```python
import numpy as np
from matplotlib.image import imread
image_raw=imread('/content/sharingan.jpg')
image_raw
```

```
array([[[0, 0, 0],
        [0, 0, 0],
        [0, 0, 0],
        ...,
        [0, 0, 0],
        [0, 0, 0],
        [0, 0, 0]],

       [[0, 0, 0],
        [0, 0, 0],
        [0, 0, 0],
        ...,
        [0, 0, 0],
        [0, 0, 0],
        [0, 0, 0]],

       [[0, 0, 0],
        [0, 0, 0],
        [0, 0, 0],
        ...,
        [0, 0, 0],
        [0, 0, 0],
        [0, 0, 0]],

       ...,

       [[0, 0, 0],
        [0, 0, 0],
        [0, 0, 0],
        ...,
        [0, 0, 0],
        [0, 0, 0],
        [0, 0, 0]],

       [[0, 0, 0],
        [0, 0, 0],
        [0, 0, 0],
        ...,
        [0, 0, 0],
        [0, 0, 0],
        [0, 0, 0]],

       [[0, 0, 0],
        [0, 0, 0],
        [0, 0, 0],
        ...,
        [0, 0, 0],
        [0, 0, 0],
        [0, 0, 0]]], dtype=uint8)
```

```python
image_raw.shape
```

```
(1080, 1080, 3)
```

```python
import matplotlib.pyplot as plt
plt.imshow(image_raw)  ##original image
```

```
<matplotlib.image.AxesImage at 0x7fcc5c141a50>
```

```python
plt.imshow(image_raw)
plt.axis(False)
plt.show()
```



```python
i=image_raw.sum(axis=2) # print the two axes
