

**ECEN 5623**  
**REAL-TIME EMBEDDED**  
**SYSTEMS**

**EXERCISE-4**  
**REAL-TIME CONTINUOUS**  
**MEDIA**

**SUBMITTED BY:**  
**SOWMYA**  
**RAMAKRISHNAN**  
**VIRAG GADA**

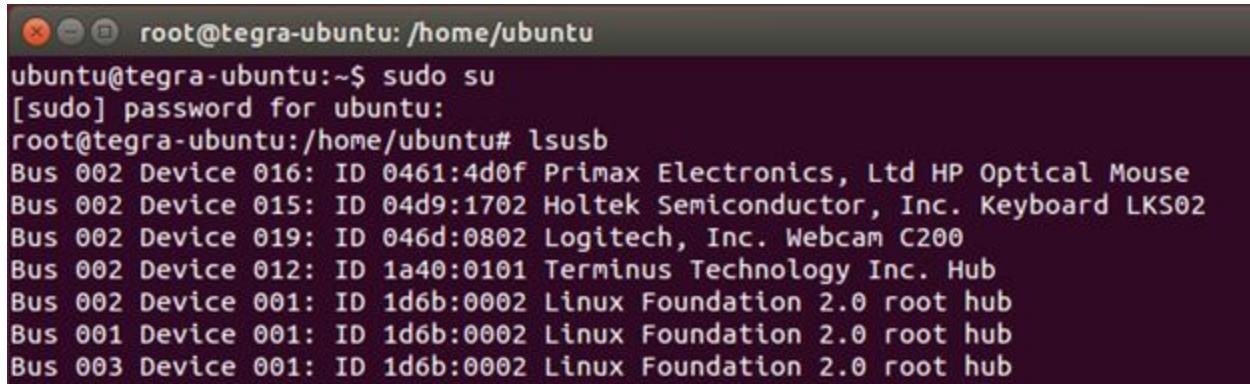
## 1.

Obtain a Logitech C200 camera or equivalent and verify that it is detected by the DE1-SoC board or Jetson Board USB driver. You can check the camera out from the TA's or purchase one of your own, or use another camera that has a compliant UVC driver. Use **lsusb**, **lsmod** and **dmesg** kernel driver configuration tool to make sure your Logitech C200 USB camera is plugged in and recognized by your DE1-SoC or Jetson (note that on the Jetson, it does not use driver modules, but rather a monolithic kernel image, so lsmod will not look the same as other Linux systems – see what you can find by exploring /cat/proc on your Jetson to find the camera USB device). For the Jetson, do **lsusb | grep C200** and prove to the TA (and more importantly yourself) with that output (screenshot) that your camera is recognized. For systems other than a Jetson, do **lsmod | grep video** and verify that the UVC driver is loaded as well (<http://www.ideasonboard.org/uvc/>). To further verify, or debug if you don't see the UVC driver loaded in response to plugging in the USB camera, do **dmesg | grep video** or just **dmesg** and scroll through the log messages to see if your USB device was found. Capture all output and annotate what you see with descriptions to the best of your understanding.

## ANSWER

A Logitech C200 camera was obtained and verified that it is detected by the Jetson Board USB Driver. The lsusb, lsmod and dmesg kernel driver configuration tools were used to make sure the Logitech C200 USB camera is plugged in and recognized by Jetson.

The camera is connected to the USB driver of the Jetson and the lsusb command was first executed. This command displays all USB buses in the system, and the devices connected to each one of them. The output shows the devices, and C200 is one of them.



```
root@tegra-ubuntu: /home/ubuntu
ubuntu@tegra-ubuntu:~$ sudo su
[sudo] password for ubuntu:
root@tegra-ubuntu:/home/ubuntu# lsusb
Bus 002 Device 016: ID 0461:4d0f Primax Electronics, Ltd HP Optical Mouse
Bus 002 Device 015: ID 04d9:1702 Holtek Semiconductor, Inc. Keyboard LKS02
Bus 002 Device 019: ID 046d:0802 Logitech, Inc. Webcam C200
Bus 002 Device 012: ID 1a40:0101 Terminus Technology Inc. Hub
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

**Execution of lsusb command**

The lsmod command is then executed to ensure that the UVC driver is loaded. The UVC driver is a USB Video Class driver which provides driver support for USB Video Class devices. It enables the video streaming functionality on the USB bus. The lsmod command shows the loadable kernel modules that are currently loaded in the system. "UVC Video" is a loaded module and this proves that the camera has been detected successfully.

## USB Hot-Plug Proof

The **lsusb | grep C200** command was executed and the screenshot below proves that the camera is recognized. It is identified as Device 19 on Bus 2.

## UVC Driver Verification

The **dmesg | grep video** command was executed and it was found that the UVC Driver was loaded in response to plugging in the USB camera. The screenshot for both the above commands is attached below.

A terminal window titled "root@tegra-ubuntu: /home/ubuntu". The window contains the following text:

```
root@tegra-ubuntu:/home/ubuntu# lsusb | grep C200
Bus 002 Device 019: ID 046d:0802 Logitech, Inc. Webcam C200
root@tegra-ubuntu:/home/ubuntu# dmesg | grep video
[    4.226909] usbcore: registered new interface driver uvcvideo
[    5.140217] uvcvideo: Found UVC 1.00 device <unnamed> (046d:0802)
[   42.365615] uvcvideo: Found UVC 1.00 device <unnamed> (046d:0802)
[  51.407109] uvcvideo: Found UVC 1.00 device <unnamed> (046d:0802)
[  65.137875] uvcvideo: Found UVC 1.00 device <unnamed> (046d:0802)
root@tegra-ubuntu:/home/ubuntu#
```

### USB Hot-Plug proof and UVC Driver Verification

The camera is thus connected, detected successfully and ready for use.

## 2.

### Option 1: Camorama

If you do not have *camorama*, do **apt-get install camorama** on your DE1-SoC board [you may need to first do **sudo add-apt-repository universe; sudo apt-get update**]. This should not only install nice camera capture GUI tools, but also the V4L2 API (described well in this series of Linux articles - <http://lwn.net/Articles/203924/> ). Running camorama should provide an interactive camera control session for your Logitech C2xx camera – if you have issues connecting to your camera do a “man camorama” and specify your camera device file entry point (e.g. /dev/video0). Run camorama and play with Hue, Color, Brightness, White Balance and Contrast, take an example image and take a screen shot of the tool and provide both in your report.

### Option 2: Cheese

If you do not have *cheese*, do **sudo apt-get install cheese** on your Jetson or DE1-SoC board or other native Linux system. This should not only install nice camera capture GUI tools, but also the V4L2 API (described well in this series of Linux articles - <http://lwn.net/Articles/203924/> ). Running cheese should provide an interactive camera control session for your Logitech C2xx camera – if you have issues connecting to your camera do a “man cheese” and specify your camera device file entry point (e.g. /dev/video0). Show that you tested your camera with a cheese screen dump and test photo.

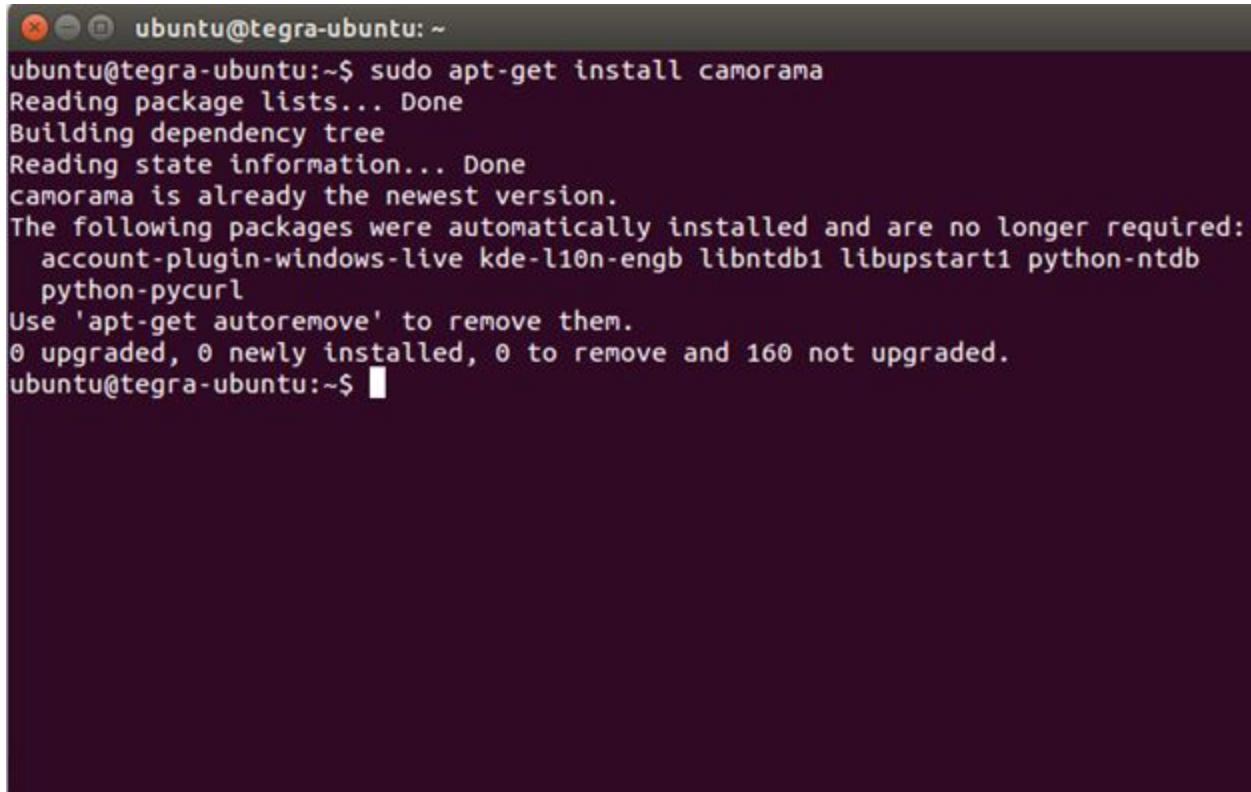
## ANSWER

### Use of the ‘Camorama’ Tool

The **sudo add-apt-repository universe**, **sudo apt-get update** and **sudo apt-get install camorama** commands were executed to install Camorama on the device. In this case, the installation was confirmed/verified.

```
ubuntu@tegra-ubuntu: ~
ubuntu@tegra-ubuntu:~$ sudo add-apt-repository universe
[sudo] password for ubuntu:
'universe' distribution component is already enabled for all sources.
ubuntu@tegra-ubuntu:~$ sudo apt-get update
Ign file: InRelease
Ign file: InRelease
Get:1 file: Release.gpg [181 B]
Get:2 file: Release.gpg [181 B]
Get:3 file: Release [574 B]
Get:4 file: Release [152 B]
Ign file: Translation-en_US
Ign file: Translation-en
Ign file: Translation-en_US
Ign file: Translation-en
Ign http://ports.ubuntu.com trusty InRelease
Hit http://ports.ubuntu.com trusty-updates InRelease
Hit http://ports.ubuntu.com trusty-security InRelease
Hit http://ports.ubuntu.com trusty Release.gpg
Hit http://ports.ubuntu.com trusty-updates/main Sources
Hit http://ports.ubuntu.com trusty-updates/restricted Sources
Hit http://ports.ubuntu.com trusty-updates/universe Sources
Hit http://ports.ubuntu.com trusty-updates/multiverse Sources
Hit http://ports.ubuntu.com trusty-updates/main armhf Packages
Hit http://ports.ubuntu.com trusty-updates/restricted armhf Packages
Hit http://ports.ubuntu.com trusty-updates/universe armhf Packages
Hit http://ports.ubuntu.com trusty-updates/multiverse armhf Packages
Hit http://ports.ubuntu.com trusty-updates/main Translation-en
Hit http://ports.ubuntu.com trusty-updates/multiverse Translation-en
Hit http://ports.ubuntu.com trusty-updates/restricted Translation-en
Hit http://ports.ubuntu.com trusty-updates/universe Translation-en
Hit http://ports.ubuntu.com trusty-security/main Sources
Hit http://ports.ubuntu.com trusty-security/restricted Sources
Hit http://ports.ubuntu.com trusty-security/universe Sources
Hit http://ports.ubuntu.com trusty-security/multiverse Sources
Hit http://ports.ubuntu.com trusty-security/main armhf Packages
Hit http://ports.ubuntu.com trusty-security/restricted armhf Packages
Hit http://ports.ubuntu.com trusty-security/universe armhf Packages
Hit http://ports.ubuntu.com trusty-security/multiverse armhf Packages
Hit http://ports.ubuntu.com trusty-security/main Translation-en
Hit http://ports.ubuntu.com trusty-security/multiverse Translation-en
Hit http://ports.ubuntu.com trusty-security/restricted Translation-en
Hit http://ports.ubuntu.com trusty-security/universe Translation-en
Hit http://ports.ubuntu.com trusty Release
Hit http://ports.ubuntu.com trusty/main Sources
Hit http://ports.ubuntu.com trusty/restricted Sources
Hit http://ports.ubuntu.com trusty/universe Sources
Hit http://ports.ubuntu.com trusty/multiverse Sources
Hit http://ports.ubuntu.com trusty/main armhf Packages
Hit http://ports.ubuntu.com trusty/restricted armhf Packages
Hit http://ports.ubuntu.com trusty/universe armhf Packages
Hit http://ports.ubuntu.com trusty/multiverse armhf Packages
Hit http://ports.ubuntu.com trusty/main Translation-en
Hit http://ports.ubuntu.com trusty/multiverse Translation-en
Hit http://ports.ubuntu.com trusty/restricted Translation-en
Hit http://ports.ubuntu.com trusty/universe Translation-en
Ign http://ports.ubuntu.com trusty/main Translation-en_US
Ign http://ports.ubuntu.com trusty/multiverse Translation-en_US
Ign http://ports.ubuntu.com trusty/restricted Translation-en_US
Ign http://ports.ubuntu.com trusty/universe Translation-en_US
Reading package lists... Done
ubuntu@tegra-ubuntu:~$
```

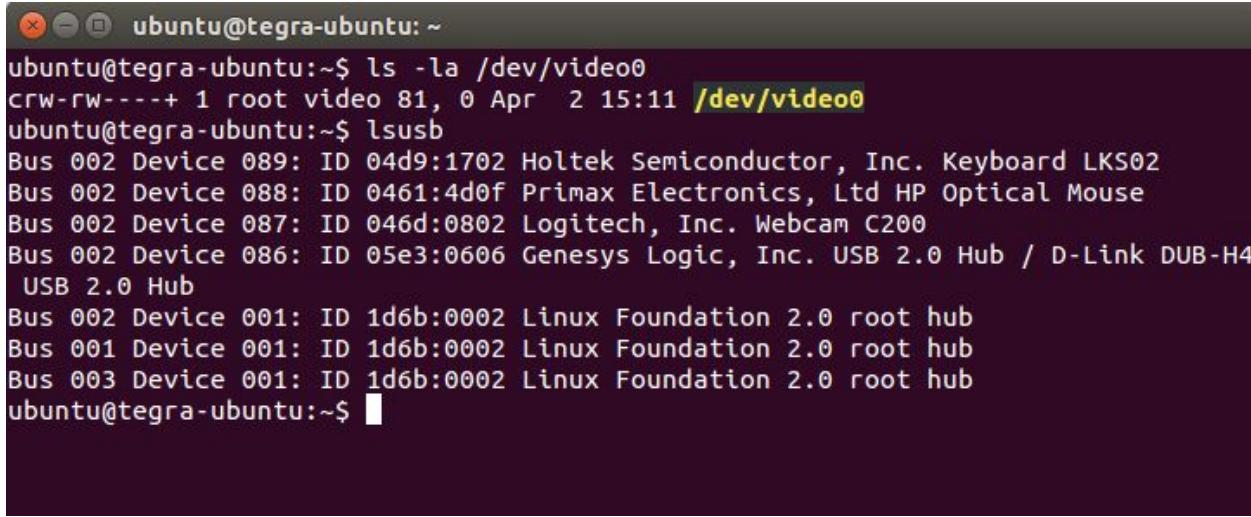
Successful installation of updates

A screenshot of a terminal window titled "ubuntu@tegra-ubuntu: ~". The window contains the following text:

