpd - Finishing
Feb 04 06:13:09 raspberrypi motion[1367]: [0] [NTC] [STR] httpd_run: motion-httpd Closing
Feb 04 06:13:09 raspberrypi motion[1367]: [0] [NTC] [STR] motion_web_control: motion-httpd thread exit
Feb 04 06:13:10 raspberrypi motion[1367]: [0] [NTC] [ALL] main: Motion terminating
```

Step 5: Now we are almost done, only we need to change one config option in Motion configuration file *(/etc/motion/motion.conf)* which is *stream_localhost off*. We have to **turn off this local host streaming**, otherwise we will not be able to access the Video feed on our network and it will be only accessible from the Raspberry Pi itself. To doing so, edit the Motion Configuration file with 'gedit' editor and turn it off,
Check the following :

- deamon on
- live_stream quality max 20 to 30
- stream_maxrate 1

Now we are done and ready to get our live feed from the USB web camera connected to Pi. Just start the Motion service using below command and **open your Raspberry Pi's IP, with port 8081**, in your browser (like 192.168.1.103:8081):

```
sudo /etc/init.d/motion start
```

And you will see the live feed from your web camera like below. Here we have used a low cost USB web camera (Quantum web camera QHM500LM), which worked smoothly with our Raspberry Pi, but you can further use a good quality camera for better resolution.

As it will show in browser, you can use any device, to watch the feed, which supports web browser like Mobile, tablet etc. Below is one snapshot from Mobile Phone:

You can always start, stop, restart and get status of Motion service using below four commands:

```
sudo /etc/init.d/motion start

sudo /etc/init.d/motion stop

sudo service motion restart

sudo service motion status
```

or you can reboot the Raspberry Pi as a troubleshooting step when necessary:

```
sudo reboot
```

Refer below Link for more options :-

https://circuitdigest.com/microcontroller-projects/raspberry-pi-surveillance-camera