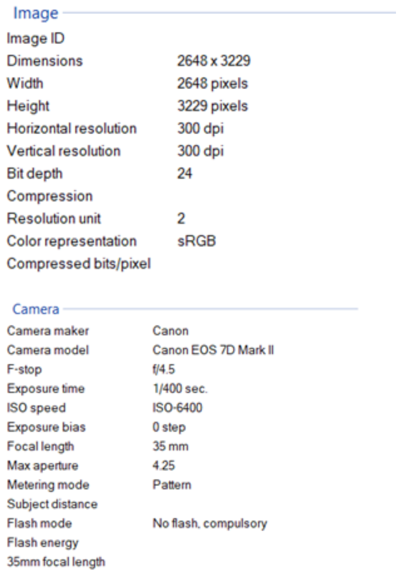
Rene L. Principe Jr.	Applied Physics 186	Dr. Maricor Soriano
2015-04622	***ACTIVITY 3 – IMAGE TYPES AND FORMATS***	

The image below is a Truecolor image that I shall be using in this activity. Listed on the right side are the Image properties. Luckily, the camera details are specified as well.



Image

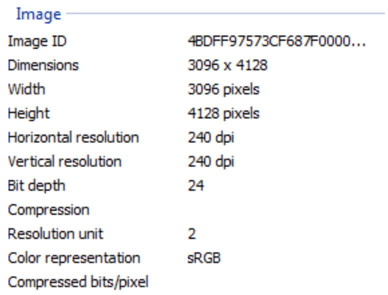
Image ID	
Dimensions	2648 x 3229
Width	2648 pixels
Height	3229 pixels
Horizontal resolution	300 dpi
Vertical resolution	300 dpi
Bit depth	24
Compression	
Resolution unit	2
Color representation	sRGB
Compressed bits/pixel	

Camera

Camera maker	Canon
Camera model	Canon EOS 7D Mark II
F-stop	f/4.5
Exposure time	1/400 sec.
ISO speed	ISO-6400
Exposure bias	0 step
Focal length	35 mm
Max aperture	4.25
Metering mode	Pattern
Subject distance	
Flash mode	No flash, compulsory
Flash energy	
35mm focal length	

OTHER SAMPLES

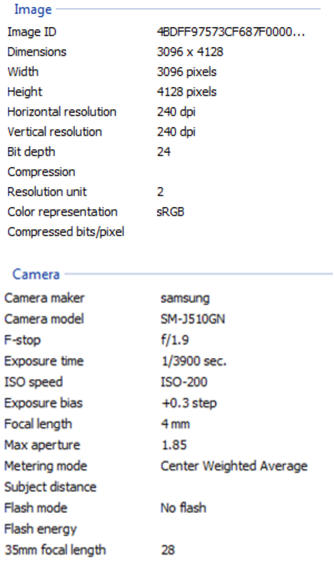
Phone Camera [RAW]



Image

Image ID	4BDFF97573CF687F0000...
Dimensions	3096 x 4128
Width	3096 pixels
Height	4128 pixels
Horizontal resolution	240 dpi
Vertical resolution	240 dpi
Bit depth	24
Compression	
Resolution unit	2
Color representation	sRGB
Compressed bits/pixel	

Phone Camera [Postproccesed]



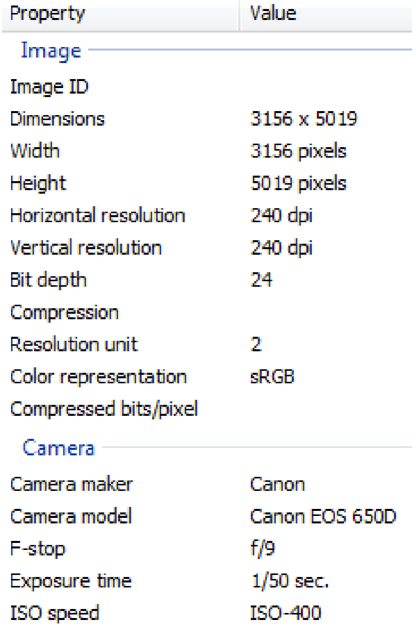
Image

Image ID	4BDFF97573CF687F0000...
Dimensions	3096 x 4128
Width	3096 pixels
Height	4128 pixels
Horizontal resolution	240 dpi
Vertical resolution	240 dpi
Bit depth	24
Compression	
Resolution unit	2
Color representation	sRGB
Compressed bits/pixel	

