

```
In [1]: import numpy as np
import cv2
from matplotlib import pyplot as plt
```

```
In [7]: img = 'data/LetterA.jpg'
```

Binarisation

```
In [8]: #Binarisation par niveau de gris
gray_img = cv2.imread(img, cv2.IMREAD_GRAYSCALE)
```

```
In [9]: # Matrice Image
gray_img
```

```
Out[9]: array([[255, 255, 255, 254, 254, 254, 255, 255, 249, 250, 255, 255, 249,
                248, 255, 250],
               [253, 250, 255, 252, 255, 255, 250, 219,  21,  13,  53, 230, 255,
                255, 250, 255],
               [254, 255, 254, 255, 255, 255, 255, 125,   0,   7,  12, 233, 251,
                255, 243, 255],
               [253, 255, 243, 255, 245, 253, 224,  20,  18,  53,  10, 214, 241,
                255, 254, 255],
               [250, 255, 241, 255, 255, 244,  95,   0, 179, 130,   1, 172, 255,
                255, 248, 253],
               [253, 255, 255, 245, 255, 201,   0,  84, 242, 163,   0,  88, 251,
                255, 255, 255],
               [255, 255, 255, 250, 240,  97,   8, 198, 249, 221,  37,  63, 247,
                255, 251, 242],
               [255, 255, 254, 255, 172,   0, 104, 255, 255, 240,  33,  21, 220,
                255, 253, 255],
               [248, 255, 249, 255,  24,  14, 199, 255, 253, 255,  72,  21, 167,
                238, 255, 255],
               [255, 254, 255, 143,   3, 145, 255, 250, 252, 248, 136,   0, 134,
                255, 255, 248],
               [251, 251, 215,  44,  27, 211, 255, 255, 253, 255, 195,  20,  85,
                248, 245, 255],
               [246, 255, 122,   0,   0,  13,   2,   0,   0,  13,   0,   0,  52,
                253, 251, 255],
               [255, 199,   0,   0,   6,   0,   7,   0,   0,   1,   2,   2,  16,
                223, 251, 253],
               [254,  68,  13, 191, 241, 250, 255, 252, 255, 241, 255,  94,   0,
                166, 248, 255],
               [195,   0,  76, 240, 255, 255, 249, 255, 255, 243, 255, 132,   1,
                134, 255, 251],
               [ 42,   0, 171, 251, 255, 255, 247, 255, 254, 255, 245, 202,   2,
                59, 242, 255],
               [  3,  58, 247, 255, 255, 254, 250, 255, 255, 248, 249, 235,   7,
                30, 231, 255],
               [167, 169, 255, 248, 247, 255, 255, 255, 248, 255, 255, 234,  57,
                68, 216, 255],
               [255, 246, 255, 255, 247, 255, 255, 247, 248, 255, 245, 255, 241,
                254, 255, 255],
               [252, 255, 253, 255, 249, 255, 255, 245, 250, 254, 241, 255, 255,
                255, 245, 248],
               [248, 255, 245, 255, 242, 253, 255, 255, 255, 255, 255, 255, 239,
                251, 245, 255],
               [255, 255, 244, 255, 248, 255, 255, 255, 246, 237, 248, 255, 255,
                255, 254, 244],
               [251, 253, 251, 255, 255, 255, 249, 255, 255, 255, 255, 255, 254,
                250, 255, 255]], dtype=uint8)
```