```
ubuntu@tegra-ubuntu:~$ sudo apt-get install camorama
Reading package lists... Done
Building dependency tree
Reading state information... Done
camorama is already the newest version.
The following packages were automatically installed and are no longer required:
  account-plugin-windows-live kde-l10n-engb libntdb1 libupstart1 python-ntdb
  python-pycurl
Use 'apt-get autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 160 not upgraded.
ubuntu@tegra-ubuntu:~$
```

### Camorama Installation

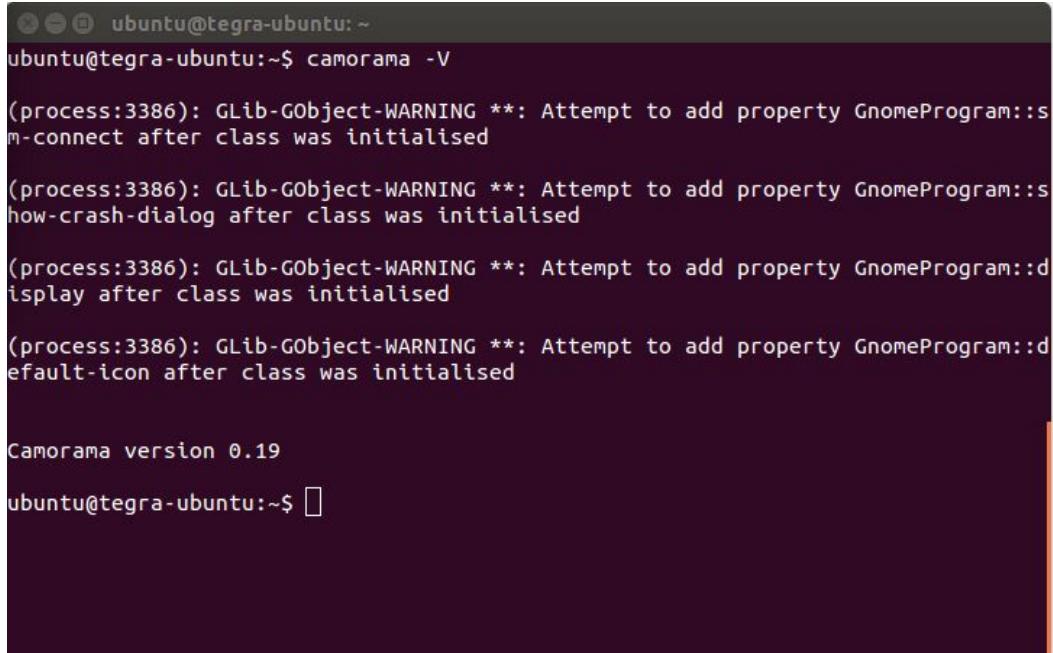
The camera was verified using the **lsusb** command and the **ls -la /dev/video0** command was used to verify the camera device entry point.

A screenshot of a terminal window titled "ubuntu@tegra-ubuntu: ~". The window contains the following text:

```
ubuntu@tegra-ubuntu:~$ ls -la /dev/video0
crw-rw----+ 1 root video 81, 0 Apr  2 15:11 /dev/video0
ubuntu@tegra-ubuntu:~$ lsusb
Bus 002 Device 089: ID 04d9:1702 Holtek Semiconductor, Inc. Keyboard LKS02
Bus 002 Device 088: ID 0461:4d0f Primax Electronics, Ltd HP Optical Mouse
Bus 002 Device 087: ID 046d:0802 Logitech, Inc. Webcam C200
Bus 002 Device 086: ID 05e3:0606 Genesys Logic, Inc. USB 2.0 Hub / D-Link DUB-H4
  USB 2.0 Hub
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
ubuntu@tegra-ubuntu:~$
```

### Verification of Camera and Camorama

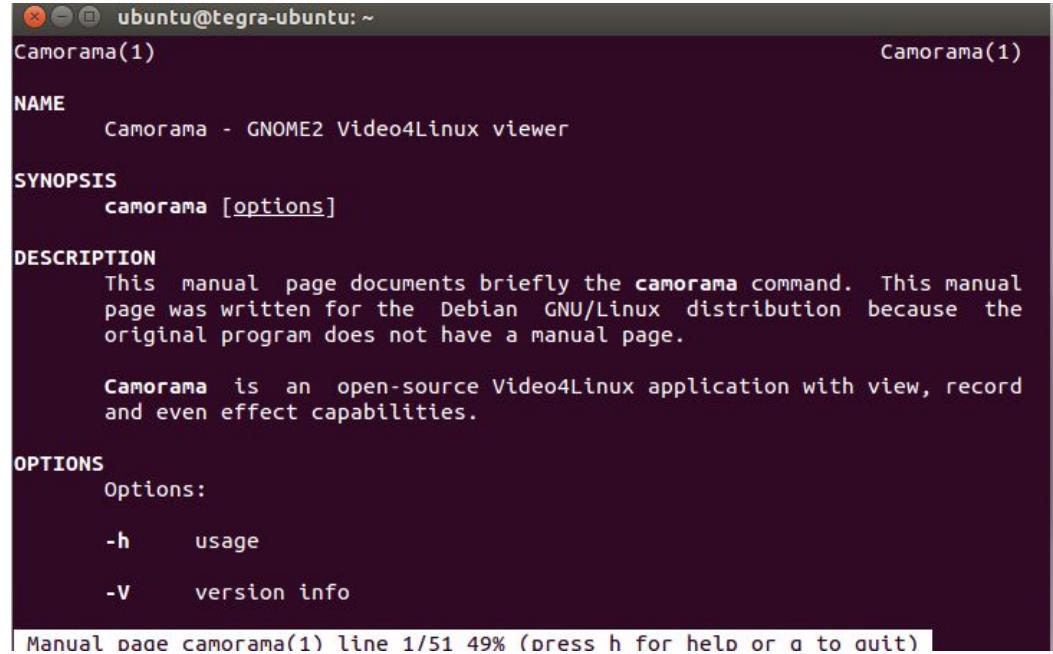
The commands **camorama -V** and **man camorama** produced the Camorama version (0.19) and the man page which showed the description of the **camorama** command and the various options that can be used with it.



```
ubuntu@tegra-ubuntu:~$ camorama -V
(process:3386): GLib-GObject-WARNING **: Attempt to add property GnomeProgram::m-connect after class was initialised
(process:3386): GLib-GObject-WARNING **: Attempt to add property GnomeProgram::show-crash-dialog after class was initialised
(process:3386): GLib-GObject-WARNING **: Attempt to add property GnomeProgram::display after class was initialised
(process:3386): GLib-GObject-WARNING **: Attempt to add property GnomeProgram::default-icon after class was initialised

Camorama version 0.19
ubuntu@tegra-ubuntu:~$ 
```

### Camorama Version



```
ubuntu@tegra-ubuntu:~$ man camorama
Camorama(1)                                         Camorama(1)

NAME
    Camorama - GNOME2 Video4Linux viewer

SYNOPSIS
    camorama [options]

DESCRIPTION
    This manual page documents briefly the camorama command. This manual
    page was written for the Debian GNU/Linux distribution because the
    original program does not have a manual page.

    Camorama is an open-source Video4Linux application with view, record
    and even effect capabilities.

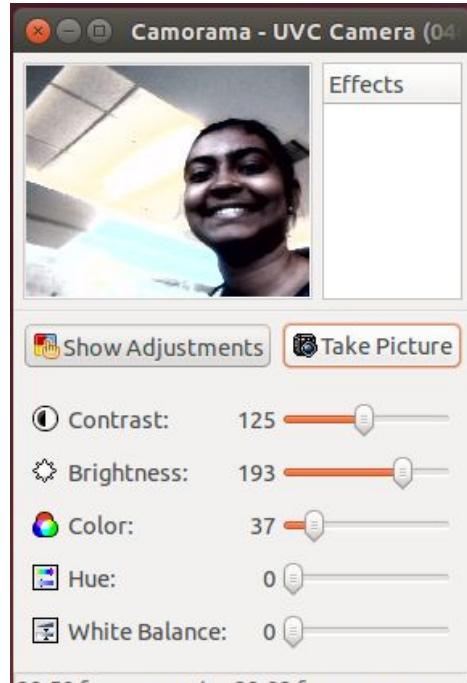
OPTIONS
    Options:
        -h      usage
        -V      version info

Manual page camorama(1) line 1/51 49% (press h for help or q to quit)
```

### Camorama Man Page

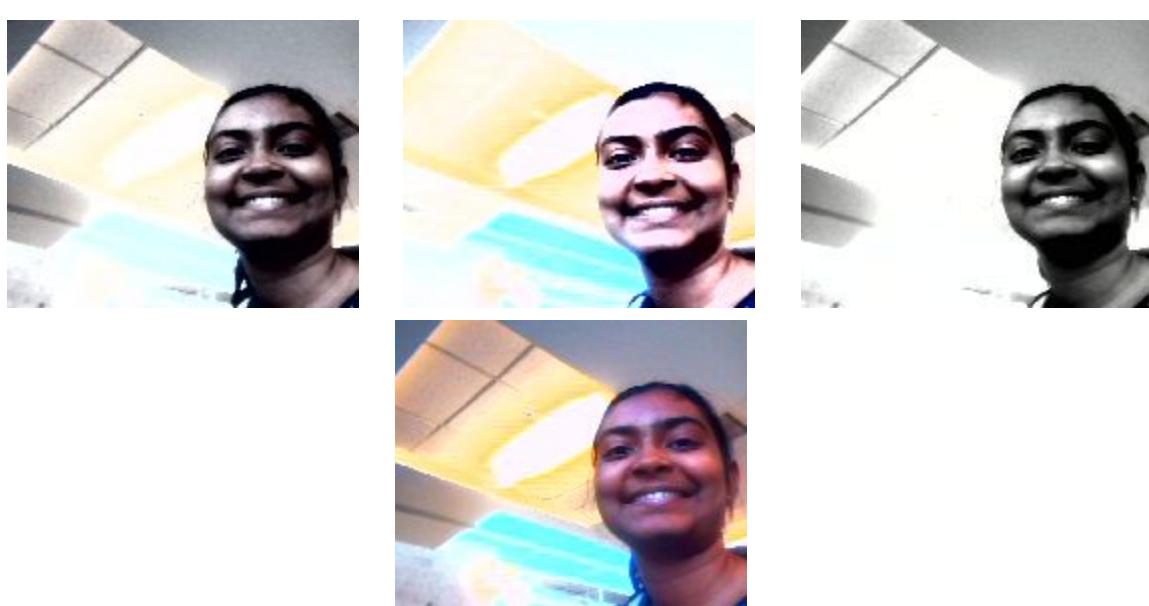
Camorama is an application that can be used to view, make remote captures and save images from a webcam device. It provides nice camera capture GUI tools, and allows us to apply a number of image filters to adjust parameters such as contrast, hue, white balance, brightness and color.

The screenshot below shows the Camorama application running.



**Camorama Execution**

Settings were adjusted and various images were captured and saved on the device, as shown below.



### 3.

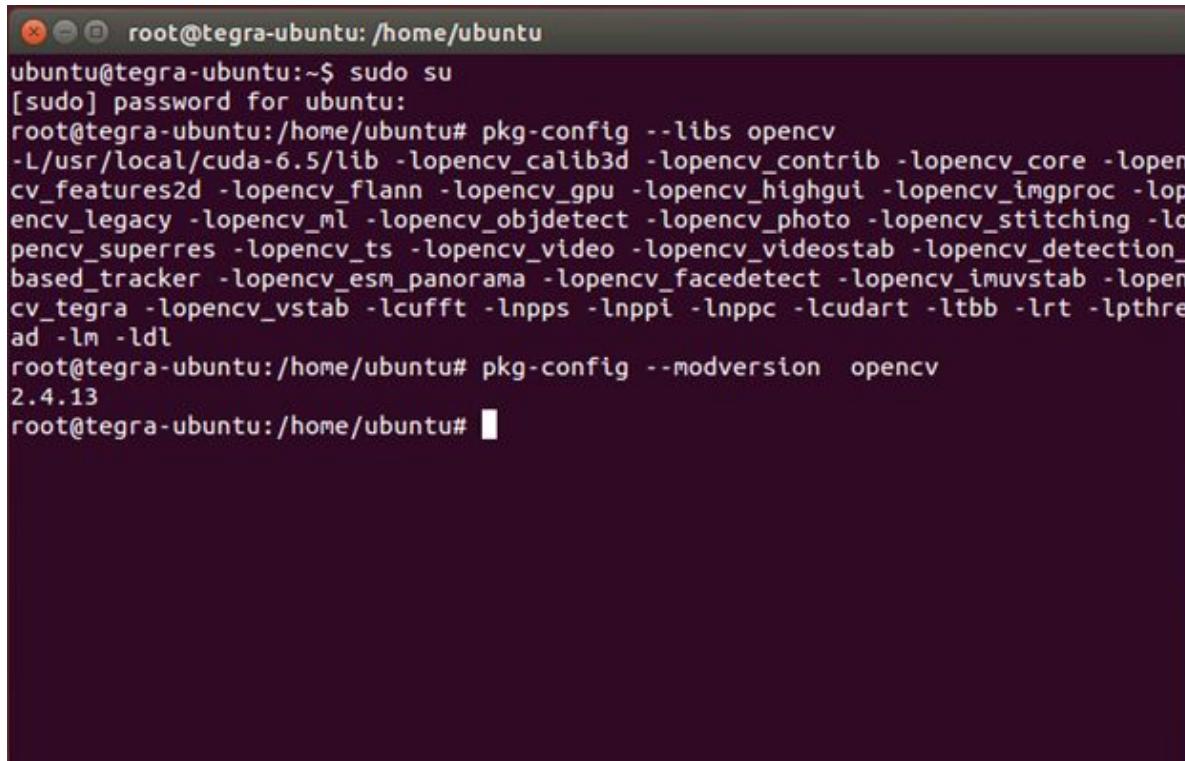
Using your verified Logitech C200 camera on a DE1-SoC or Jetson, verify that it can stream continuously using to a raw image buffer for transformation and processing using example code from the [computer-vision](#) or computer\_vision\_cv3\_tested folder such as [simple-capture](#), [simpler-capture](#), or [simpler-capture-2](#). Read the code and modify the device that is opened if necessary to get this to work. Provide a screen shot to prove that you got continuous capture to work. Note that simpler capture requires installation of OpenCV on your DE1-SoC, Jetson, or native Linux system.

For the Jetson this will likely already be available on your board, but if not, please follow [simple instructions found here to install openCV](#) [the “Option 2, Building the public OpenCV library from source” is the recommended approach with –DWITH\_CUDA=OFF. Don’t install CUDA and please leave it off when you build OpenCV.

For the DE1-SoC please find the files soc\_system.rbf and SettingUp.pdf on D2L. They contain instructions for setting up board and installing OpenCV. The TAs have set up using these and are able to complete exercise 4 requirements. The cmake command for opencv installation has been changed so that it works on the board too. Alternatively, you can use OpenCV port found here: <http://rocketboards.org/foswiki/view/Projects/OpenCVPort>. If you have trouble getting OpenCV to work on your board, try running it on your laptop under Linux first. You can use [OpenCV install](#), [OpenCV with Python bindings](#) for 2.x and 3.x for this.

## ANSWER

Nvidia Jetson TK1 has OpenCV pre-installed as an in-built package. The **pkg-config --libs opencv** command was used to verify that the OpenCV libraries are loaded. The version of OpenCV installed was verified using the command **pkg-config --modversion opencv**.

A screenshot of a terminal window titled "root@tegra-ubuntu: /home/ubuntu". The terminal shows the following command execution:

