Input Format: RGB24

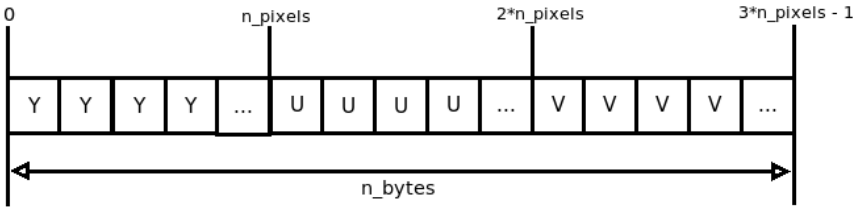
- Each pixel is represented by 24 bits (1 byte color component)

Read Byte								Green Byte								Blue Byte							
0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7

- For a 640x480 image
  - Total pixels: n_pixels = 640 * 480 = 307,2 kB
  - Total bytes: n_bytes = 640 * 480 * 3 = 921,6 kB
- Source image obtained using the following bash command:
  - *raspiyuv -w 640 -h 480 -bgr -o rgb_image.bgr*

Output Format: YUV

- YUV is a color encoding system based on a color space composed on three components.
  - Y: “Brightness” component, known as “luma”.
  - U: Chrominance (color) component.
  - V: Chrominance (color) component.
- YUV 4:4:4
  - Y, U and V components foreach pixel

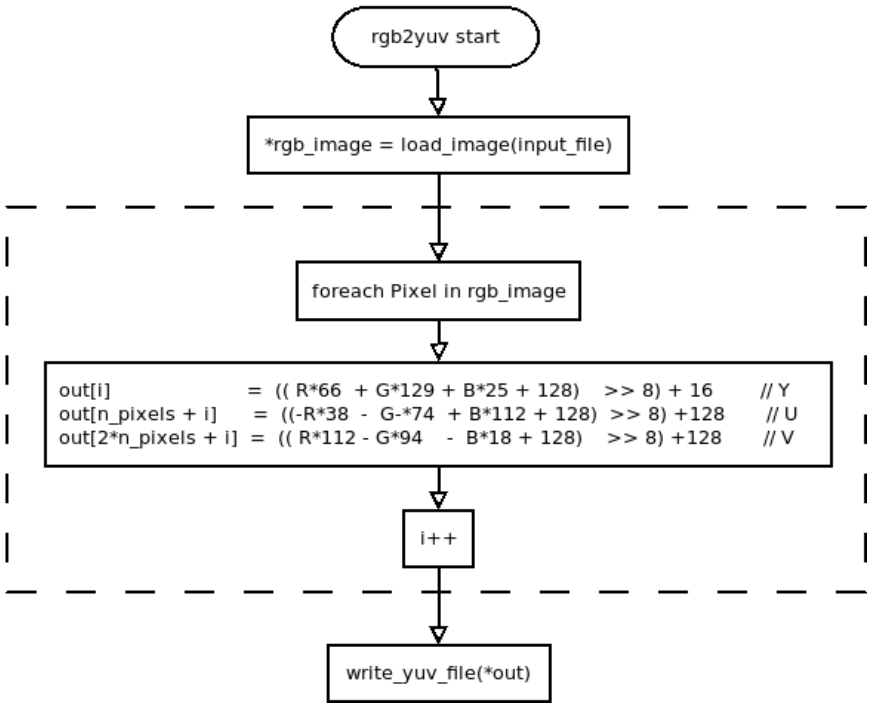


Conversion Algorithm

- **RGB to YUV conversion is done through the following equation:**

Y = ((66 * R + 129 * G + 25 * B + 128) >> 8) + 16
U = ((-38 * R - 74 * G + 112 * B + 128) >> 8) + 128 (Eq.1)
V = ((112 * R - 94 * G - 18 * B + 128) >> 8) + 128

Equation 1 is used on each pixel to compute the YUV components for the output image. The following flow chart illustrates the conversion process.



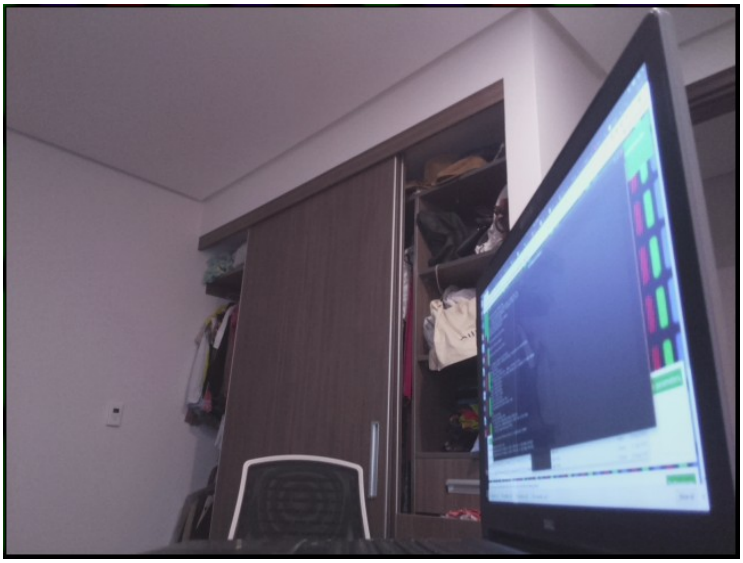
Results

- Sample Input Image (rgb_image.bgr)

- Image parameter (used to visualize image in http://rawpixels.net/)

Raw Data	rgb_image.bgr
Width	640
Height	480
Predefined Format	RGB24
Pixel Format	BGRA
Alpha First	No
Pixel Plane	Packed

- Sample Output Image (out.yuv)



- Image parameter (used to visualize image in http://rawpixels.net/)

Raw Data	out.yuv
Width	640
Height	480
Predefined Format	YUV444p
Pixel Format	YUV
Alpha First	No
Pixel Plane	Planar

- Execution time: Following table shows the average time spent in five iterations executing the conversion of rgb_image.bgr into out.yuv.

Iteration	Time (s)	Average
0	0.129493	0.0998358
1	0.090032	
2	0.089594	
3	0.091745	
4	0.098315	

-