

RDK-C Normal Thumbnail

- [Introduction](#)
- [Build Procedure](#)
- [Normal Thumbnail Validation Procedure](#)
- [Limitations](#)

Introduction

Normal Thumbnail will get YUV data from RPI-0 camera device and it will convert YUV buffer into JPEG image based on openCV and then stored generated JPEG image in local /tmp directory . This page dedicated to bringing up and validation of Normal Thumbnail functionality in RPI-0/RPI-3 B+

Build Procedure

Please refer below link to build camera image

Morty:

[RDK-C Build Instruction for RPI-0](#)

[RDK-C Build Instruction for RPI-0](#)

Dunfell:

[RDK-C rdk-next Yocto 3.1 dunfell build for Raspberrypi](#)

Normal Thumbnail Validation Procedure

STEP 1:

Add require SSID and PSK in /etc/wpa_supplicant.conf file in below format

```
network={  
ssid="username"  
psk="password"  
}
```

Console output

```
ctrl_interface=/var/run/wpa_supplicant  
ctrl_interface_group=0  
update_config=1  
  
network={  
ssid="RDK"  
psk="Comcast1"  
}
```

STEP 2:

Reboot the Target

After Reboot don't do step 1 and 2.

Note : Step 1 & 2 is only applicable for fresh target boot-up with new image.

STEP 3:

WiFi connection is needed to copy captured JPEG Image from /tmp directory into your PC.

Check WiFi connection by using below command.

ifconfig

Console output

```
root@raspberrypi0-rdk-camera:~# ifconfig
lo          Link encap:Local Loopback
            inet addr:127.0.0.1  Mask:255.0.0.0
            inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING  MTU:65536  Metric:1
            RX packets:87 errors:0 dropped:0 overruns:0 frame:0
            TX packets:87 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:4552 (4.4 KiB)  TX bytes:4552 (4.4 KiB)

wlan0       Link encap:Ethernet  HWaddr B8:27:EB:2E:72:2B
            inet addr:192.168.43.246  Bcast:192.168.43.255  Mask:255.255.255.0
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
            RX packets:23 errors:0 dropped:0 overruns:0 frame:0
            TX packets:44 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:2893 (2.8 KiB)  TX bytes:5887 (5.7 KiB)
```

STEP 4:

check loaded module by using below command

lsmod

Console output

```
root@raspberrypi0-rdk-camera:~# lsmod
Module                Size  Used by
bcm2835_v4l2          40563  0
v4l2_common           4809  1 bcm2835_v4l2
videobuf2_vmalloc     6264  1 bcm2835_v4l2
videobuf2_memops      1528  1 videobuf2_vmalloc
videobuf2_v4l2       12640  1 bcm2835_v4l2
videobuf2_core        27389  2 bcm2835_v4l2,videobuf2_v4l2
videodev             154457  4 v4l2_common,videobuf2_core,bcm2835_v4l2,videobuf2_v4l2
media                 23307  1 videodev
brcmfmac             258239  0
brcmutil              7590  1 brcmfmac
snd_bcm2835           21405  0
cfg80211             492836  1 brcmfmac
snd_pcm               79872  1 snd_bcm2835
rfkill               19936  3 cfg80211
snd_timer             20294  1 snd_pcm
snd                   52949  3 snd_timer,snd_bcm2835,snd_pcm
lirc_rpi              6840  0
lirc_dev              7533  1 lirc_rpi
uio_pdrv_genirq       3469  0
uio                   8703  1 uio_pdrv_genirq
fixed                 2876  0
sch_fq_codel          9662  2
ipv6                  384101  18
```

STEP 5:

check camera device there or not by using below command

ls /dev/video0

Console output

```
root@raspberrypi0-rdk-camera:~# ls /dev/video0
/dev/video0
```

STEP 6:

check rdkcm mediaserver(RMS) binary is running or not.if it is running then we need to stop RMS binary running because we can't able to validate Normal Thumbnail while running RMS application.

Console output : RMS application running status

```
root@raspberrypi0-rdk-camera:~# ps -Af | grep rdkcm mediaserver
root      659      1  22 10:24 ?        00:00:43 ./rdkcm mediaserver ../config/config.lua
root     4911    381   0 10:28 ttyS0    00:00:00 grep rdkcm mediaserver
```

if we get above status then need to do below step.

Console output : Stop RMS application

```
root@raspberrypi0-rdk-camera:~# systemctl stop rms-launcher
```

STEP 7:

Run normal_thumbnail binary by using below command

Console output

```
root@raspberrypi0-rdk-camera:~# normal_thumbnail
```

STEP 8:

Get JPEG image from local /tmp directory.

Console output

```
root@raspberrypi0-rdk-camera:~# cd /tmp/
root@raspberrypi0-rdk-camera:/tmp# ls thumbnail2020-06-28\10\33\57\..jpeg
thumbnail2020-06-28:10:33:57\..jpeg
```

STEP 9:

Copy JPEG image from your RPI /tmp directory into your local PC.

Console output

```
xxxxxx@yyyyy-Lenovo-B480:~/THUMBNAIL$ scp root@RPI_DEVICE_WIFI_IP:/tmp/thumbnail* .

Example:
xxxxxx@yyyyy-Lenovo-B480:~/THUMBNAIL$ scp root@192.168.43.246:/tmp/thumbnail* .
thumbnail2020-06-28:10:33:57\..jpeg
100% 16KB 16.4KB/s 00:00
xxxxxx@yyyyy-Lenovo-B480:~/THUMBNAIL$
```

Limitations

Can't able to validate RMS(RDK Media Streamer) and CVR(Continuous Video Recoding) features while validating Normal Thumbnail feature.