```
root@tegra-ubuntu:~$ sudo su
[sudo] password for ubuntu:
root@tegra-ubuntu:/home/ubuntu# pkg-config --libs opencv
-L/usr/local/cuda-6.5/lib -lopencv_calib3d -lopencv_contrib -lopencv_core -lopencv_features2d -lopencv_flann -lopencv_gpu -lopencv_highgui -lopencv_imgproc -lopencv_legacy -lopencv_ml -lopencv_objdetect -lopencv_photo -lopencv_stitching -lopencv_superres -lopencv_ts -lopencv_video -lopencv_videostab -lopencv_detection_based_tracker -lopencv_esm_panorama -lopencv_facedetect -lopencv_imuvstab -lopencv_tegra -lopencv_vstab -lcufft -lnpps -lnppi -lnppc -lcudart -ltbb -lrt -lpthread -lm -ldl
root@tegra-ubuntu:/home/ubuntu# pkg-config --modversion opencv
2.4.13
root@tegra-ubuntu:/home/ubuntu#
```

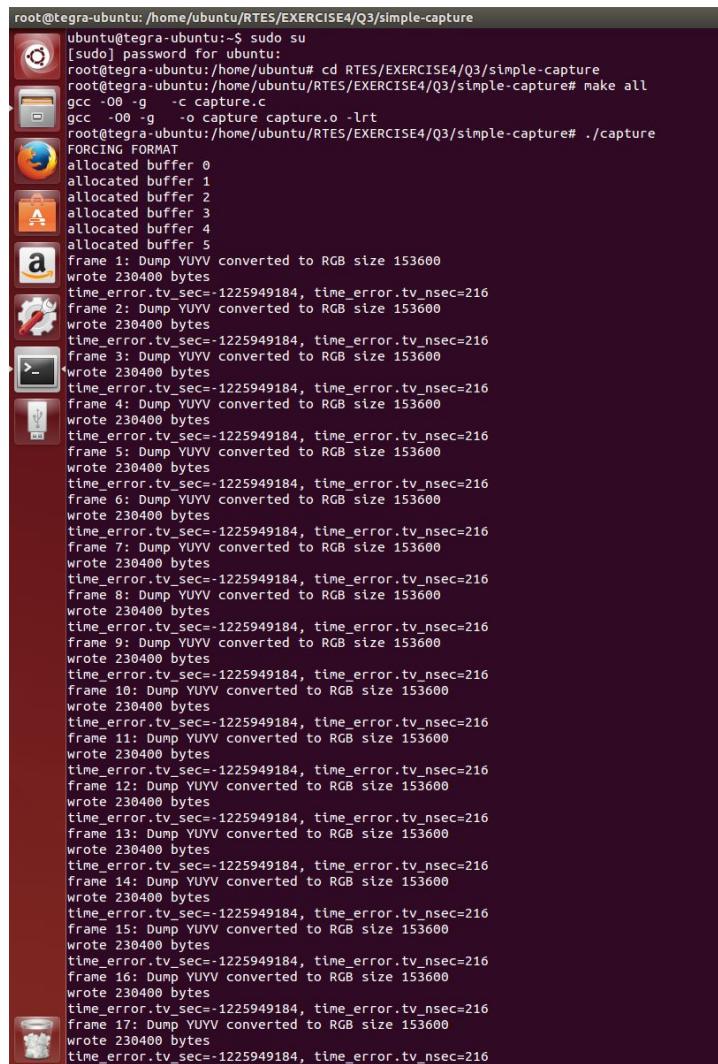
The terminal window has a dark background and light-colored text. The title bar is visible at the top.

Verification of OpenCV Installation

Using the verified Logitech C200 on the Jetson TK1, Example code from Dr.Siewart's Computer-Vision folder was used to verify continuous streaming using raw image buffer for transformation and processing. The codes were read, minor modifications were done after which they were built and executed to obtain outputs.

### simple-capture

This program uses the UVC drivers to acquire digital video from a source. Each acquired frame is time-stamped and saved as a ppm file. The code thus captures 30 frames, timestamps and saves them as ppm files in the same directory. The screenshots showing the execution of the code is as follows.

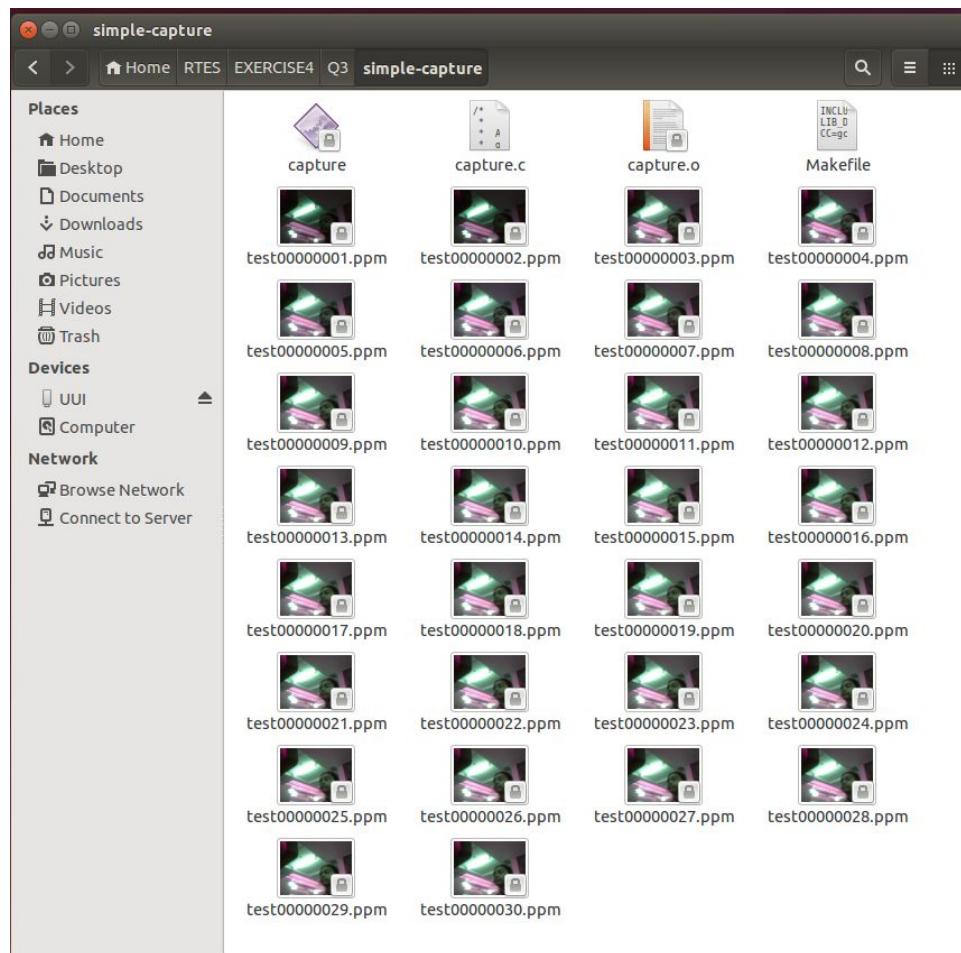


```
root@tegra-ubuntu: /home/ubuntu/RTES/EXERCISE4/Q3/simple-capture
ubuntu@tegra-ubuntu:~$ sudo su
[sudo] password for ubuntu:
root@tegra-ubuntu:/home/ubuntu# cd RTEs/EXERCISE4/Q3/simple-capture
root@tegra-ubuntu:/home/ubuntu/RTEs/EXERCISE4/Q3/simple-capture# make all
gcc -O0 -g -o capture capture.c -lrt
root@tegra-ubuntu:/home/ubuntu/RTEs/EXERCISE4/Q3/simple-capture# ./capture
FORCING FORMAT
allocated buffer 0
allocated buffer 1
allocated buffer 2
allocated buffer 3
allocated buffer 4
allocated buffer 5
frame 1: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 2: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 3: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 4: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 5: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 6: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 7: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 8: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 9: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 10: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 11: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 12: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 13: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 14: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 15: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 16: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 17: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
```

simple-capture Execution - 1