Camera

Camera maker	samsung
Camera model	SM-J510GN
F-stop	f/1.9
Exposure time	1/3900 sec.
ISO speed	ISO-200
Exposure bias	+0.3 step
Focal length	4 mm
Max aperture	1.85
Metering mode	Center Weighted Average
Subject distance	
Flash mode	No flash
Flash energy	
35mm focal length	28

DSLR [Postproccesed]



Property	Value
Image	
Image ID	
Dimensions	3156 x 5019
Width	3156 pixels
Height	5019 pixels
Horizontal resolution	240 dpi
Vertical resolution	240 dpi
Bit depth	24
Compression	
Resolution unit	2
Color representation	sRGB
Compressed bits/pixel	
Camera	
Camera maker	Canon
Camera model	Canon EOS 650D
F-stop	f/9
Exposure time	1/50 sec.
ISO speed	ISO-400

Using GIMP, **properties of the image** can be checked as shown in the screenshot

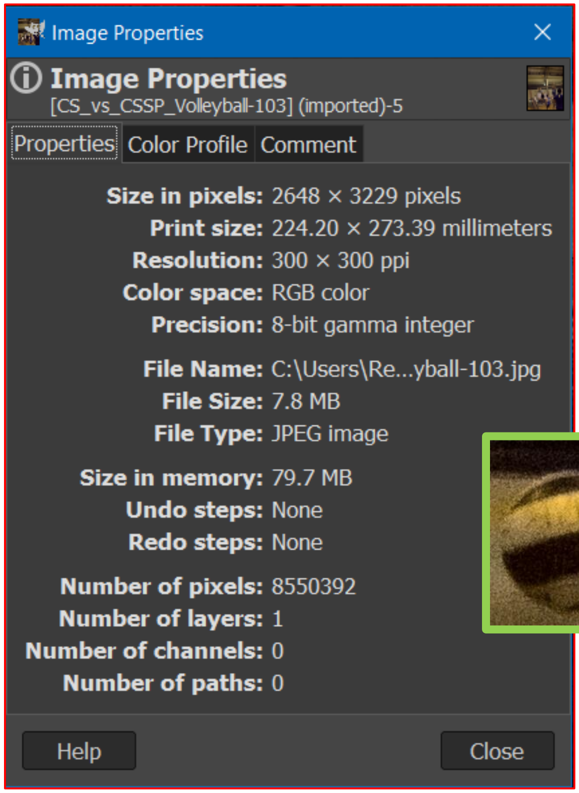


Image Properties

Image Properties

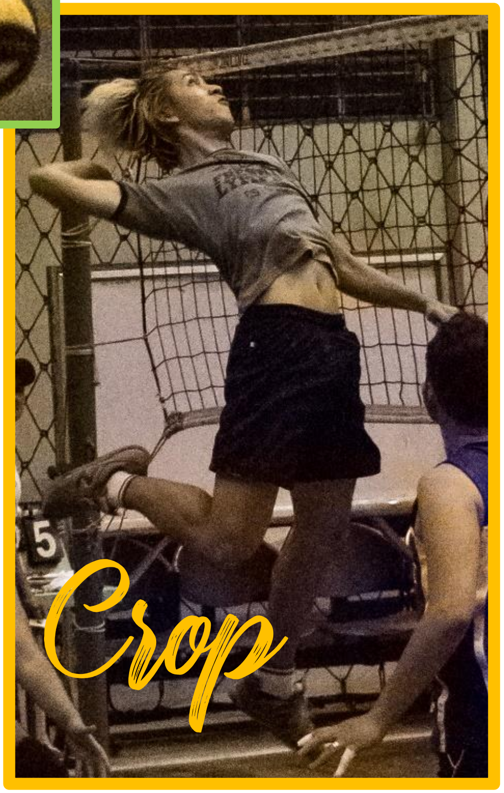
[CS_vs_CSSP_Volleyball-103] (imported)-5

