

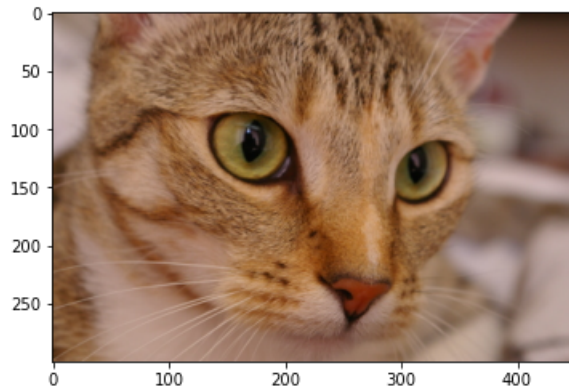
Session 3- image processing Basics with Skimage

```
In [4]: # importing libraries
import skimage
from skimage import io
from skimage import data
import matplotlib.pyplot as plt
%matplotlib inline
```

```
In [2]: cat= data.chelsea() # import an image from data
```

```
In [3]: plt.imshow(cat)
```

```
Out[3]: <matplotlib.image.AxesImage at 0x1f1b1f47a60>
```

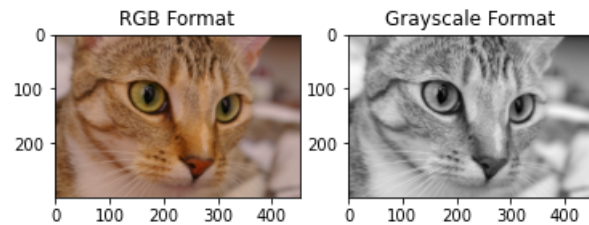


```
In [4]: # changing the image format Reading RGB image and converting into grayscale image
from skimage import color
img_new = color.rgb2gray(cat)

plt.subplot(121)
plt.imshow(cat)
plt.title('RGB Format')

plt.subplot(122)
plt.imshow(img_new, 'gray')
plt.title('Grayscale Format')
```

Out[4]: Text(0.5, 1.0, 'Grayscale Format')

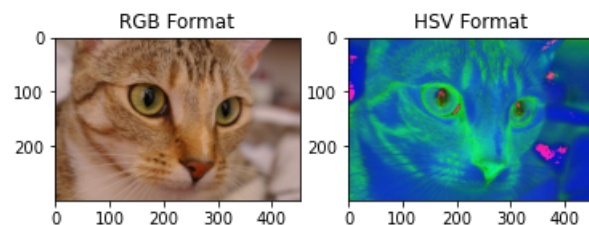


```
In [5]: # changing the image format Reading RGB image and converting into HSV (brightness) or HSL (Hue, Saturation & Lightness)
from skimage import color
img_new = color.rgb2hsv(cat)

plt.subplot(121)
plt.imshow(cat)
plt.title('RGB Format')

plt.subplot(122)
plt.imshow(img_new, 'gray')
plt.title('HSV Format')
```

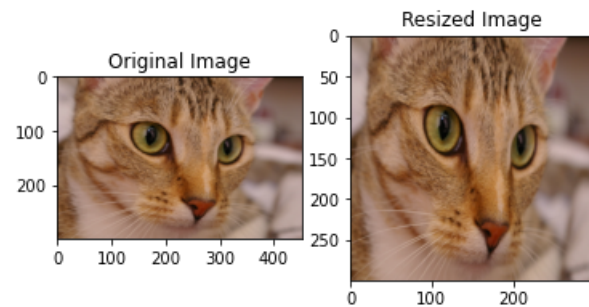
Out[5]: Text(0.5, 1.0, 'HSV Format')



```
In [6]: #Resizing Images
from skimage.transform import resize
img_resized = resize(cat, (300, 300))

#plot images
plt.subplot(121)
plt.imshow(cat)
plt.title('Original Image')
plt.subplot(122),
plt.imshow(img_resized)
plt.title('Resized Image')
```

Out[6]: Text(0.5, 1.0, 'Resized Image')