```
root@tegra-ubuntu: /home/ubuntu/RTES/EXERCISE4/Q3/simple-capture
frame 10: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 11: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 12: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 13: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 14: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 15: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 16: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 17: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 18: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 19: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 20: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 21: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 22: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 23: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 24: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 25: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 26: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 27: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 28: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 29: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
frame 30: Dump YUVV converted to RGB size 153600
wrote 230400 bytes
time_error.tv_sec=-1225949184, time_error.tv_nsec=216
root@tegra-ubuntu:/home/ubuntu/RTES/EXERCISE4/Q3/simple-capture#
```

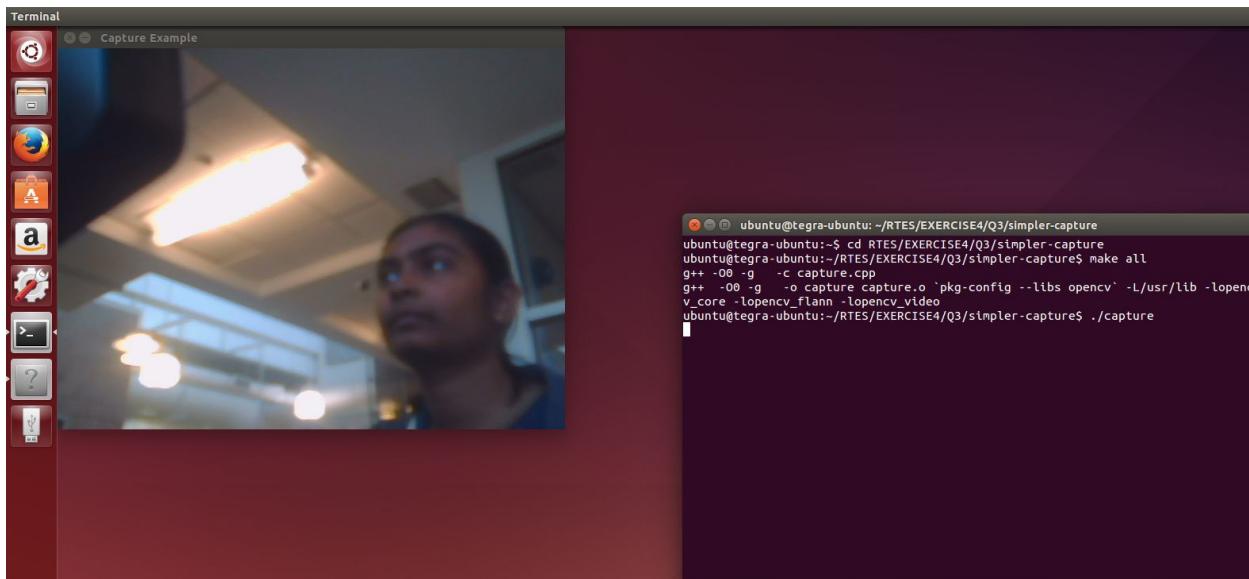
simple-capture Execution - 2



The ppm files saved in directory

## **simpler-capture**

In this program, the OpenCV API is used to stream and capture continuous media from C200. A window named “Capture Example” is created when the code is built and executed. The frame image is displayed on the window and this is done using the cvShowImage() function. Thus, continuous video streaming is obtained. The code terminates when the ESC key is pressed. The screenshot showing the execution of the code is as follows.

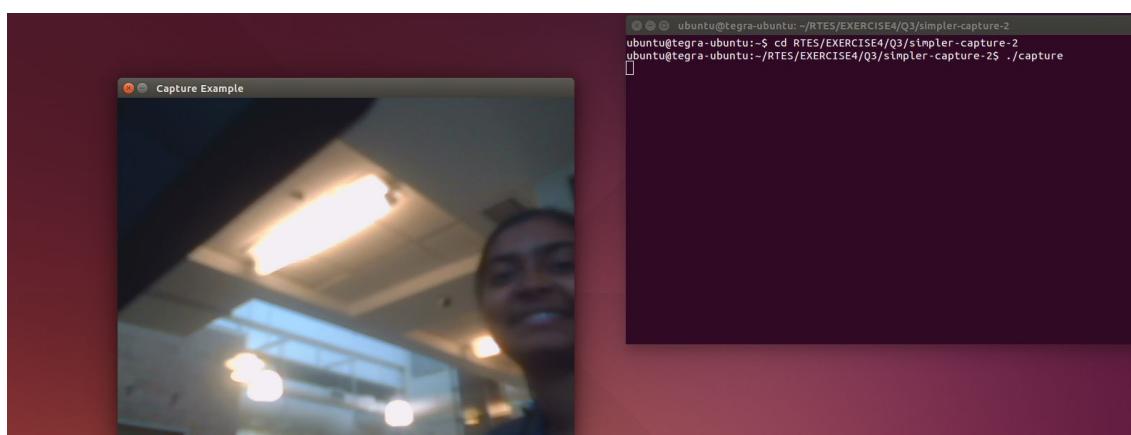


**simpler-capture Execution**

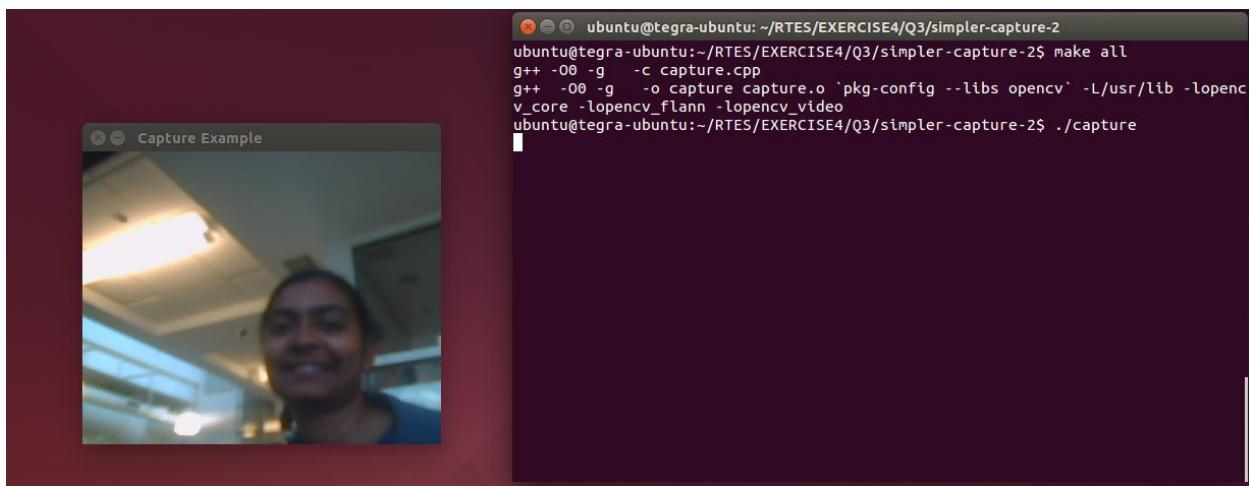
## **simpler-capture-2**

This program is similar to Simpler-capture in its use of the openCV API for continuous image capture, but additionally uses the cvSetCaptureProperty() that allows the user to set the resolution of the image to be captured. The desired height and width of frames to be captured are entered as arguments to the function. The resolution was changed a couple of times to obtain various outputs.

The screenshots showing the execution of the code are as follows.



**simpler-capture-2 Execution - Default resolution**



simpler-capture-2 Execution - Changed Resolution - 1

```
capture.cpp (~/RTES/EXERCISE4/Q3/simpler-capture-2) - gedit
capture.cpp x

/*
 * Example by Sam Siewert
 */
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
#include <iostream>

#include <opencv2/core/core.hpp>
#include <opencv2/highgui/highgui.hpp>

using namespace cv;
using namespace std;

#define LRES 400
#define VRES 480

int main( int argc, char** argv )
{
    cvNamedWindow("Capture Example", CV_WINDOW_AUTOSIZE);
    CvCapture* capture = cvCreateCameraCapture(1);
    IplImage* frame;

    cvSetCaptureProperty(capture, CV_CAP_PROP_FRAME_WIDTH, LRES);
    cvSetCaptureProperty(capture, CV_CAP_PROP_FRAME_HEIGHT, VRES);

    while(1)
    {
        frame=cvQueryFrame(capture);

        if(!frame) break;

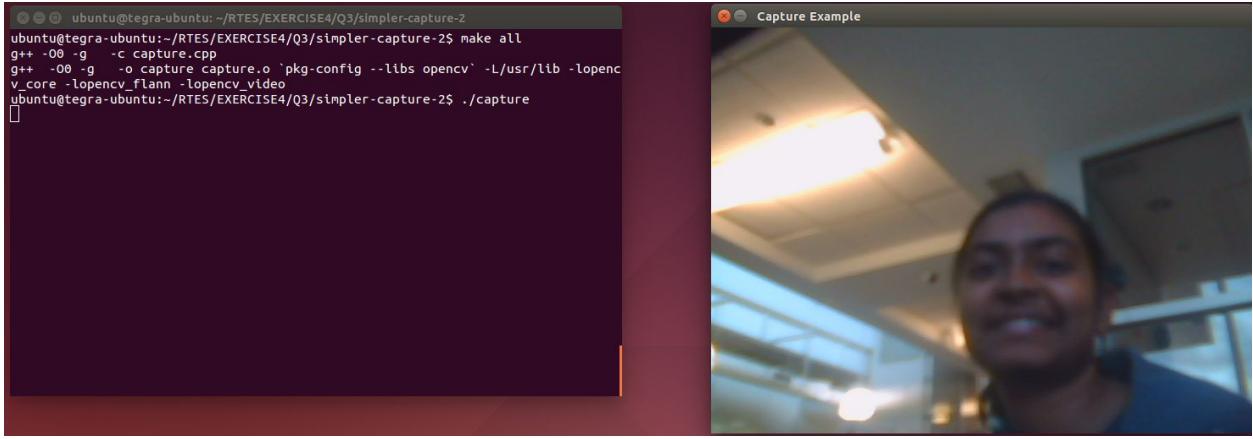
        cvShowImage("Capture Example", frame);

        char c = cvWaitKey(33);
        if( c == 27 ) break;
    }

    cvReleaseCapture(&capture);
    cvDestroyWindow("Capture Example");
}
```

C++ ▾ Tab Width: 8 ▾ Ln 17, Col 17 INS

simpler-capture-2 Changes in Resolution - 1



**simpler-capture-2 Execution - Changed Resolution - 2**

```

/*
 * Example by Sam Siewert
 */
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
#include <iostream>

#include <opencv2/core/core.hpp>
#include <opencv2/highgui/highgui.hpp>

using namespace cv;
using namespace std;

#define HRES 1080
#define VRES 640

int main( int argc, char** argv )
{
    cvNamedWindow("Capture Example", CV_WINDOW_AUTOSIZE);
    CvCapture* capture = cvCreateCameraCapture(1);
    IplImage* frame;

    cvSetCaptureProperty(capture, CV_CAP_PROP_FRAME_WIDTH, HRES);
    cvSetCaptureProperty(capture, CV_CAP_PROP_FRAME_HEIGHT, VRES);

    while(1)
    {
        frame=cvQueryFrame(capture);

        if(!frame) break;

        cvShowImage("Capture Example", frame);