i.shape
```

```
(1080, 1080)
```

```python
ib=i/i.max()
ib.max()
```

```
1.0
```
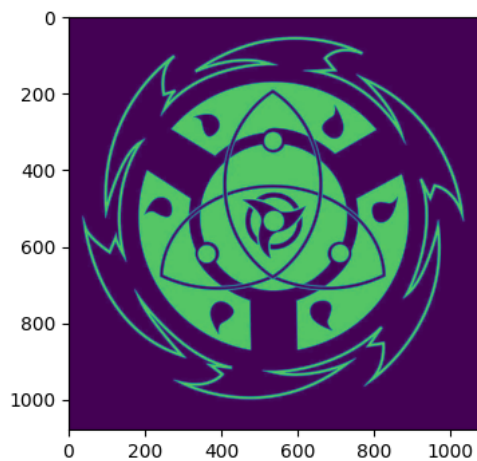
```python
plt.figure(figsize=(6,4))
plt.imshow(ib)
```

```
<matplotlib.image.AxesImage at 0x7fcc4c5ad000>
```



```python
plt.figure(figsize=(6,4))
plt.imshow(ib,cmap=plt.cm.gray) ##gray image
```

```
<matplotlib.image.AxesImage at 0x7fcc4c412e60>
```



```python
from sklearn.decomposition import PCA, IncrementalPCA
pca=PCA()
pca.fit(ib)
var=np.cumsum(pca.explained_variance_ratio_)*100
k=np.argmax(var>95)
print("elements are "+str(k))
```
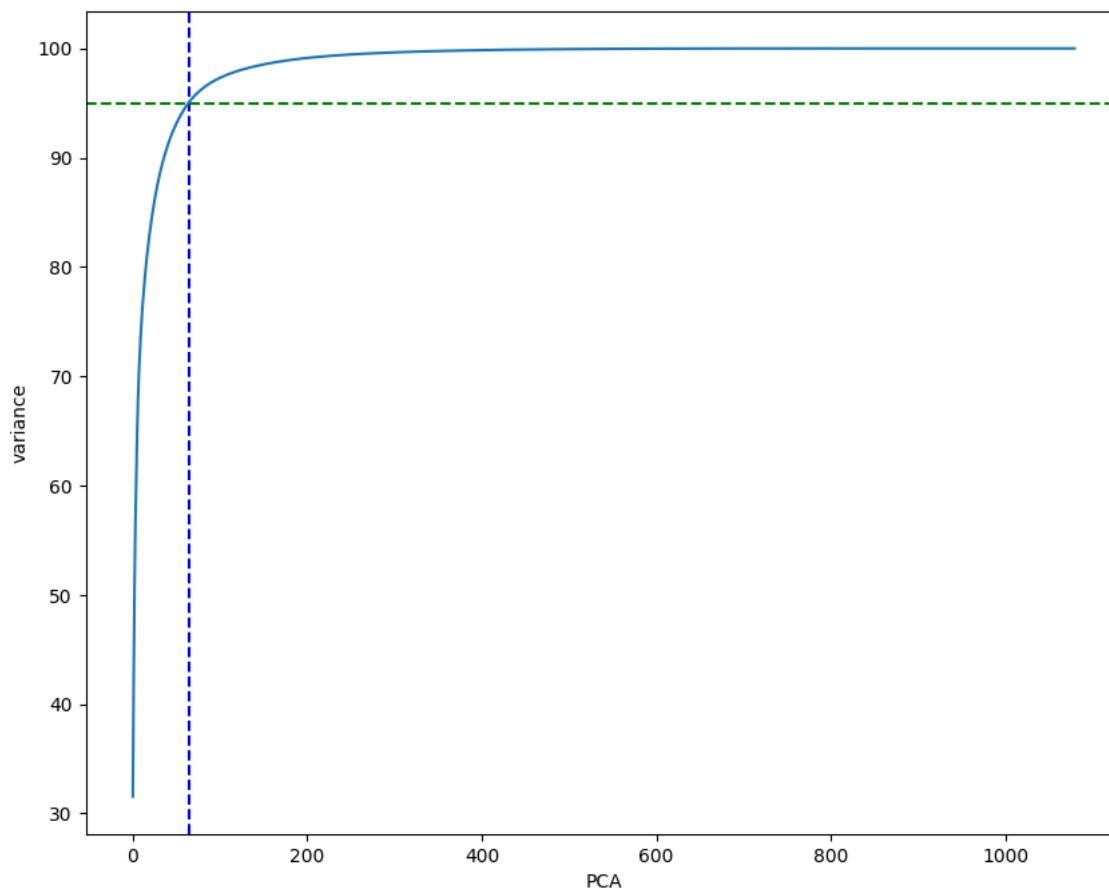
```
elements are 64
```



```python
plt.figure(figsize=(10,8))
plt.ylabel('variance')
plt.xlabel('PCA')
plt.axvline(x=k,color='b',linestyle='--')
plt.axhline(y=95,color='g',linestyle='--')
ax=plt.plot(var)
```



```python
ipca=IncrementalPCA(n_components=2)
i=ipca.inverse_transform(ipca.fit_transform(ib))
plt.figure(figsize=(6,4))
plt.imshow(i)
```

```
<matplotlib.image.AxesImage at 0x7fcc1b55a590>
```



```
plt.imshow(i,cmap=plt.cm.gray)
```
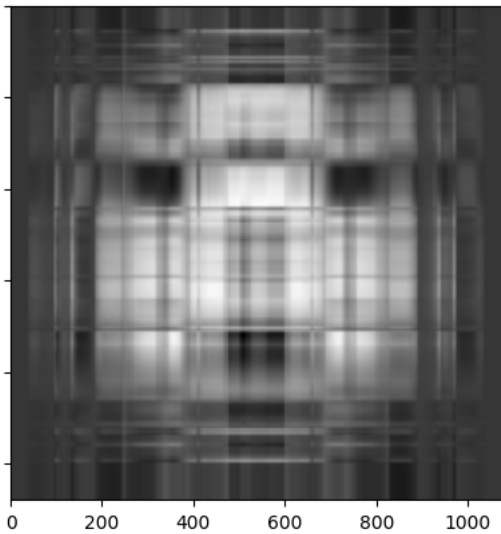
```
<matplotlib.image.AxesImage at 0x7fcc1b5d3730>
```



```
from mpl_toolkits.mplot3d import Axes3D
