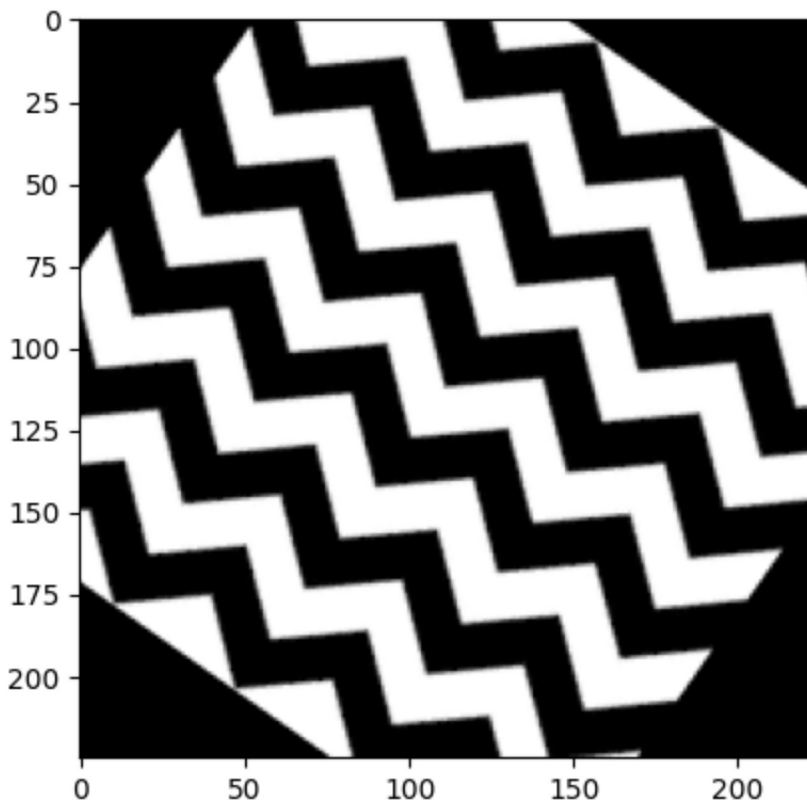


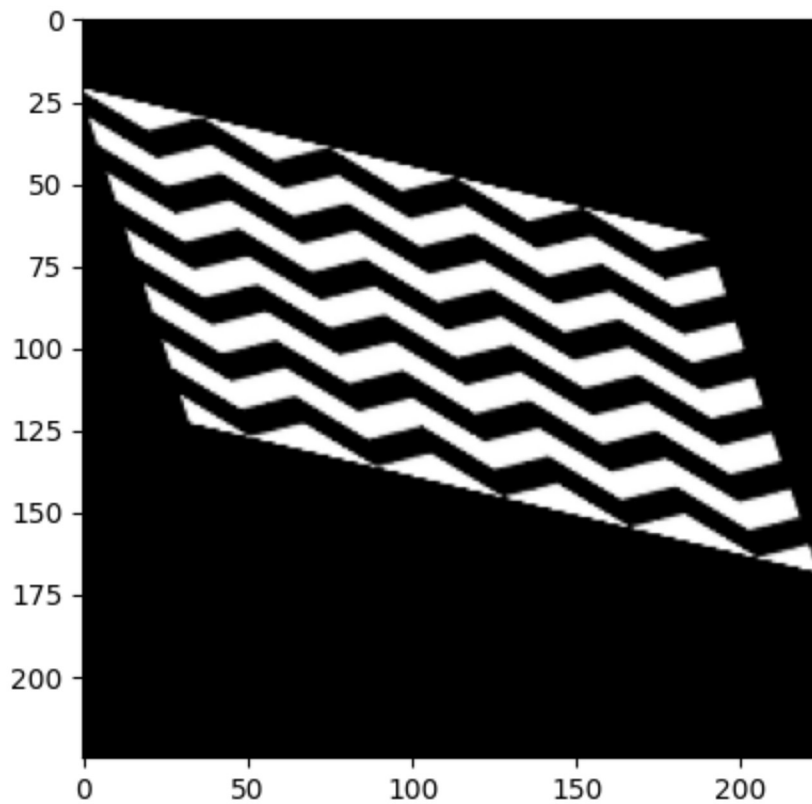
```
In [ ]: import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import cv2
```

```
In [ ]: img = cv2.imread('./image/zi.png', cv2.IMREAD_GRAYSCALE)
rows, cols = img.shape
M = cv2.getRotationMatrix2D(((cols-1)/2.0, (rows-1)/2.0), 145, 1)
dst = cv2.warpAffine(img, M, (cols, rows))
plt.imshow(dst, cmap='gray')
```