        char c = cvWaitKey(33);
        if( c == 27 ) break;
    }

    cvReleaseCapture(&capture);
    cvDestroyWindow("Capture Example");
};


```

**simpler-capture-2 Changes in Resolution - 2**

#### 4.

Choose a continuous transformation OpenCV example from [computer-vision](#) such as the [canny-interactive](#), [hough-interactive](#), [hough-elliptical-interactive](#), or [stereo-transform-improved](#) or the from the same 4 transforms in [computer vision cv3 tested](#). Show a screen shot to prove you built and ran the code. Provide a detailed explanation of the code and research uses for the continuous transformation by looking up API functions in the OpenCV manual (<http://docs.opencv.org>) and for stereo-transform-improved either implement or explain how you could make this work continuously rather than snapshot only.

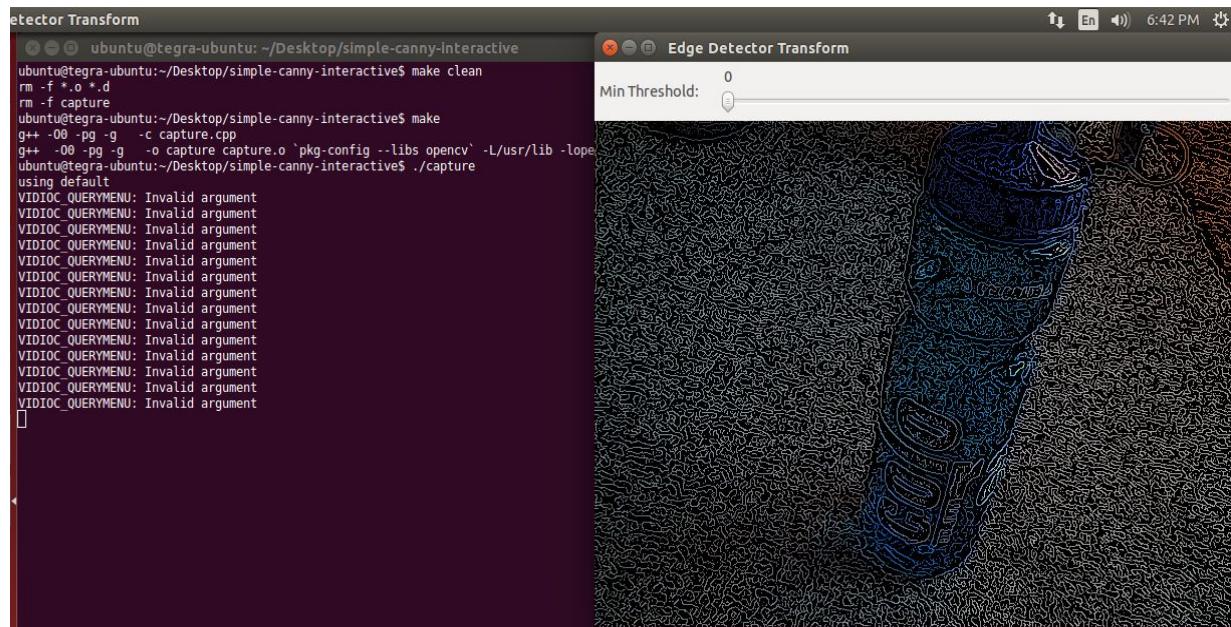
## ANSWER

### Canny Interactive:

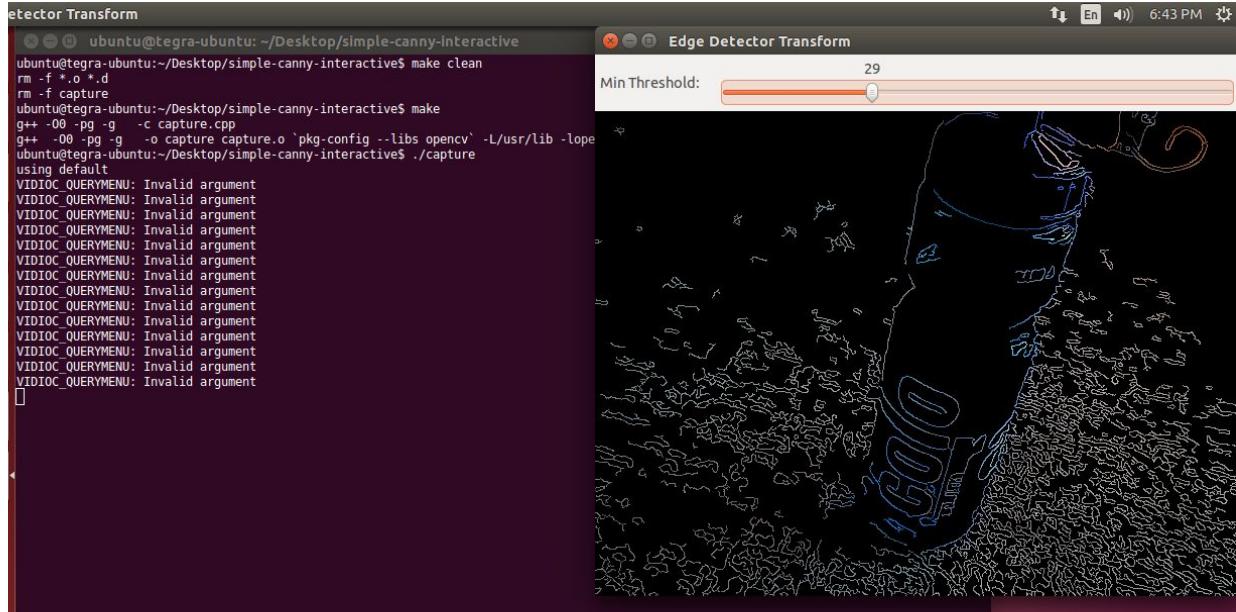
The Canny transform is an edge detection algorithm which uses multi-stage algorithm to detect edges in an image.

### Description:

- Using cvCreateCameraCapture we first select the camera device we want to open
- We create a named window and trackbar using namedWindow and createTrackbar functions. The trackbar can be used to set the threshold for the edge detector algorithm
- Then to capture a single frame from the continuous media stream we use cvQueryFrame
- We convert the frame into a matrix form using the Mat class
- The RGB image is converted to Gray scale using the cvtColor function with CV\_RGB2GRAY flag
- The image is blurred with a filter of kernel size 3
- We use the predefined Canny function to apply the transform detected edges, threshold and the Sobel kernel size to be used
- The destination image is filled with 0's using Scalar::all(0) to make it completely black
- Using .copyTo function we map only the areas of the image that are identified as edges (on a black background)
- Finally the result is displayed using imshow



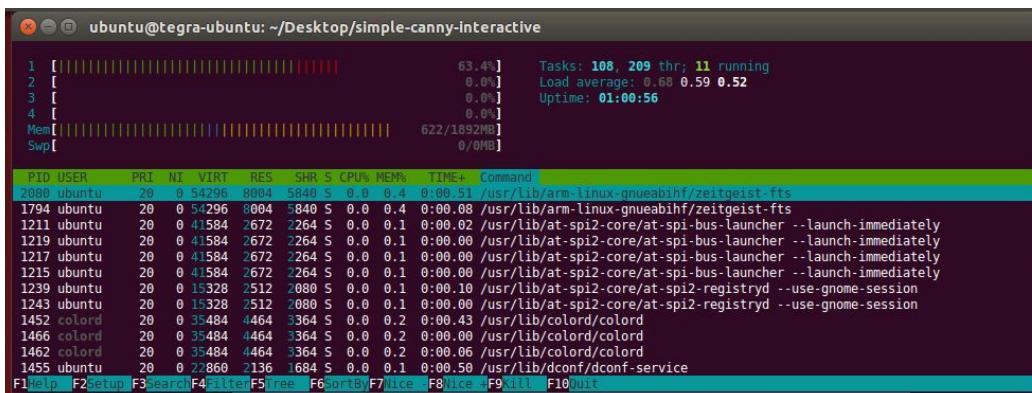
## simple-canny-interactive output - 1



## simple-canny-interactive output - 2

### CPU Loading:

Using htop we find the loading on each CPU core. In this example we can see that only 1 CPU core is being used and it is loaded upto 63.4%



## Simple-canny-interactive CPU loading

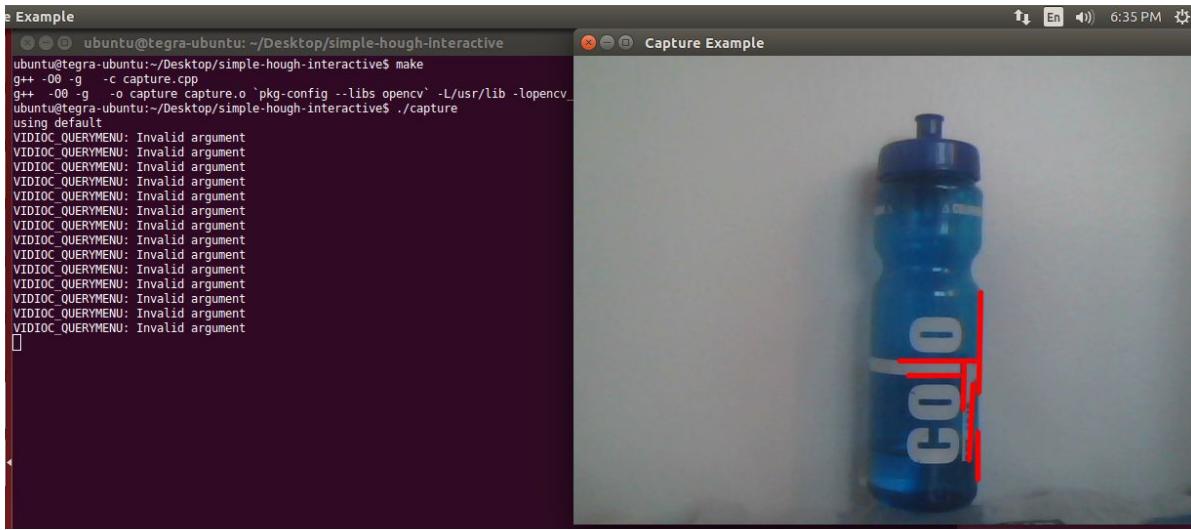
### Hough Interactive:

Hough transform is a feature extraction technique used in image analysis. The Hough Line transform is used to detect straight lines. First an edge detection algorithm is used for pre-processing.

### Description:

- Using cvCreateCameraCapture we first select the camera device we want to open
- Then to capture a single frame from the continuous media stream we use cvQueryFrame

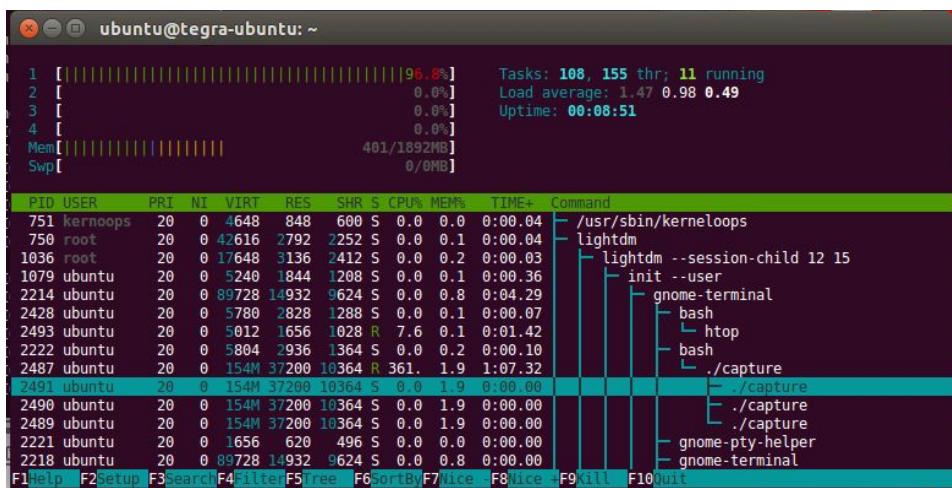
- We convert the frame into a matrix form using the Mat class
  - The edges of the image are detected using the Canny transform
  - The Gray image is converted to BGR using the cvtColor function with CV\_GRAY2BGR flag
  - We then apply the Hough transform using the HoughLinesP function which takes destination, a vector of lines to store parameters of detected lines, resolution, threshold, minimum number of points to form a line and maximum gap between 2 points in order to consider them to lie on a single straight line.
  - Then we display the result by drawing the lines using for loop and line function
  - Finally the result is displayed using imshow



## **simple-hough-interactive output**

## CPU Loading:

Using htop we find the loading on each CPU core. In this example we can see that only 1 CPU core is being used and it is loaded upto 96.8%



## **simple-hough-interactive CPU loading**

## 5.

Using a Logitech C200, choose 3 real-time interactive transformations to compare in terms of average frame rate at a given resolution for a range of at least 3 resolutions in a common aspect ratio (e.g. 4:3 for 1280x960, 640x480, 320x240, 160x120, 80x60) by adding time-stamps to analyze the potential throughput. Based on average analysis, pick a reasonable soft real-time deadline (e.g. if average frame rate is 12 Hz, choose a deadline of 100 milliseconds to provide some margin) and convert the processing to SCED\_FIFO and determine if you can meet deadlines with predictability and measure jitter in the frame rate relative to your deadline.

## ANSWER

**Design Concept:** For this part we used the simple-canny, simple-hough and hough-elliptical transforms. We modified the example-stereo-transform-improved program from Prof. Sam Siewerts example codes to include these 3 transforms. The user can select the transform to run by adding an input argument along with the executable e.g. “./Q5 c”. We find the Average Frame Rate by running the 3 transforms for different resolutions and calculating the delays between successive captures. For the second part we use these Average Frame Rates to assign Soft Real-time deadlines which are just ~20ms greater than the frame rates.

**Algorithm Analysis:** For the first part we run the program such that only one transform is being run. The user can input the argument along with the executable to select the transform to run. By individually running the program for each transform and 3 different resolutions we get the Average Frame Rates as seen below.

Resolution	Canny	Hough	Hough Elliptical
640 x 480	30.6	70.3	35.6
320 x 240	9.1	34.6	20.6
160 x 120	2.3	11.1	19.5

Average Frame Rate (ms)

Algorithm:

- Execute the program along with the transform to run. Ex. “./Q5 h” for hough transform
- Based on the input we switch to select the transform to run
- To calculate the average frame rate get the time between 2 successive captures
- Exit by pressing ‘q’

For the second part we decide the soft-real time deadline using the Frame rates found above. Once we get the time between each capture and release of the camera resource, we calculate the jitter by subtracting the delay with the deadlines.

Algorithm:

- Calculate deadlines based on average frame rate from previous program
- Initialize semaphores for each transform, this is used for signalling, initialize canny semaphore to 1 and rest to 0
- Create 3 threads, 1 for each transform(canny, hough and hough-elliptical)

- As the threads are created they wait on semaphores to begin processing. Since canny semaphore is 1 it is acquired and task starts running.
- Once Canny is done it signals Hough and at the end of its processing it signals Hough Elliptical and this keeps on happening
- Since they are signalling each other in a loop only 1 of the tasks has control over the camera
- Inside each task we calculate the frame rate again and subtract it with the deadline to get the jitter
- Repeat the task for each transform with different resolution

Resolution	Canny	Hough	Hough Elliptical
640 x 480	50	90	55
320 x 240	30	55	40
160 x 120	20	30	40

Soft real-time deadline computed(ms)

**Prototype Analysis:** To determine the soft real time deadline and jitter we first need to calculate the average frame rate for each transform. So we use a single process environment where a particular transform is run for 3 separate resolutions based on user input. By determining the appropriate we then select the deadlines for our next program.

Canny:

```

Frame @ 1522724192 sec, 698721354 nsec, dt= 1.95 msec, avedt= 0.65 msec, rate=1537.73 fps
Frame @ 1522724192 sec, 740917790 nsec, dt= 2.19 msec, avedt= 1.03 msec, rate=966.84 fps
Frame @ 1522724192 sec, 793502257 nsec, dt= 1.80 msec, avedt= 1.19 msec, rate=842.49 fps
Frame @ 1522724192 sec, 832864316 nsec, dt= 2.95 msec, avedt= 1.48 msec, rate=675.07 fps
Frame @ 1522724192 sec, 870542053 nsec, dt= 2.29 msec, avedt= 1.60 msec, rate=626.15 fps
Frame @ 1522724192 sec, 908438706 nsec, dt= 2.01 msec, avedt= 1.65 msec, rate=606.57 fps
Frame @ 1522724192 sec, 949322337 nsec, dt= 1.93 msec, avedt= 1.68 msec, rate=595.43 fps
Frame @ 1522724192 sec, 988260899 nsec, dt= 3.43 msec, avedt= 1.85 msec, rate=539.27 fps
Frame @ 1522724193 sec, 27971539 nsec, dt= 1.86 msec, avedt= 1.86 msec, rate=539.04 fps
Frame @ 1522724193 sec, 66109773 nsec, dt= 3.76 msec, avedt= 2.01 msec, rate=496.62 fps
Frame @ 1522724193 sec, 103813927 nsec, dt= 2.02 msec, avedt= 2.01 msec, rate=496.41 fps
Frame @ 1522724193 sec, 146157799 nsec, dt= 1.84 msec, avedt= 2.00 msec, rate=499.53 fps
Frame @ 1522724193 sec, 182991125 nsec, dt= 1.98 msec, avedt= 2.00 msec, rate=499.89 fps
Frame @ 1522724193 sec, 221277942 nsec, dt= 1.73 msec, avedt= 1.98 msec, rate=504.10 fps
Frame @ 1522724193 sec, 259880506 nsec, dt= 2.13 msec, avedt= 1.99 msec, rate=501.92 fps
Frame @ 1522724193 sec, 300628806 nsec, dt= 2.23 msec, avedt= 2.01 msec, rate=498.63 fps
Frame @ 1522724193 sec, 342389764 nsec, dt= 3.59 msec, avedt= 2.09 msec, rate=478.71 fps
Frame @ 1522724193 sec, 382333819 nsec, dt= 5.41 msec, avedt= 2.25 msec, rate=443.50 fps
Frame @ 1522724193 sec, 419677976 nsec, dt= 3.50 msec, avedt= 2.31 msec, rate=432.10 fps
Frame @ 1522724193 sec, 457569128 nsec, dt= 1.82 msec, avedt= 2.29 msec, rate=436.35 fps
Frame @ 1522724193 sec, 502689564 nsec, dt= 1.73 msec, avedt= 2.27 msec, rate=441.06 fps
Frame @ 1522724193 sec, 546616674 nsec, dt= 4.18 msec, avedt= 2.35 msec, rate=426.06 fps
Frame @ 1522724193 sec, 585647069 nsec, dt= 6.41 msec, avedt= 2.51 msec, rate=398.47 fps
Frame @ 1522724193 sec, 624122884 nsec, dt= 2.39 msec, avedt= 2.51 msec, rate=399.20 fps
Frame @ 1522724193 sec, 663259861 nsec, dt= 2.29 msec, avedt= 2.50 msec, rate=400.45 fps
Frame @ 1522724193 sec, 801385899 nsec, dt= 3.12 msec, avedt= 2.52 msec, rate=396.90 fps
Frame @ 1522724193 sec, 947039384 nsec, dt= 2.96 msec, avedt= 2.53 msec, rate=394.53 fps
Frame @ 1522724194 sec, 74130664 nsec, dt= 6.06 msec, avedt= 2.65 msec, rate=377.06 fps
Frame @ 1522724194 sec, 195654734 nsec, dt=12.28 msec, avedt= 2.96 msec, rate=337.53 fps
Frame @ 1522724194 sec, 317288303 nsec, dt=11.87 msec, avedt= 3.24 msec, rate=308.55 fps
Frame @ 1522724194 sec, 452266695 nsec, dt= 2.94 msec, avedt= 3.23 msec, rate=309.42 fps
Frame @ 1522724194 sec, 590617148 nsec, dt= 4.23 msec, avedt= 3.26 msec, rate=306.65 fps
Frame @ 1522724194 sec, 740012773 nsec, dt= 6.77 msec, avedt= 3.36 msec, rate=297.50 fps
Frame @ 1522724194 sec, 863716328 nsec, dt= 7.75 msec, avedt= 3.48 msec, rate=287.09 fps
Frame @ 1522724195 sec, 778122 nsec, dt= 3.29 msec, avedt= 3.48 msec, rate=287.52 fps
Frame @ 1522724195 sec, 118468885 nsec, dt= 4.00 msec, avedt= 3.49 msec, rate=286.38 fps
Frame @ 1522724195 sec, 249504889 nsec, dt= 4.13 msec, avedt= 3.51 msec, rate=285.04 fps
Frame @ 1522724195 sec, 381321886 nsec, dt= 2.95 msec, avedt= 3.49 msec, rate=286.17 fps
Frame @ 1522724195 sec, 517202689 nsec, dt= 2.86 msec, avedt= 3.48 msec, rate=287.44 fps
Frame @ 1522724195 sec, 649182853 nsec, dt= 2.79 msec, avedt= 3.46 msec, rate=288.80 fps
Frame @ 1522724195 sec, 781216266 nsec, dt= 2.87 msec, avedt= 3.45 msec, rate=289.95 fps
Frame @ 1522724195 sec, 917160235 nsec, dt= 2.90 msec, avedt= 3.44 msec, rate=291.01 fps

```

160 x 120