import cv2
o_img=cv2.imread('/content/sharingan.jpg')
img=cv2.cvtColor(o_img,cv2.COLOR_BGR2RGB)
v=img.reshape(-1,3)
plt.imshow(o_img)
```

```
<matplotlib.image.AxesImage at 0x7fcc0ec68130>
```
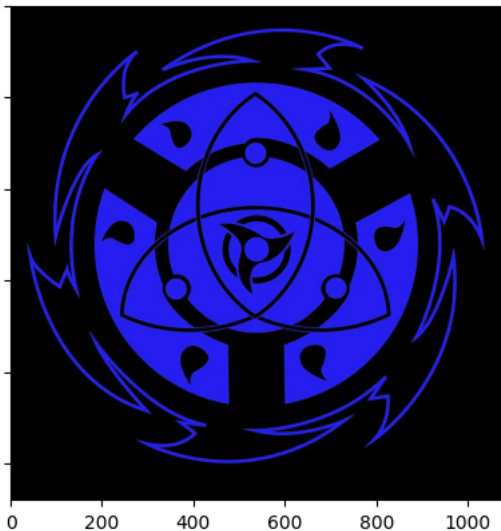


```
plt.imshow(img)
```
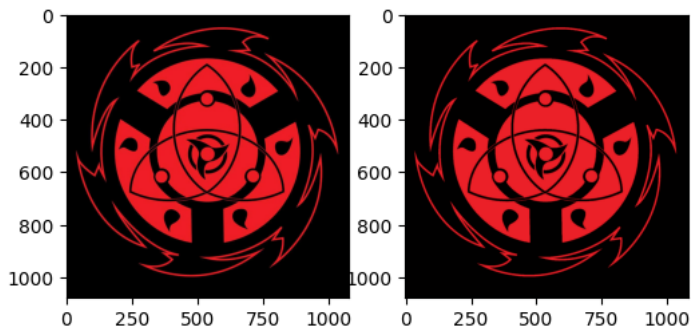
```
<matplotlib.image.AxesImage at 0x7fcc0ecdd180>
```



```python
import numpy as np
v=np.float32(v)
criteria=(cv2.TERM_CRITERIA_EPS+cv2.TERM_CRITERIA_MAX_ITER,10,1.0)
k=3
attempts=10
ret,label,center=cv2.kmeans(v,k,None,criteria,attempts,cv2.KMEANS_PP_CENTERS)
center=np.uint8(center)
res=center[label.flatten()]
r=res.reshape(img.shape)
f1=6
f2=4
plt.figure(figsize=(f1,f2))
plt.subplot(1,2,1),plt.imshow(img)
plt.subplot(1,2,2),plt.imshow(r)
```

```
(<Axes: >, <matplotlib.image.AxesImage at 0x7fcc0e39a140>)
```



```python
gray_image=cv2.cvtColor(image_raw,cv2.COLOR_BGR2GRAY) ##convert to grey image
plt.imshow(gray_image)
```

```
<matplotlib.image.AxesImage at 0x7fcc0e231b70>
```
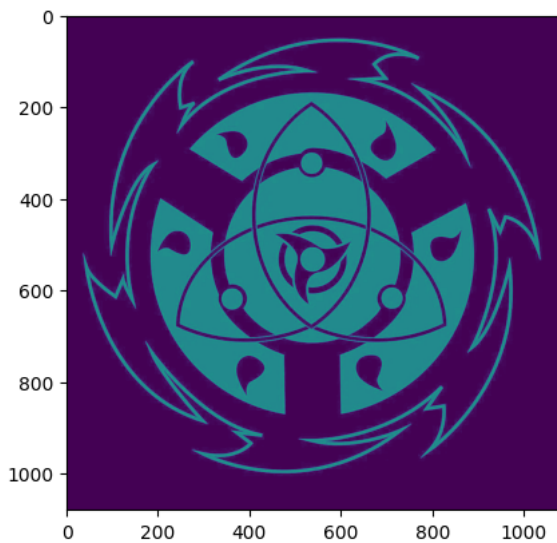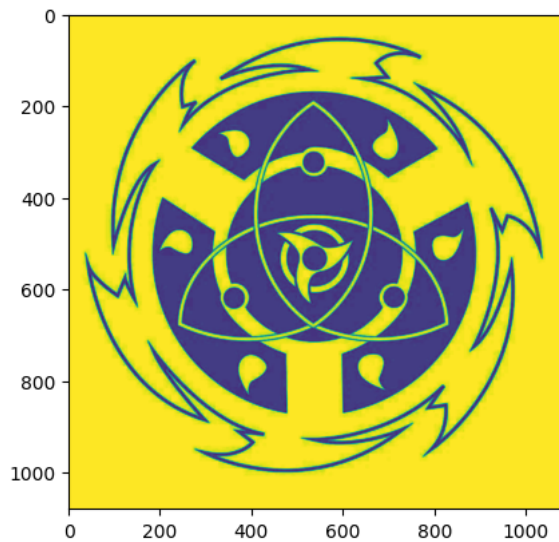


```python
inverted_image=255-gray_image ## converting to invert image
plt.imshow(inverted_image)
```

<matplotlib.image.AxesImage at 0x7fcc0e2a4400>


```