Size in pixels: 2648 × 3229 pixels
Print size: 224.20 × 273.39 millimeters
Resolution: 300 × 300 ppi
Color space: RGB color
Precision: 8-bit gamma integer

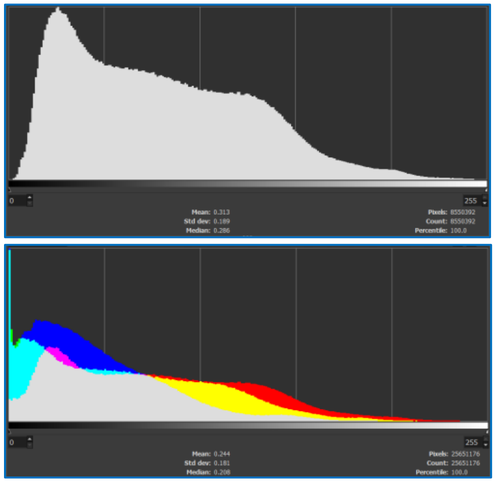
File Name: C:\Users\Re...yball-103.jpg
File Size: 7.8 MB
File Type: JPEG image

Size in memory: 79.7 MB
Undo steps: None
Redo steps: None

Number of pixels: 8550392
Number of layers: 1
Number of channels: 0
Number of paths: 0



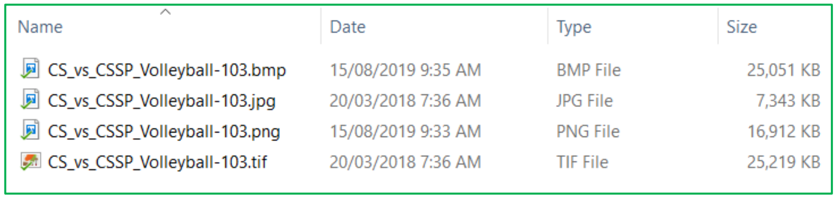
HISTOGRAM



Mean: 0.313
Std dev: 0.189
Median: 0.286
Pixels: 8550392
Count: 8550392
Percentile: 100.0

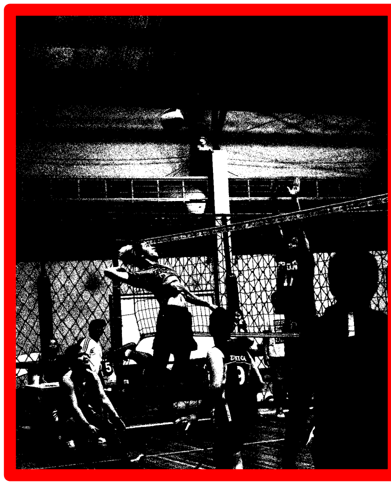
Mean: 0.244
Std dev: 0.181
Median: 0.208
Pixels: 25651176
Count: 25651176
Percentile: 100.0

Histogram depicts a more occurrence of darker pixels as the distribution is shifted to the left.



Name	Date	Type	Size
CS_vs_CSSP_Volleyball-103.bmp	15/08/2019 9:35 AM	BMP File	25,051 KB
CS_vs_CSSP_Volleyball-103.jpg	20/03/2018 7:36 AM	JPG File	7,343 KB
CS_vs_CSSP_Volleyball-103.png	15/08/2019 9:33 AM	PNG File	16,912 KB
CS_vs_CSSP_Volleyball-103.tif	20/03/2018 7:36 AM	TIF File	25,219 KB

The image file was then exported on to different file types and there is an evident size difference for each file type. The reasons behind size compression, etc. shall be explained in the latter portions of this report.