```

Frame @ 1522723751 sec, 440713995 nsec, dt=16.06 msec, avedt= 5.35 msec, rate=186.84 fps
Frame @ 1522723751 sec, 489609745 nsec, dt=16.47 msec, avedt= 8.13 msec, rate=122.99 fps
Frame @ 1522723751 sec, 535638264 nsec, dt= 7.41 msec, avedt= 7.99 msec, rate=125.20 fps
Frame @ 1522723751 sec, 582866942 nsec, dt= 8.83 msec, avedt= 8.13 msec, rate=123.05 fps
Frame @ 1522723751 sec, 627597554 nsec, dt=10.54 msec, avedt= 8.47 msec, rate=118.04 fps
Frame @ 1522723751 sec, 674234236 nsec, dt= 7.84 msec, avedt= 8.39 msec, rate=119.16 fps
Frame @ 1522723751 sec, 723493317 nsec, dt= 9.03 msec, avedt= 8.46 msec, rate=118.16 fps
Frame @ 1522723751 sec, 771647655 nsec, dt=10.02 msec, avedt= 8.62 msec, rate=116.03 fps
Frame @ 1522723751 sec, 826067534 nsec, dt= 9.13 msec, avedt= 8.66 msec, rate=115.41 fps
Frame @ 1522723751 sec, 873410961 nsec, dt=11.65 msec, avedt= 8.91 msec, rate=112.19 fps
Frame @ 1522723751 sec, 920758972 nsec, dt= 7.98 msec, avedt= 8.84 msec, rate=113.11 fps
Frame @ 1522723751 sec, 971231127 nsec, dt= 9.13 msec, avedt= 8.86 msec, rate=112.84 fps
Frame @ 1522723752 sec, 21495701 nsec, dt=11.58 msec, avedt= 9.04 msec, rate=110.58 fps
Frame @ 1522723752 sec, 65410485 nsec, dt=11.51 msec, avedt= 9.20 msec, rate=108.73 fps
Frame @ 1522723752 sec, 114758982 nsec, dt= 7.41 msec, avedt= 9.09 msec, rate=109.98 fps
Frame @ 1522723752 sec, 167236707 nsec, dt=10.04 msec, avedt= 9.14 msec, rate=109.35 fps
Frame @ 1522723752 sec, 215679960 nsec, dt=14.40 msec, avedt= 9.42 msec, rate=106.14 fps
Frame @ 1522723752 sec, 263892215 nsec, dt= 7.13 msec, avedt= 9.31 msec, rate=107.45 fps
Frame @ 1522723752 sec, 310995894 nsec, dt=10.33 msec, avedt= 9.36 msec, rate=106.89 fps
Frame @ 1522723752 sec, 362890540 nsec, dt= 9.95 msec, avedt= 9.38 msec, rate=106.58 fps
Frame @ 1522723752 sec, 506679226 nsec, dt=12.35 msec, avedt= 9.51 msec, rate=105.14 fps
Frame @ 1522723752 sec, 608861196 nsec, dt=13.06 msec, avedt= 9.66 msec, rate=103.53 fps
Frame @ 1522723752 sec, 754084955 nsec, dt= 8.89 msec, avedt= 9.63 msec, rate=103.86 fps
Frame @ 1522723752 sec, 899002717 nsec, dt=11.42 msec, avedt= 9.70 msec, rate=103.12 fps
Frame @ 1522723753 sec, 28927998 nsec, dt=23.75 msec, avedt=10.22 msec, rate=97.87 fps
Frame @ 1522723753 sec, 161293512 nsec, dt=26.29 msec, avedt=10.79 msec, rate=92.66 fps
Frame @ 1522723753 sec, 288821975 nsec, dt=27.41 msec, avedt=11.36 msec, rate=87.99 fps
Frame @ 1522723753 sec, 433578238 nsec, dt=16.94 msec, avedt=11.55 msec, rate=86.57 fps
Frame @ 1522723753 sec, 569148063 nsec, dt=11.43 msec, avedt=11.55 msec, rate=86.60 fps
Frame @ 1522723753 sec, 689956989 nsec, dt=22.64 msec, avedt=11.89 msec, rate=84.08 fps
Frame @ 1522723753 sec, 834753669 nsec, dt=18.12 msec, avedt=12.08 msec, rate=82.77 fps
Frame @ 1522723753 sec, 962460548 nsec, dt=29.47 msec, avedt=12.59 msec, rate=79.40 fps
Frame @ 1522723754 sec, 113946848 nsec, dt=11.81 msec, avedt=12.57 msec, rate=79.55 fps
Frame @ 1522723754 sec, 205886637 nsec, dt=21.12 msec, avedt=12.81 msec, rate=78.07 fps
Frame @ 1522723754 sec, 366633124 nsec, dt= 6.83 msec, avedt=12.65 msec, rate=79.07 fps

```

320 x 240

```

Frame @ 1522723318 sec, 217799110 nsec, dt=28.12 msec, avedt= 9.37 msec, rate=106.69 fps
Frame @ 1522723318 sec, 292469947 nsec, dt=29.19 msec, avedt=14.33 msec, rate=69.80 fps
Frame @ 1522723318 sec, 390639270 nsec, dt=29.73 msec, avedt=17.41 msec, rate=57.44 fps
Frame @ 1522723318 sec, 459541985 nsec, dt=37.59 msec, avedt=20.77 msec, rate=48.14 fps
Frame @ 1522723318 sec, 546045565 nsec, dt=27.94 msec, avedt=21.80 msec, rate=45.88 fps
Frame @ 1522723318 sec, 622908969 nsec, dt=35.88 msec, avedt=23.56 msec, rate=42.45 fps
Frame @ 1522723318 sec, 726532165 nsec, dt=33.22 msec, avedt=24.63 msec, rate=40.60 fps
Frame @ 1522723318 sec, 832771758 nsec, dt=50.73 msec, avedt=27.24 msec, rate=36.71 fps
Frame @ 1522723318 sec, 930715582 nsec, dt=45.58 msec, avedt=28.91 msec, rate=34.59 fps
Frame @ 1522723319 sec, 11129792 nsec, dt=34.30 msec, avedt=29.36 msec, rate=34.06 fps
Frame @ 1522723319 sec, 87221952 nsec, dt=36.99 msec, avedt=29.94 msec, rate=33.40 fps
Frame @ 1522723319 sec, 165044432 nsec, dt=31.76 msec, avedt=30.07 msec, rate=33.25 fps
Frame @ 1522723319 sec, 251259263 nsec, dt=33.78 msec, avedt=30.32 msec, rate=32.98 fps
Frame @ 1522723319 sec, 329831738 nsec, dt=35.48 msec, avedt=30.64 msec, rate=32.63 fps
Frame @ 1522723319 sec, 418421967 nsec, dt=32.81 msec, avedt=30.77 msec, rate=32.50 fps
Frame @ 1522723319 sec, 493542552 nsec, dt=33.76 msec, avedt=30.94 msec, rate=32.32 fps
Frame @ 1522723319 sec, 620858898 nsec, dt=28.28 msec, avedt=30.80 msec, rate=32.47 fps
Frame @ 1522723319 sec, 792818731 nsec, dt=37.15 msec, avedt=31.11 msec, rate=32.14 fps
Frame @ 1522723319 sec, 960321016 nsec, dt=66.60 msec, avedt=32.80 msec, rate=30.48 fps
Frame @ 1522723320 sec, 84992300 nsec, dt=36.08 msec, avedt=32.95 msec, rate=30.35 fps
Frame @ 1522723320 sec, 216660279 nsec, dt=36.10 msec, avedt=33.09 msec, rate=30.22 fps
Frame @ 1522723320 sec, 334786863 nsec, dt=45.70 msec, avedt=33.62 msec, rate=29.75 fps
Frame @ 1522723320 sec, 418840379 nsec, dt=34.23 msec, avedt=33.64 msec, rate=29.73 fps
Frame @ 1522723320 sec, 551616933 nsec, dt=36.85 msec, avedt=33.76 msec, rate=29.62 fps
Frame @ 1522723320 sec, 699491703 nsec, dt=36.06 msec, avedt=33.85 msec, rate=29.54 fps
Frame @ 1522723320 sec, 831787177 nsec, dt=41.00 msec, avedt=34.10 msec, rate=29.32 fps
Frame @ 1522723320 sec, 946678954 nsec, dt=44.34 msec, avedt=34.46 msec, rate=29.02 fps
Frame @ 1522723321 sec, 87103615 nsec, dt=31.67 msec, avedt=34.36 msec, rate=29.10 fps
Frame @ 1522723321 sec, 228436020 nsec, dt=37.78 msec, avedt=34.47 msec, rate=29.01 fps
Frame @ 1522723321 sec, 352807056 nsec, dt=41.88 msec, avedt=34.71 msec, rate=28.81 fps
Frame @ 1522723321 sec, 477077259 nsec, dt=37.13 msec, avedt=34.78 msec, rate=28.75 fps
Frame @ 1522723321 sec, 619921319 nsec, dt=29.16 msec, avedt=34.61 msec, rate=28.89 fps
Frame @ 1522723321 sec, 743366612 nsec, dt=37.50 msec, avedt=34.70 msec, rate=28.82 fps
Got Quit

```

640 x 480

Hough:

```
Frame @ 1522724253 sec, 29955696 nsec, dt= 5.03 msec, avedt= 2.64 msec, rate=378.86 fps
Frame @ 1522724253 sec, 74827383 nsec, dt=13.39 msec, avedt= 4.79 msec, rate=208.75 fps
Frame @ 1522724253 sec, 122139719 nsec, dt= 9.12 msec, avedt= 5.51 msec, rate=181.43 fps
Frame @ 1522724253 sec, 168095315 nsec, dt=10.39 msec, avedt= 6.21 msec, rate=161.08 fps
Frame @ 1522724253 sec, 222689767 nsec, dt= 9.34 msec, avedt= 6.60 msec, rate=151.51 fps
Frame @ 1522724253 sec, 266769875 nsec, dt=15.04 msec, avedt= 7.54 msec, rate=132.66 fps
Frame @ 1522724253 sec, 317966518 nsec, dt= 8.26 msec, avedt= 7.61 msec, rate=131.39 fps
Frame @ 1522724253 sec, 380174000 nsec, dt=15.44 msec, avedt= 8.32 msec, rate=120.15 fps
Frame @ 1522724253 sec, 429912652 nsec, dt=18.36 msec, avedt= 9.16 msec, rate=109.18 fps
Frame @ 1522724253 sec, 480587965 nsec, dt=13.51 msec, avedt= 9.49 msec, rate=105.34 fps
Frame @ 1522724253 sec, 530373201 nsec, dt=11.78 msec, avedt= 9.66 msec, rate=103.55 fps
Frame @ 1522724253 sec, 579822772 nsec, dt=13.85 msec, avedt= 9.94 msec, rate=100.64 fps
Frame @ 1522724253 sec, 625301121 nsec, dt=10.44 msec, avedt= 9.97 msec, rate=100.32 fps
Frame @ 1522724253 sec, 681229313 nsec, dt= 9.43 msec, avedt= 9.94 msec, rate=100.64 fps
Frame @ 1522724253 sec, 730691968 nsec, dt=16.64 msec, avedt=10.31 msec, rate=97.01 fps
Frame @ 1522724253 sec, 785024755 nsec, dt=10.88 msec, avedt=10.34 msec, rate=96.73 fps
Frame @ 1522724253 sec, 833455083 nsec, dt=16.55 msec, avedt=10.65 msec, rate=93.91 fps
Frame @ 1522724253 sec, 885568469 nsec, dt=11.82 msec, avedt=10.70 msec, rate=93.42 fps
Frame @ 1522724253 sec, 938151435 nsec, dt=14.34 msec, avedt=10.87 msec, rate=92.00 fps
Frame @ 1522724254 sec, 11182675 nsec, dt=16.44 msec, avedt=11.11 msec, rate=89.99 fps
Frame @ 1522724254 sec, 104663605 nsec, dt=35.34 msec, avedt=12.12 msec, rate=82.50 fps
Frame @ 1522724254 sec, 264978069 nsec, dt=41.02 msec, avedt=13.28 msec, rate=75.32 fps
Frame @ 1522724254 sec, 368517179 nsec, dt=55.07 msec, avedt=14.88 msec, rate=67.18 fps
Frame @ 1522724254 sec, 509710609 nsec, dt=39.65 msec, avedt=15.80 msec, rate=63.28 fps
Frame @ 1522724254 sec, 658220738 nsec, dt=46.94 msec, avedt=16.91 msec, rate=59.12 fps
Frame @ 1522724254 sec, 768324969 nsec, dt=61.42 msec, avedt=18.45 msec, rate=54.20 fps
Frame @ 1522724254 sec, 921438733 nsec, dt=41.57 msec, avedt=19.22 msec, rate=52.03 fps
Frame @ 1522724255 sec, 32505207 nsec, dt=55.63 msec, avedt=20.39 msec, rate=49.03 fps
Frame @ 1522724255 sec, 177487944 nsec, dt=38.26 msec, avedt=20.95 msec, rate=47.73 fps
Frame @ 1522724255 sec, 322485348 nsec, dt=48.96 msec, avedt=21.80 msec, rate=45.87 fps
Frame @ 1522724255 sec, 447135810 nsec, dt=58.07 msec, avedt=22.87 msec, rate=43.73 fps
Frame @ 1522724255 sec, 573798009 nsec, dt=52.36 msec, avedt=23.71 msec, rate=42.18 fps
Frame @ 1522724255 sec, 705266007 nsec, dt=48.17 msec, avedt=24.39 msec, rate=41.00 fps
Frame @ 1522724255 sec, 846267772 nsec, dt=42.26 msec, avedt=24.87 msec, rate=40.20 fps
Frame @ 1522724255 sec, 979719173 nsec, dt=52.08 msec, avedt=25.59 msec, rate=39.08 fps
Frame @ 1522724256 sec, 107532113 nsec, dt=48.64 msec, avedt=26.18 msec, rate=38.20 fps
Frame @ 1522724256 sec, 242963167 nsec, dt=44.77 msec, avedt=26.65 msec, rate=37.53 fps
Got_Quit
```

160 x 120