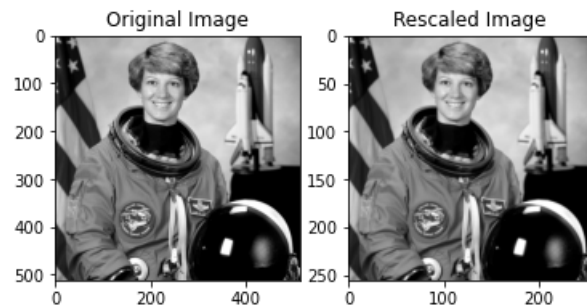
```
In [19]: from skimage.transform import rescale
from skimage import color

astro = color.rgb2gray(data.astronaut())

img_rescaled = rescale(astro, 0.50, anti_aliasing= True)
plt.subplot(121)
plt.imshow(astro, cmap='gray')
plt.title('Original Image')

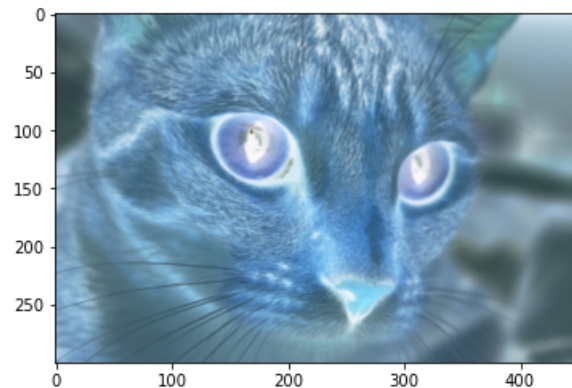
plt.subplot(122)
plt.imshow(img_rescaled, cmap='gray' )
plt.title('Rescaled Image')
```

Out[19]: Text(0.5, 1.0, 'Rescaled Image')



```
In [8]: from skimage import util          # Negative of the image
cats = data.chelsea()
inverted_cat = util.invert(cats)
plt.imshow(inverted_cat)
```

Out[8]: <matplotlib.image.AxesImage at 0x1f1b545af10>



```
In [9]: inverted_cat[0,:10,0]
```

```
Out[9]: array([112, 112, 114, 114, 114, 114, 114, 112, 111, 110], dtype=uint8)
```

```
In [10]: cats[0,:10,0]
```

```
Out[10]: array([143, 143, 141, 141, 141, 141, 141, 143, 144, 145], dtype=uint8)
```

```
In [11]: cats.min()
```

```
Out[11]: 0
```

```
In [12]: cats.max()
```

```
Out[12]: 231
```

```
In [13]: # histogram
img5= data.camera()
img5.shape
```

```
Out[13]: (512, 512)
```

```
In [14]: img5.ravel()      # converts the matrix into one dimensional array
```

```
Out[14]: array([156, 157, 160, ..., 121, 113, 111], dtype=uint8)
```