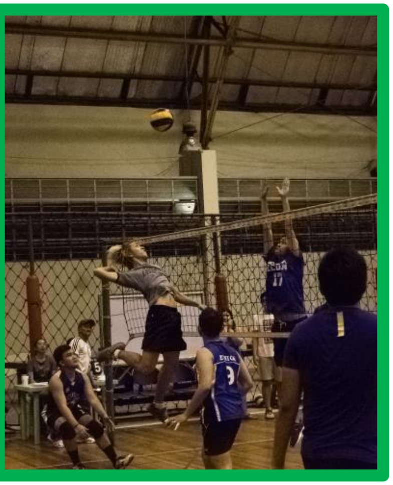
BINARY IMAGE

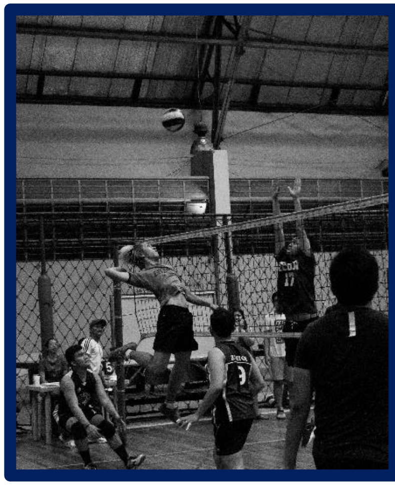
Size: 301 KB

Given a certain threshold between 0-256, all pixel values above the threshold is white (1) and the rest is black (0). Here, threshold is 128.

INDEXED IMAGE

Size: 5048 KB





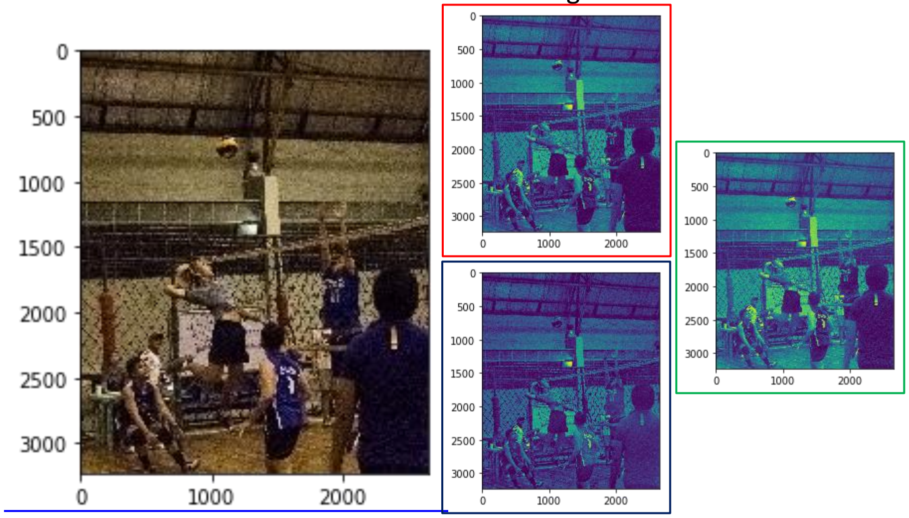
GRAYSCALE IMAGE

Size: 5745 KB

The gray pixel value is computed using the equation 0.21 Red + 0.72 Green + 0.07 Blue.