```
Frame @ 1522724015 sec, 155997094 nsec, dt= 0.00 msec, avedt= 0.00 msec, rate= 0.00 fps
Frame @ 1522724015 sec, 261900028 nsec, dt=12.46 msec, avedt= 0.00 msec, rate= 0.00 fps
Frame @ 1522724015 sec, 333405700 nsec, dt=11.43 msec, avedt= 3.81 msec, rate=262.54 fps
Frame @ 1522724015 sec, 407482605 nsec, dt=27.95 msec, avedt= 9.84 msec, rate=101.59 fps
Frame @ 1522724015 sec, 465471037 nsec, dt=23.19 msec, avedt=12.51 msec, rate=79.92 fps
Frame @ 1522724015 sec, 537218541 nsec, dt=20.83 msec, avedt=13.90 msec, rate=71.95 fps
Frame @ 1522724015 sec, 614209008 nsec, dt=35.52 msec, avedt=16.99 msec, rate=58.86 fps
Frame @ 1522724015 sec, 688978658 nsec, dt=35.49 msec, avedt=19.30 msec, rate=51.81 fps
Frame @ 1522724015 sec, 768211276 nsec, dt=35.02 msec, avedt=21.05 msec, rate=47.51 fps
Frame @ 1522724015 sec, 855334424 nsec, dt=39.85 msec, avedt=22.93 msec, rate=43.61 fps
Frame @ 1522724015 sec, 927989838 nsec, dt=47.80 msec, avedt=25.19 msec, rate=39.70 fps
Frame @ 1522724016 sec, 13404497 nsec, dt=35.86 msec, avedt=26.08 msec, rate=38.35 fps
Frame @ 1522724016 sec, 83728177 nsec, dt=43.71 msec, avedt=27.44 msec, rate=36.45 fps
Frame @ 1522724016 sec, 145743498 nsec, dt=33.85 msec, avedt=27.89 msec, rate=35.85 fps
Frame @ 1522724016 sec, 230961409 nsec, dt=25.29 msec, avedt=27.72 msec, rate=36.08 fps
Frame @ 1522724016 sec, 299456685 nsec, dt=46.27 msec, avedt=28.88 msec, rate=34.63 fps
Frame @ 1522724016 sec, 419333022 nsec, dt=32.46 msec, avedt=29.09 msec, rate=34.38 fps
Frame @ 1522724016 sec, 548040215 nsec, dt=72.62 msec, avedt=31.51 msec, rate=31.74 fps
Frame @ 1522724016 sec, 669012462 nsec, dt=89.85 msec, avedt=34.58 msec, rate=28.92 fps
Frame @ 1522724016 sec, 808396248 nsec, dt=76.68 msec, avedt=36.68 msec, rate=27.26 fps
Frame @ 1522724016 sec, 950148601 nsec, dt=79.90 msec, avedt=38.74 msec, rate=25.81 fps
Frame @ 1522724017 sec, 72994334 nsec, dt=95.87 msec, avedt=41.34 msec, rate=24.19 fps
Frame @ 1522724017 sec, 209969553 nsec, dt=81.83 msec, avedt=43.10 msec, rate=23.20 fps
Frame @ 1522724017 sec, 3400334487 nsec, dt=84.57 msec, avedt=44.83 msec, rate=22.31 fps
Frame @ 1522724017 sec, 466512279 nsec, dt=85.84 msec, avedt=46.47 msec, rate=21.52 fps
Frame @ 1522724017 sec, 595366221 nsec, dt=79.14 msec, avedt=47.72 msec, rate=20.95 fps
Frame @ 1522724017 sec, 740107719 nsec, dt=75.93 msec, avedt=48.77 msec, rate=20.50 fps
Frame @ 1522724017 sec, 862440790 nsec, dt=86.12 msec, avedt=50.10 msec, rate=19.96 fps
Frame @ 1522724017 sec, 994620292 nsec, dt=73.63 msec, avedt=50.91 msec, rate=19.64 fps
Frame @ 1522724018 sec, 131021682 nsec, dt=69.78 msec, avedt=51.54 msec, rate=19.40 fps
Frame @ 1522724018 sec, 285086116 nsec, dt=77.01 msec, avedt=52.36 msec, rate=19.10 fps
Frame @ 1522724018 sec, 416537290 nsec, dt=95.18 msec, avedt=53.70 msec, rate=18.62 fps
Frame @ 1522724018 sec, 538538779 nsec, dt=93.25 msec, avedt=54.90 msec, rate=18.21 fps
Frame @ 1522724018 sec, 658557449 nsec, dt=83.54 msec, avedt=55.74 msec, rate=17.94 fps
Frame @ 1522724018 sec, 800511717 nsec, dt=69.97 msec, avedt=56.15 msec, rate=17.81 fps
Frame @ 1522724018 sec, 929894072 nsec, dt=80.96 msec, avedt=56.84 msec, rate=17.59 fps
Frame @ 1522724019 sec, 63081317 nsec, dt=78.89 msec, avedt=57.43 msec, rate=17.41 fps
Frame @ 1522724019 sec, 191698761 nsec, dt=73.28 msec, avedt=57.85 msec, rate=17.29 fps
Frame @ 1522724019 sec, 327738070 nsec, dt=72.78 msec, avedt=58.23 msec, rate=17.17 fps
Frame @ 1522724019 sec, 461359145 nsec, dt=72.89 msec, avedt=58.60 msec, rate=17.06 fps
Got_Quit
```

320 x 240

```

Frame @ 1522723568 sec, 536681550 nsec, dt=17.93 msec, avedt= 6.00 msec, rate= 6.00 fps
Frame @ 1522723569 sec, 571220156 nsec, dt=37.17 msec, avedt=12.39 msec, rate=80.70 fps
Frame @ 1522723569 sec, 674754792 nsec, dt=58.19 msec, avedt=23.84 msec, rate=41.94 fps
Frame @ 1522723569 sec, 786220208 nsec, dt=49.16 msec, avedt=28.91 msec, rate=34.59 fps
Frame @ 1522723569 sec, 892309243 nsec, dt=62.14 msec, avedt=34.45 msec, rate=29.03 fps
Frame @ 1522723570 sec, 6528890 nsec, dt=65.13 msec, avedt=38.83 msec, rate=25.75 fps
Frame @ 1522723570 sec, 109714529 nsec, dt=68.46 msec, avedt=42.53 msec, rate=23.51 fps
Frame @ 1522723570 sec, 214170075 nsec, dt=62.52 msec, avedt=44.75 msec, rate=22.34 fps
Frame @ 1522723570 sec, 322573928 nsec, dt=60.05 msec, avedt=46.28 msec, rate=21.61 fps
Frame @ 1522723570 sec, 430029704 nsec, dt=62.69 msec, avedt=47.78 msec, rate=20.93 fps
Frame @ 1522723570 sec, 535362078 nsec, dt=58.90 msec, avedt=48.70 msec, rate=20.53 fps
Frame @ 1522723570 sec, 734465653 nsec, dt=60.86 msec, avedt=49.64 msec, rate=20.15 fps
Frame @ 1522723570 sec, 933039649 nsec, dt=152.47 msec, avedt=56.98 msec, rate=17.55 fps
Frame @ 1522723571 sec, 156958807 nsec, dt=152.15 msec, avedt=63.33 msec, rate=15.79 fps
Frame @ 1522723571 sec, 355557470 nsec, dt=174.28 msec, avedt=70.26 msec, rate=14.23 fps
Frame @ 1522723571 sec, 542875291 nsec, dt=147.39 msec, avedt=74.80 msec, rate=13.37 fps
Frame @ 1522723571 sec, 730267779 nsec, dt=142.81 msec, avedt=78.58 msec, rate=12.73 fps
Frame @ 1522723571 sec, 927818864 nsec, dt=141.84 msec, avedt=81.91 msec, rate=12.21 fps
Frame @ 1522723572 sec, 120915897 nsec, dt=151.33 msec, avedt=85.38 msec, rate=11.71 fps
Frame @ 1522723572 sec, 345241802 nsec, dt=145.41 msec, avedt=88.24 msec, rate=11.33 fps
Frame @ 1522723572 sec, 547853604 nsec, dt=171.53 msec, avedt=92.02 msec, rate=10.87 fps
Frame @ 1522723572 sec, 731187202 nsec, dt=159.14 msec, avedt=94.94 msec, rate=10.53 fps
Frame @ 1522723572 sec, 921397587 nsec, dt=137.08 msec, avedt=96.70 msec, rate=10.34 fps
Frame @ 1522723573 sec, 118439426 nsec, dt=144.61 msec, avedt=98.61 msec, rate=10.14 fps
Frame @ 1522723573 sec, 302563769 nsec, dt=148.23 msec, avedt=100.52 msec, rate= 9.95 fps
Frame @ 1522723573 sec, 490875584 nsec, dt=138.27 msec, avedt=101.92 msec, rate= 9.81 fps
Frame @ 1522723573 sec, 678589902 nsec, dt=137.00 msec, avedt=103.17 msec, rate= 9.69 fps
Frame @ 1522723573 sec, 870866607 nsec, dt=145.30 msec, avedt=104.63 msec, rate= 9.56 fps
Frame @ 1522723574 sec, 81941601 nsec, dt=145.99 msec, avedt=106.00 msec, rate= 9.43 fps
Frame @ 1522723574 sec, 303422776 nsec, dt=160.31 msec, avedt=107.76 msec, rate= 9.28 fps
Got Quit

```

640 x 480

## Hough Elliptical:

```

circles.size = 0
Frame @ 1522724293 sec, 411562401 nsec, dt=18.04 msec, avedt= 9.58 msec, rate=104.42 fps
circles.size = 0
Frame @ 1522724293 sec, 486733458 nsec, dt=31.05 msec, avedt=13.87 msec, rate=72.09 fps
circles.size = 0
Frame @ 1522724293 sec, 550729344 nsec, dt=39.33 msec, avedt=18.12 msec, rate=55.20 fps
circles.size = 0
Frame @ 1522724293 sec, 613611987 nsec, dt= 8.72 msec, avedt=16.77 msec, rate=59.62 fps
circles.size = 0
Frame @ 1522724293 sec, 681728761 nsec, dt=26.91 msec, avedt=18.04 msec, rate=55.43 fps
circles.size = 0
Frame @ 1522724293 sec, 751114192 nsec, dt=31.46 msec, avedt=19.53 msec, rate=51.20 fps
circles.size = 0
Frame @ 1522724293 sec, 814416499 nsec, dt=31.91 msec, avedt=20.77 msec, rate=48.15 fps
circles.size = 0
Frame @ 1522724293 sec, 880336121 nsec, dt=23.52 msec, avedt=21.02 msec, rate=47.58 fps
circles.size = 0
Frame @ 1522724293 sec, 946314243 nsec, dt=26.54 msec, avedt=21.48 msec, rate=46.56 fps
circles.size = 0
Frame @ 1522724294 sec, 13707355 nsec, dt=28.62 msec, avedt=22.03 msec, rate=45.39 fps
circles.size = 0
Frame @ 1522724294 sec, 83400867 nsec, dt=29.34 msec, avedt=22.55 msec, rate=44.34 fps
circles.size = 0
Frame @ 1522724294 sec, 265491594 nsec, dt=30.15 msec, avedt=23.06 msec, rate=43.37 fps
circles.size = 0
Frame @ 1522724294 sec, 528031674 nsec, dt=143.67 msec, avedt=30.60 msec, rate=32.68 fps
circles.size = 0
Frame @ 1522724294 sec, 806029145 nsec, dt=130.92 msec, avedt=36.50 msec, rate=27.40 fps
circles.size = 0
Frame @ 1522724295 sec, 50731267 nsec, dt=158.90 msec, avedt=43.30 msec, rate=23.10 fps
circles.size = 0
Frame @ 1522724295 sec, 328361491 nsec, dt=137.78 msec, avedt=48.27 msec, rate=20.72 fps
circles.size = 0
Frame @ 1522724295 sec, 589492914 nsec, dt=114.27 msec, avedt=51.57 msec, rate=19.39 fps
circles.size = 0
Frame @ 1522724295 sec, 866714891 nsec, dt=126.21 msec, avedt=55.12 msec, rate=18.14 fps
circles.size = 0
Frame @ 1522724296 sec, 131165041 nsec, dt=141.19 msec, avedt=59.04 msec, rate=16.94 fps
circles.size = 0
Frame @ 1522724296 sec, 395423942 nsec, dt=140.92 msec, avedt=62.60 msec, rate=15.98 fps
circles.size = 0
Frame @ 1522724296 sec, 646436853 nsec, dt=152.03 msec, avedt=66.32 msec, rate=15.08 fps
Got Quit

```

160 x 120