Out[]: <matplotlib.image.AxesImage at 0x1fcc9ced690>



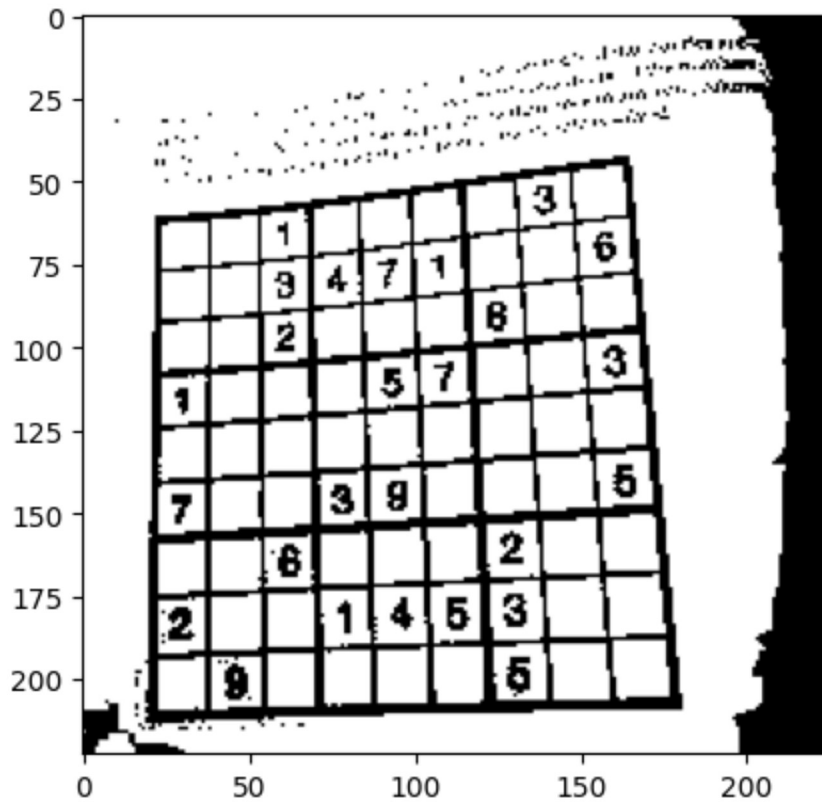
```
In [ ]: img = cv2.imread('./image/zi.png')
srcTri = np.array([[0, 0], [img.shape[1] - 1, 0],
                  [0, img.shape[0] - 1]]).astype(np.float32)
dstTri = np.array([[0, img.shape[1]*0.25], [img.shape[1]*0.85, img.shape[0]
                  * 0.45], [img.shape[1]*0.15, img.shape[0]*0.7]]).astype(np.float32)
warp_mat = cv2.getAffineTransform(srcTri, dstTri)
warp_dst = cv2.warpAffine(img, warp_mat, (img.shape[1], img.shape[0]))
rows, cols = warp_dst.shape[:2]
M = cv2.getRotationMatrix2D(((cols-1)/2.0, (rows-1)/2.0), 180, 1)
dst = cv2.warpAffine(warp_dst, M, (cols, rows))
plt.imshow(dst, cmap='gray')
plt.show()
```



```
In [ ]: img = cv2.imread('./image/images.jpg', cv2.IMREAD_GRAYSCALE)
res ,th1 = cv2.threshold(img , 150 , 170 , cv2.THRESH_BINARY)
th2 = cv2.adaptiveThreshold(img,255,cv2.ADAPTIVE_THRESH_MEAN_C,\
                             cv2.THRESH_BINARY,11,2)
th3 = cv2.adaptiveThreshold(img,255,cv2.ADAPTIVE_THRESH_GAUSSIAN_C,\
                             cv2.THRESH_BINARY,11,2)

# plt.imshow(th1)
plt.imshow(th1 , cmap = 'gray')
```

```
Out[ ]: <matplotlib.image.AxesImage at 0x1fccc1bead0>
```



```
In [ ]: import cv2
import numpy as np
from matplotlib import pyplot as plt

img = cv2.imread('./image/Big_Tiger_Cub.jpg', cv2.IMREAD_GRAYSCALE)

mean_filter = np.ones((3,3)) / 9
x = cv2.getGaussianKernel(5,10)
gaussian = np.outer(x, x)

scharr = np.array([[-3, 0, 3],
                  [-10,0,10],
                  [-3, 0, 3]])
sobel_x = np.array([[-1, 0, 1],
                  [-2, 0, 2],
                  [-1, 0, 1]])
sobel_y = np.array([[-1,-2,-1],
                  [0, 0, 0],
                  [1, 2, 1]])
laplacian = np.array([[0, 1, 0],
                  [1,-4, 1],
                  [0, 1, 0]])
filters = [mean_filter, gaussian, laplacian, sobel_x, sobel_y, scharr]
filter_name = ['mean_filter', 'gaussian', 'laplacian', 'sobel_x', \
              'sobel_y', 'scharr']

filtered_images = [cv2.filter2D(img, -1, filt) for filt in filters]

for i in range(6):
    plt.subplot(2,3,i+1), plt.imshow(filtered_images[i], cmap='gray')
```

```
plt.title(filter_name[i]), plt.xticks([]), plt.yticks([])  
plt.show()
```

mean_filter



gaussian



laplacian



sobel_x



sobel_y



scharr