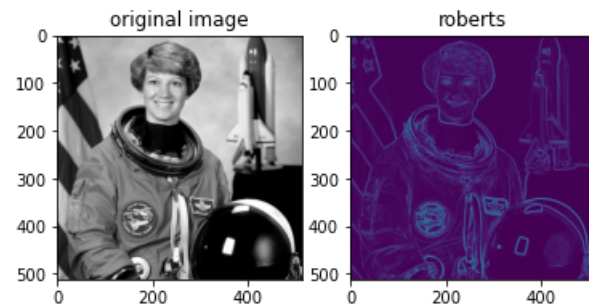
```
In [15]: plt.hist(img5.ravel(), bins =256, histtype = 'step', color = 'blue')
```

```
Out[15]: (array([ 102.,  76.,  89., 114., 159., 209., 335., 1173., 3523.,
5129., 4490., 4980., 5762., 6212., 6067., 4480., 2805., 1375.,
860., 625., 498., 503., 426., 430., 379., 398., 418.,
402., 358., 362., 354., 349., 355., 381., 371., 379.,
348., 375., 390., 398., 394., 349., 327., 345., 311.,
263., 256., 279., 274., 258., 258., 245., 242., 261.,
265., 268., 285., 276., 334., 308., 337., 309., 313.,
370., 276., 301., 245., 230., 244., 242., 222., 217.,
230., 238., 235., 224., 205., 220., 213., 208., 164.,
203., 222., 216., 230., 243., 251., 281., 315., 328.,
334., 327., 333., 381., 417., 420., 473., 525., 573.,
620., 650., 687., 727., 760., 803., 816., 924., 986.,
981., 1002., 1085., 1070., 1174., 1202., 1219., 1190., 1314.,
1321., 1432., 1399., 1453., 1539., 1477., 1509., 1598., 1592.,
1691., 1659., 1745., 1705., 1734., 1702., 1663., 1598., 1655.,
1673., 1626., 1556., 1435., 1435., 1398., 1449., 1391., 1514.,
1590., 1800., 1964., 2096., 2176., 2147., 2225., 2163., 2185.,
2117., 2258., 2494., 2524., 2885., 3045., 3134., 3767., 4147.,
4596., 4906., 4790., 4497., 4195., 3993., 3737., 3415., 2906.,
3735., 3716., 3665., 3635., 3605., 3555., 3503., 3437., 3365.]
```

```
In [16]: #edge detection filters
from skimage.filters import roberts, sobel, scharr, prewitt
astro = color.rgb2gray(data.astronaut())

edge_roberts = roberts(astro)
edge_sobel = sobel(astro)
edge_scharr = scharr(astro)
edge_prewitt = prewitt(astro)
plt.subplot(121)
plt.imshow(astro, 'gray')
plt.title('original image')
plt.subplot(122)
plt.imshow(edge_roberts)
plt.title('roberts')
```

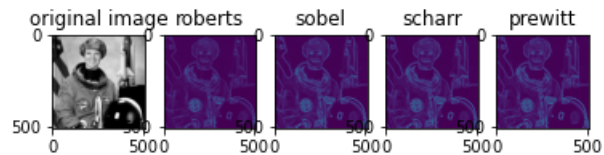
Out[16]: Text(0.5, 1.0, 'roberts')



```
In [17]: from skimage.filters import roberts, sobel, scharr, prewitt
astro = color.rgb2gray(data.astronaut())

edge_roberts = roberts(astro)
edge_sobel = sobel(astro)
edge_scharr = scharr(astro)
edge_prewitt = prewitt(astro)
plt.subplot(151)
plt.imshow(astro, 'gray')
plt.title('original image')
plt.subplot(152)
plt.imshow(edge_roberts)
plt.title('roberts')
plt.subplot(153)
plt.imshow(edge_sobel)
plt.title('sobel')
plt.subplot(154)
plt.imshow(edge_scharr)
plt.title('scharr')
plt.subplot(155)
plt.imshow(edge_prewitt)
plt.title('prewitt')
```

Out[17]: Text(0.5, 1.0, 'prewitt')

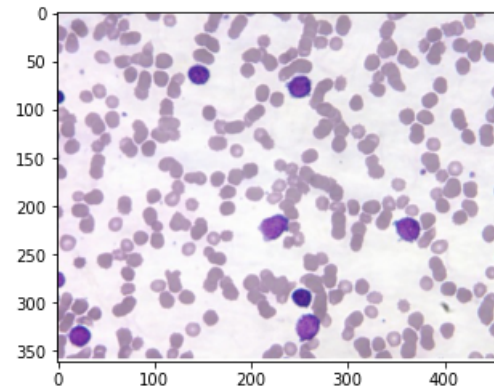


Segmentation

```
In [18]: cells = io.imread('cells.jpg')
```

```
In [19]: plt.imshow(cells)
```

```
Out[19]: <matplotlib.image.AxesImage at 0x1f1b5818f40>
```



```
In [20]: cells.mean()
```

```
Out[20]: 215.5730443092332
```

```
In [21]: cells.max()
```

```
Out[21]: 255
```

```
In [22]: cells.min()
```

```
Out[22]: 0
```

```
In [23]: mask = cells > 100  
cells[mask] = 255
```



```
In [24]: plt.imshow(cells, cmap="gray")
```

```
Out[24]: <matplotlib.image.AxesImage at 0x1f1b668bac0>
```