```

Frame @ 1522724060 sec, 798709896 nsec, dt=26.07 msec, avedt= 8.69 msec, rate=115.06 fps
Circles.size = 0
Frame @ 1522724060 sec, 848892965 nsec, dt=16.69 msec, avedt=10.69 msec, rate=93.55 fps
Circles.size = 0
Frame @ 1522724060 sec, 910288123 nsec, dt=10.51 msec, avedt=10.65 msec, rate=93.85 fps
Circles.size = 0
Frame @ 1522724060 sec, 985940599 nsec, dt=24.99 msec, avedt=13.04 msec, rate=76.66 fps
Circles.size = 0
Frame @ 1522724061 sec, 48036418 nsec, dt=31.10 msec, avedt=15.62 msec, rate=64.00 fps
Circles.size = 0
Frame @ 1522724061 sec, 111672477 nsec, dt=23.83 msec, avedt=16.65 msec, rate=60.06 fps
Circles.size = 0
Frame @ 1522724061 sec, 192009170 nsec, dt=26.22 msec, avedt=17.71 msec, rate=56.46 fps
Circles.size = 0
Frame @ 1522724061 sec, 245995629 nsec, dt=39.40 msec, avedt=19.88 msec, rate=50.30 fps
Circles.size = 0
Frame @ 1522724061 sec, 318472794 nsec, dt=14.92 msec, avedt=19.43 msec, rate=51.47 fps
Circles.size = 0
Frame @ 1522724061 sec, 381760688 nsec, dt=33.58 msec, avedt=20.61 msec, rate=48.52 fps
Circles.size = 0
Frame @ 1522724061 sec, 453260026 nsec, dt=21.15 msec, avedt=20.65 msec, rate=48.43 fps
Circles.size = 0
Frame @ 1522724061 sec, 514291770 nsec, dt=30.81 msec, avedt=21.38 msec, rate=46.78 fps
Circles.size = 0
Frame @ 1522724061 sec, 579318569 nsec, dt=23.83 msec, avedt=21.54 msec, rate=46.43 fps
Circles.size = 0
Frame @ 1522724061 sec, 875796264 nsec, dt=28.04 msec, avedt=21.95 msec, rate=45.57 fps
Circles.size = 0
Frame @ 1522724062 sec, 130214417 nsec, dt=154.95 msec, avedt=29.77 msec, rate=33.59 fps
Circles.size = 0
Frame @ 1522724062 sec, 406401419 nsec, dt=137.38 msec, avedt=35.75 msec, rate=27.97 fps
Circles.size = 0
Frame @ 1522724062 sec, 693977176 nsec, dt=141.02 msec, avedt=41.29 msec, rate=24.22 fps
Circles.size = 0
Frame @ 1522724062 sec, 953503710 nsec, dt=190.35 msec, avedt=48.74 msec, rate=20.52 fps
Circles.size = 0
Frame @ 1522724063 sec, 193039133 nsec, dt=180.61 msec, avedt=55.02 msec, rate=18.17 fps
Circles.size = 0
Frame @ 1522724063 sec, 451166011 nsec, dt=154.27 msec, avedt=59.53 msec, rate=16.80 fps

```

320 x 240

```

Frame @ 1522723638 sec, 998302922 nsec, dt=39.99 msec, avedt=13.33 msec, rate=75.02 fps
Circles.size = 0
Frame @ 1522723639 sec, 75125735 nsec, dt=42.03 msec, avedt=20.51 msec, rate=48.77 fps
Circles.size = 0
Frame @ 1522723639 sec, 155504358 nsec, dt=34.61 msec, avedt=23.33 msec, rate=42.87 fps
Circles.size = 0
Frame @ 1522723639 sec, 252730366 nsec, dt=34.46 msec, avedt=25.18 msec, rate=39.71 fps
Circles.size = 0
Frame @ 1522723639 sec, 351573864 nsec, dt=48.51 msec, avedt=28.51 msec, rate=35.07 fps
Circles.size = 0
Frame @ 1522723639 sec, 450243696 nsec, dt=49.71 msec, avedt=31.16 msec, rate=32.09 fps
Circles.size = 0
Frame @ 1522723639 sec, 535673618 nsec, dt=50.82 msec, avedt=33.35 msec, rate=29.99 fps
Circles.size = 0
Frame @ 1522723639 sec, 630534726 nsec, dt=42.24 msec, avedt=34.24 msec, rate=29.21 fps
Circles.size = 0
Frame @ 1522723639 sec, 723871177 nsec, dt=49.06 msec, avedt=35.58 msec, rate=28.10 fps
Circles.size = 0
Frame @ 1522723639 sec, 839246063 nsec, dt=46.68 msec, avedt=36.51 msec, rate=27.39 fps
Circles.size = 0
Frame @ 1522723640 sec, 80705345 nsec, dt=66.03 msec, avedt=38.78 msec, rate=25.79 fps
Circles.size = 0
Frame @ 1522723640 sec, 355329319 nsec, dt=182.93 msec, avedt=49.08 msec, rate=20.38 fps
Circles.size = 0
Frame @ 1522723640 sec, 632388110 nsec, dt=191.62 msec, avedt=58.58 msec, rate=17.07 fps
Circles.size = 0
Frame @ 1522723640 sec, 881755088 nsec, dt=198.78 msec, avedt=67.34 msec, rate=14.85 fps
Circles.size = 0
Frame @ 1522723641 sec, 154023828 nsec, dt=184.54 msec, avedt=74.24 msec, rate=13.47 fps
Circles.size = 0
Frame @ 1522723641 sec, 431274118 nsec, dt=192.95 msec, avedt=80.83 msec, rate=12.37 fps
Circles.size = 0
Frame @ 1522723641 sec, 713701789 nsec, dt=199.36 msec, avedt=87.07 msec, rate=11.49 fps
Circles.size = 0
Frame @ 1522723641 sec, 981784891 nsec, dt=212.60 msec, avedt=93.35 msec, rate=10.71 fps
Circles.size = 0
Frame @ 1522723642 sec, 238453236 nsec, dt=220.56 msec, avedt=99.40 msec, rate=10.06 fps
Circles.size = 0
Frame @ 1522723642 sec, 482490750 nsec, dt=187.42 msec, avedt=103.40 msec, rate= 9.67 fps

```

640 x 480

The camera is the shared resource between the 3 threads, so to make sure that only 1 of them is using the resource we use a binary semaphore for signalling. The threads are scheduled using FIFO scheduling policy. We also use the semaphores to decide which task should be running in what order.

```

Timestamp when HOUGH captures stopped Seconds:1522725083 and Nanoseconds:75124963
Hough Jitter obtained is -12 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522725083 and Nanoseconds:75209296
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522725083 and Nanoseconds:113616787
circles.size = 0
Hough Eclipse Jitter obtained is 2 ms

Timestamp when Canny capture started: Seconds:1522725083 and Nanoseconds:113685704
Timestamp when CANNY capture stopped Seconds:1522725083 and Nanoseconds:151363866
Canny Jitter obtained is -17 nanoseconds

Timestamp when HOUGH capture started: Seconds:1522725083 and Nanoseconds:151414783
Timestamp when HOUGH captures stopped Seconds:1522725083 and Nanoseconds:194478076
Hough Jitter obtained is -13 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522725083 and Nanoseconds:194569576
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522725083 and Nanoseconds:231978990
circles.size = 0
Hough Eclipse Jitter obtained is 3 ms

Timestamp when Canny capture started: Seconds:1522725083 and Nanoseconds:232063073
Timestamp when CANNY capture stopped Seconds:1522725083 and Nanoseconds:269502737
Canny Jitter obtained is -17 nanoseconds

Timestamp when HOUGH capture started: Seconds:1522725083 and Nanoseconds:269552487
Timestamp when HOUGH captures stopped Seconds:1522725083 and Nanoseconds:314581850
Hough Jitter obtained is -15 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522725083 and Nanoseconds:314641517
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522725083 and Nanoseconds:351223270
circles.size = 0
Hough Eclipse Jitter obtained is 4 ms

Timestamp when Canny capture started: Seconds:1522725083 and Nanoseconds:351293520
Timestamp when CANNY capture stopped Seconds:1522725083 and Nanoseconds:388256771
Canny Jitter obtained is -16 nanoseconds

Timestamp when HOUGH capture started: Seconds:1522725083 and Nanoseconds:388303687
Timestamp when HOUGH captures stopped Seconds:1522725083 and Nanoseconds:432004643
Hough Jitter obtained is -13 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522725083 and Nanoseconds:432071226
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522725083 and Nanoseconds:468577146
circles.size = 0
Hough Eclipse Jitter obtained is 4 ms

Timestamp when Canny capture started: Seconds:1522725083 and Nanoseconds:468665229
Timestamp when CANNY capture stopped Seconds:1522725083 and Nanoseconds:507185386

```

160 x 120

```

Timestamp when Canny capture started: Seconds:1522724974 and Nanoseconds:674324393
Timestamp when CANNY capture stopped Seconds:1522724974 and Nanoseconds:718418182
Canny Jitter obtained is -14 nanoseconds

Timestamp when HOUGH capture started: Seconds:1522724974 and Nanoseconds:718473515
Timestamp when HOUGH captures stopped Seconds:1522724974 and Nanoseconds:779905187
Hough Jitter obtained is -6 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522724974 and Nanoseconds:780070353
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522724974 and Nanoseconds:834578405
circles.size = 0
Hough Eclipse Jitter obtained is -14 ms

Timestamp when Canny capture started: Seconds:1522724974 and Nanoseconds:834712488
Timestamp when CANNY capture stopped Seconds:1522724974 and Nanoseconds:883315663
Canny Jitter obtained is -18 nanoseconds

Timestamp when HOUGH capture started: Seconds:1522724974 and Nanoseconds:883436829
Timestamp when HOUGH captures stopped Seconds:1522724974 and Nanoseconds:979700268
Hough Jitter obtained is -41 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522724974 and Nanoseconds:979805684
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522724975 and Nanoseconds:26846037
circles.size = 0
Hough Eclipse Jitter obtained is -8 ms

Timestamp when Canny capture started: Seconds:1522724975 and Nanoseconds:26922619
Timestamp when CANNY capture stopped Seconds:1522724975 and Nanoseconds:74457718
Canny Jitter obtained is -17 nanoseconds

Timestamp when HOUGH capture started: Seconds:1522724975 and Nanoseconds:74509635
Timestamp when HOUGH captures stopped Seconds:1522724975 and Nanoseconds:142674679
Hough Jitter obtained is -13 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522724975 and Nanoseconds:142749429
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522724975 and Nanoseconds:184843814
circles.size = 0
Hough Eclipse Jitter obtained is -2 ms

Timestamp when Canny capture started: Seconds:1522724975 and Nanoseconds:184905563
Timestamp when CANNY capture stopped Seconds:1522724975 and Nanoseconds:231812500
Canny Jitter obtained is -16 nanoseconds

Timestamp when HOUGH capture started: Seconds:1522724975 and Nanoseconds:231914416
Timestamp when HOUGH captures stopped Seconds:1522724975 and Nanoseconds:329815845
Hough Jitter obtained is -42 nanoseconds

```

320 x 240

```

Timestamp when HOUGH capture started: Seconds:1522724764 and Nanoseconds:348654751
Timestamp when HOUGH captures stopped Seconds:1522724764 and Nanoseconds:542103064
Hough Jitter obtained is -103 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522724764 and Nanoseconds:542218897
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522724764 and Nanoseconds:642116408
circles.size = 0
Hough Eclipse Jitter obtained is -44 ms

Timestamp when Canny capture started: Seconds:1522724764 and Nanoseconds:642200824
Timestamp when CANNY capture stopped Seconds:1522724764 and Nanoseconds:718851570
Canny Jitter obtained is -26 nanoseconds

Timestamp when HOUGH capture started: Seconds:1522724764 and Nanoseconds:718898070
Timestamp when HOUGH captures stopped Seconds:1522724765 and Nanoseconds:140377635
Hough Jitter obtained is -332 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522724765 and Nanoseconds:140472135
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522724765 and Nanoseconds:226972233
circles.size = 0
Hough Eclipse Jitter obtained is -31 ms

Timestamp when Canny capture started: Seconds:1522724765 and Nanoseconds:227053066
Timestamp when CANNY capture stopped Seconds:1522724765 and Nanoseconds:294303125
Canny Jitter obtained is -17 nanoseconds

Timestamp when HOUGH capture started: Seconds:1522724765 and Nanoseconds:294407791
Timestamp when HOUGH captures stopped Seconds:1522724765 and Nanoseconds:490817667
Hough Jitter obtained is -106 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522724765 and Nanoseconds:490892333
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522724765 and Nanoseconds:573099877
circles.size = 0
Hough Eclipse Jitter obtained is -27 ms

Timestamp when Canny capture started: Seconds:1522724765 and Nanoseconds:573212543
Timestamp when CANNY capture stopped Seconds:1522724765 and Nanoseconds:641249179
Canny Jitter obtained is -18 nanoseconds

Timestamp when HOUGH capture started: Seconds:1522724765 and Nanoseconds:641293179
Timestamp when HOUGH captures stopped Seconds:1522724765 and Nanoseconds:811914058
Hough Jitter obtained is -80 nanoseconds

Timestamp when HOUGH ELIPTICAL capture started: Seconds:1522724765 and Nanoseconds:812024391
Timestamp when HOUGH ELIPTICAL capture stopped Seconds:1522724765 and Nanoseconds:900537310
circles.size = 0
Hough Eclipse Jitter obtained is -33 ms

```

640 x 480

### Final predictable response jitter analysis:

Resolution	Canny	Hough	Hough Elliptical
640 x 480	-20	-150	-35
320 x 240	-15	-20	-8
160 x 120	-17	-14	3

Average Jitter (ms)

Jitter is calculated by taking the difference between the deadline and the frame rate. Positive values mean that the deadline was met and negative values mean that it was missed.

### References:

- [http://mercury.pr.erau.edu/~siewerts/cs415/code/computer\\_vision\\_cv3\\_tested/](http://mercury.pr.erau.edu/~siewerts/cs415/code/computer_vision_cv3_tested/)
- <http://lwn.net/Articles/203924/>
- [Canny Edge Detector](#)
- [Hough Line Transform](#)