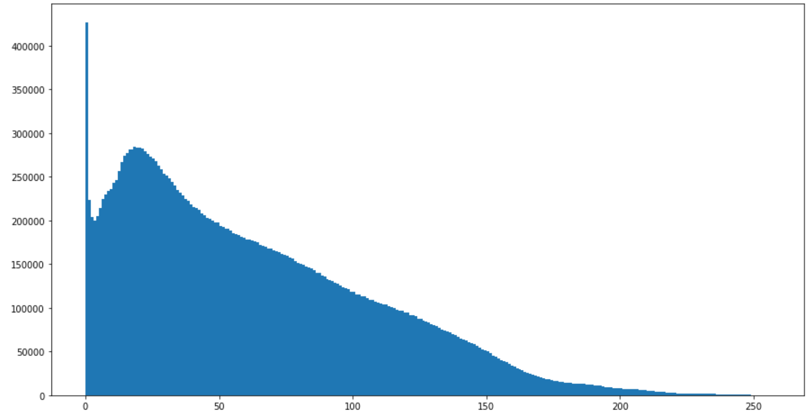
https://www.johndcook.com/blog/2009/08/24/algorithms-convert-color-grayscale/

Now, we explore various commands in Python to manipulate an image. In order to load an image, I used matplotlib.pyplot's "**imread**". According to its documentation, it "Reads an image from a file into an array". It treats an image as if it is a matrix of values, thus, operations would be much easier. To display the image I loaded, I used the command "**imshow**". As shown below, the image coordinate has its origin on the top left portion. To check the dimensions, "**numpy.shape**" command shows that the image that I loaded has shape **(3229, 2648, 3)**. The third dimension represent the channels since what I loaded is a true color image.



To write the image, "**savefig**" command is sufficient if one wishes to specify transparency, format, dpi, etc.

Image histogram can be viewed by implementing the "plt.hist2d()" command.



To extract the image data, I found a code online that outputs necessary information.

```
from PIL.ExifTags import TAGS
def get_exif(fn):
    ret = {}
    i = Image.open(fn)
    info = i._getexif()
    for tag, value in info.items():
        decoded = TAGS.get(tag,tag)
        ret[decoded] = value
    return ret
```

OUTPUT:

```
{'ResolutionUnit': 2,
 'ExifOffset': 254,
 'Make': 'Canon',
 'Model': 'Canon EOS 7D Mark II',
 'Software': 'Adobe Photoshop Lightroom 6.5.1 (Windows)',
 'DateTime': '2018:03:25 17:22:15',
 'Artist': 'MikeGo',
 'Copyright': '2018',
 'XResolution': (300, 1),
 'YResolution': (300, 1),
 'ExifVersion': b'0230',
 'ShutterSpeedValue': (8643856, 1000000),
 'ApertureValue': (433985, 100000),
 'DateTimeOriginal': '2018:03:20 07:36:04',
 'DateTimeDigitized': '2018:03:20 07:36:04',
 'ExposureBiasValue': (0, 1),
 'MaxApertureValue': (425, 100),
 'MeteringMode': 5,
 'ColorSpace': 1,
 'Flash': 16,
 'FocalLength': (35, 1),
 'ExposureMode': 1,
 'WhiteBalance': 1,
 'SceneCaptureType': 0,
 'FocalPlaneXResolution': (79937569, 32768),
 'FocalPlaneYResolution': (79937569, 32768),
 'FocalPlaneResolutionUnit': 3,
 'SubsecTimeOriginal': '48',
 'SubsecTimeDigitized': '48',
 'ExposureTime': (1, 400),
 'FNumber': (45, 10),
 'ExposureProgram': 1,
 'CustomRendered': 0,
 'ISOSpeedRatings': 6400,
 34864: 2,
 'BodySerialNumber': '048021004641',
 34866: 6400,
 'LensSpecification': ((18, 1), (135, 1), (0, 0), (0, 0)),
 'LensModel': 'EF-S18-135mm f/3.5-5.6 IS STM',
 'LensSerialNumber': '00002a80de'}
```

from: https://stackoverflow.com/questions/4764932/in-python-how-do-i-read-the-exif-data-for-an-image