Negative Images

Negative images are useful for enhancing white or grey detail embedded in dark regions

S = max_intensity - r

In [4]:

```
#import libraries
import numpy as np
from PIL import Image
import matplotlib.pyplot as plt
```

In [5]:

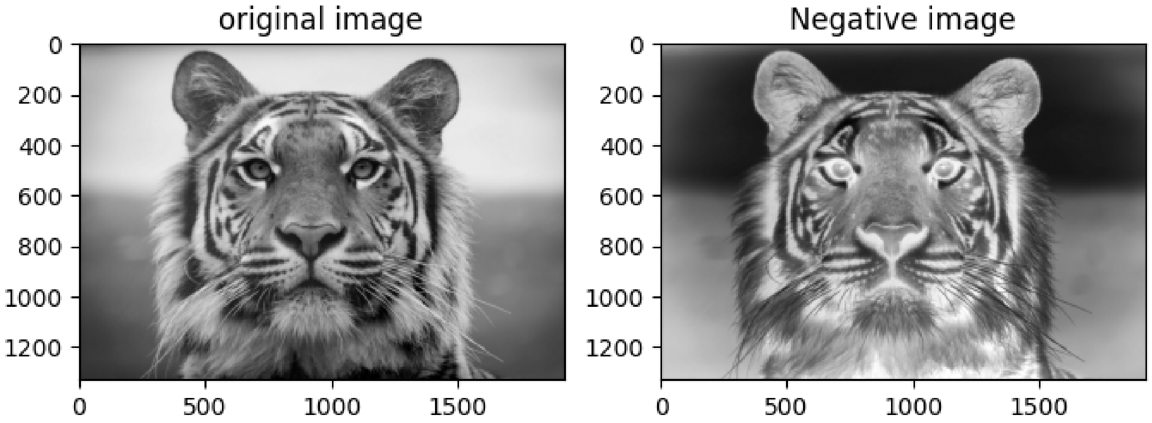
```
def Negative_Image(image):
  # image = image.resize((400,400), Image.Resampling.LANCZOS)
  # convert to numpy array
  numpy_image = np.array(image)

  row = numpy_image.shape[0]
  column = numpy_image.shape[1]

  new_array = np.zeros(shape=(row,column))

  for i in range(row):
    for j in range(column):
      new_array[i][j] = 255 - numpy_image[i][j]

  negative_image = Image.fromarray(new_array)
  negative_image = negative_image.convert("L")

  return negative_image
```

In [6]:

```
# reading image and converting to gray scale
image = Image.open('../images/tiger.jpg').convert('L')
# calling negative function
negative_image = Negative_Image(image)

#displaying the images
fig = plt.figure()
fig.set_figheight(8)
fig.set_figwidth(8)

fig.add_subplot(1,2,1)
plt.imshow(image, cmap='gray')
plt.title('original image')

fig.add_subplot(1,2,2)
plt.imshow(negative_image, cmap='gray')
plt.title('Negative image')
```

Out[6]:

```
Text(0.5, 1.0, 'Negative image')
```


original image
Negative image
0
200
400
600
800
1000
1200
0
500
1000
1500
0
200
400
600
800
1000
1200
0
500
1000
1500