

Image Compression using Python

```
import cv2
import numpy as np
import matplotlib.pyplot as plt

img=cv2.imread("D:\Image Analysis\Penguin.jpg",0)

print(img.shape)
print(img.size)
cv2.imshow("original img",img)
cv2.waitKey(0)
```

Output :

```
(226L, 290L)
65540
```



Convert image in to a matrix format

```
imgmat = np.array(img,float)
print(imgmat.shape)
print(imgmat.size)

imgmat = np.matrix(imgmat)
print(imgmat)
```

Output :

```
(226L, 290L)
65540
[[ 162.  163.  164. ..., 155.  154.  154.]
 [ 163.  164.  164. ..., 156.  155.  155.]
```

```
[ 165.  165.  165. ...,  157.  156.  156.]
...,
[ 114.  114.  115. ...,  102.  115.  109.]
[ 134.  126.  109. ...,  110.  125.  109.]
[ 124.  134.  130. ...,  124.  122.  106.]]
```

compute Singular value decomposition

```
U, sigma, V = np.linalg.svd(imgmat)
```

```
reconstimg = np.matrix(U[:, :1]) * np.diag(sigma[:1]) * np.matrix(V[:1, :])
plt.imshow(reconstimg, cmap='gray')
plt.show()
```

Reconstructing image using different number of vectors

```
for i in xrange(5,31,5):
    reconstimg = np.matrix(U[:, :i]) * np.diag(sigma[:i]) * np.matrix(V[:i, :])
    plt.imshow(reconstimg, cmap='gray')
    title = "n = %s" % i
    plt.title(title)
    plt.show()
```

Output :









