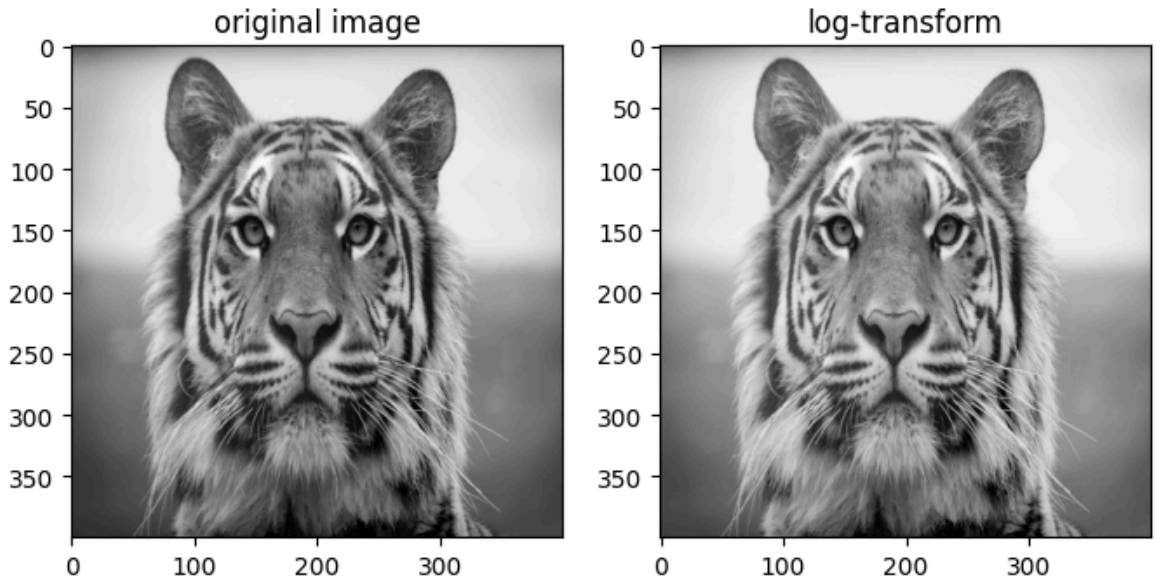
Log Transform

In [1]:
```
import numpy as np
from PIL import Image
import matplotlib.pyplot as plt
```

In [2]:
```
def log_transform(input_image):
  # resizing image
  img = input_image.resize((400,400), Image.Resampling.LANCZOS)
  # convert to numpy array
  numpy_image = np.array(img)
  numpy_image = numpy_image/255
  numpy_image = numpy_image + 1
  numpy_image = np.log(numpy_image)
  print(type(numpy_image))
  numpy_image = numpy_image * 255
  numpy_image = np.around(numpy_image,decimals=0)
  log_image = Image.fromarray(numpy_image)
  log_image = log_image.convert("L")

    #plotting input and output images
  # set up side-by-side image display
  fig = plt.figure()
  fig.set_figheight(6)
  fig.set_figwidth(8)

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

  fig.add_subplot(1,2,2)
  plt.imshow(log_image, cmap='gray')
  plt.title('log-transform')

  return log_image
```

In [3]:
```
# reading image and converting to gray scale
img = Image.open("../images/tiger.jpg").convert('L')
# display image
a = log_transform(img)
```

```
<class 'numpy.ndarray'>
```


original image
log-transform
0
50
100
150
200
250
300
350
0
100
200
300