## gaussian Blur image
gblur_image1=cv2.GaussianBlur(inverted_image,(5,5),0) ##kernel size
plt.imshow(gblur_image1)
```
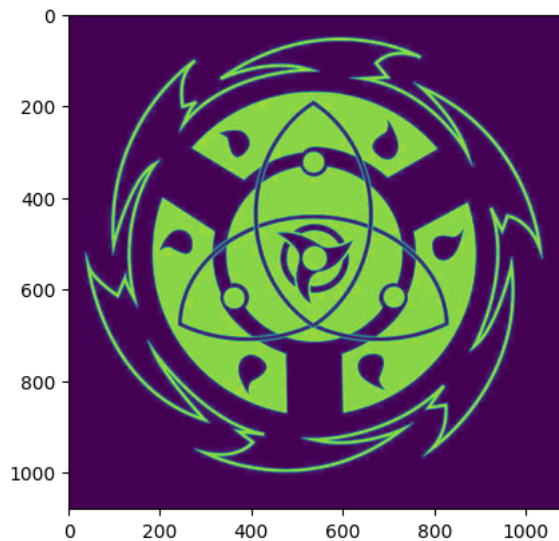
<matplotlib.image.AxesImage at 0x7fcc0e2a4280>


```
## inverted gaussian blur image
invgblur_image=255-gblur_image1
plt.imshow(invgblur_image)
```

<matplotlib.image.AxesImage at 0x7fcc0e179780>


```
blur_image=255-inverted_image
p_image=255-blur_image
pencil_sketch1=cv2.divide(gray_image,p_image,scale=256.0)
plt.imshow(pencil_sketch1)
```

<matplotlib.image.AxesImage at 0x7fcc0dff6890>


```
sketch_img=cv2.divide(gray_image,invgblur_image,scale=256.0)
plt.axis(False)
plt.imshow(sketch_img)
```

<matplotlib.image.AxesImage at 0x7fcc0e056410>


```
#original image vs Pencil sketch image comparison
plt.subplot(1,2,1)
plt.title('This is Original image',size=10)
plt.imshow(image_raw)
plt.axis('off')
plt.subplot(1,2,2)
plt.title('This is pencil sketch',size=10)
plt.imshow(sketch_img)
plt.axis('off')
plt.show()
```
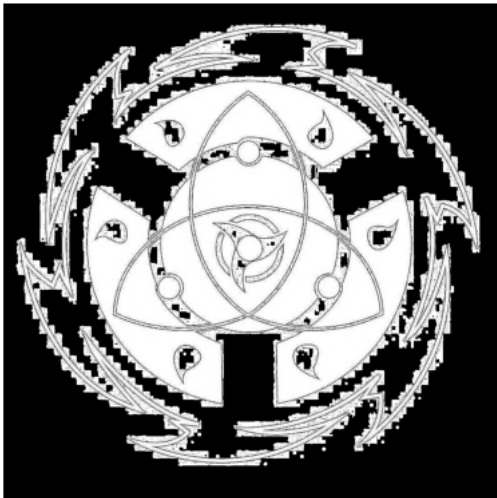


```
plt.title('This is rgb sketch image',size=10)
sketch=cv2.cvtColor(sketch_img,cv2.COLOR_BGR2RGB)
plt.imshow(sketch)
plt.axis('off')
plt.show()
```

This is rgb sketch image

✓ 0s   completed at 10:35   ● ✕