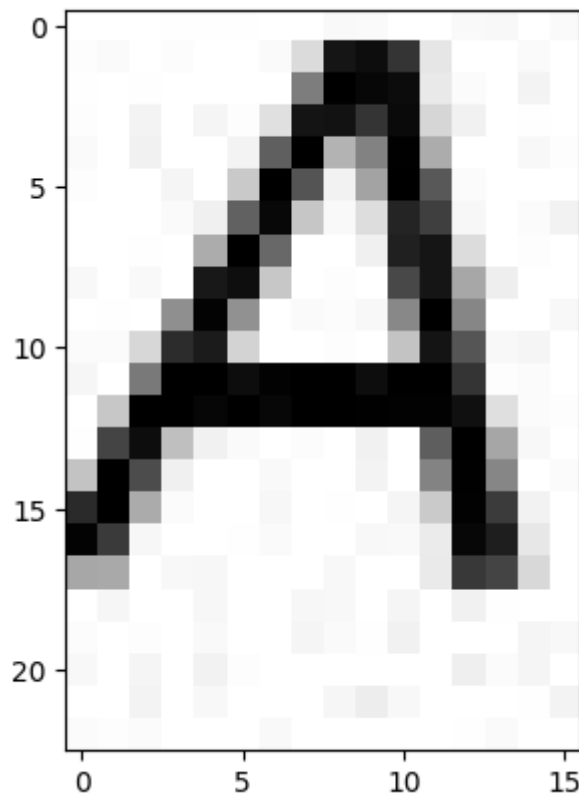
```
In [10]: # Nombres d'éléments par dimension
gray_img.shape
```

```
Out[10]: (23, 16)
```

```
In [11]: # Les quatres premiers éléments des 3 premières dimensions
print(gray_img[:3,:4])
```

```
[[255 255 255 254]
 [253 250 255 252]
 [254 255 254 255]]
```

```
In [12]: plt.imshow(gray_img, cmap="gray")
plt.show()
print(np.array_str(gray_img, precision=2, suppress_small=True))
```



```
[[255 255 255 254 254 254 255 255 249 250 255 255 249 248 255 250]
 [253 250 255 252 255 255 250 219 21 13 53 230 255 255 250 255]
 [254 255 254 255 255 255 255 125 0 7 12 233 251 255 243 255]
 [253 255 243 255 245 253 224 20 18 53 10 214 241 255 254 255]
 [250 255 241 255 255 244 95 0 179 130 1 172 255 255 248 253]
 [253 255 255 245 255 201 0 84 242 163 0 88 251 255 255 255]
 [255 255 255 250 240 97 8 198 249 221 37 63 247 255 251 242]
 [255 255 254 255 172 0 104 255 255 240 33 21 220 255 253 255]
 [248 255 249 255 24 14 199 255 253 255 72 21 167 238 255 255]
 [255 254 255 143 3 145 255 250 252 248 136 0 134 255 255 248]
 [251 251 215 44 27 211 255 255 253 255 195 20 85 248 245 255]
 [246 255 122 0 0 13 2 0 0 13 0 0 52 253 251 255]
 [255 199 0 0 6 0 7 0 0 1 2 2 16 223 251 253]
 [254 68 13 191 241 250 255 252 255 241 255 94 0 166 248 255]
 [195 0 76 240 255 255 249 255 255 243 255 132 1 134 255 251]
 [ 42 0 171 251 255 255 247 255 254 255 245 202 2 59 242 255]
 [ 3 58 247 255 255 254 250 255 255 248 249 235 7 30 231 255]
 [167 169 255 248 247 255 255 255 248 255 255 234 57 68 216 255]
 [255 246 255 255 247 255 255 247 248 255 245 255 241 254 255 255]
 [252 255 253 255 249 255 255 245 250 254 241 255 255 255 245 248]
 [248 255 245 255 242 253 255 255 255 255 255 239 251 245 255]
 [255 255 244 255 248 255 255 255 246 237 248 255 255 255 254 244]
 [251 253 251 255 255 255 249 255 255 255 255 254 250 255 255]]
```

Le seuillage

[Lien original](#)

```
In [13]: # Binarisation
img = cv2.imread('data/LetterA.jpg', 0)
```

```
In [14]: ## Function pour afficher plusieurs images
def plot_img(images, titles):
    fig, axs = plt.subplots(nrows = 1, ncols = len(images), figsize = (15, 15))
    for i, p in enumerate(images):
        axs[i].imshow(p, 'gray')
        axs[i].set_title(titles[i])
        #axs[i].axis('off')
    plt.show()
```

```
In [15]: #On supprime les valeurs en dessous du seuil 127 (niveau de gris)
#cv2.threshold(img, thresh_value, maxVal, style)
ret, img_binary = cv2.threshold(img, 127, 255, cv2.THRESH_BINARY)
```

```
In [16]: # Afficher image
images = [img, img_binary]
titles = ['Original image', 'THRESH_BINARY']
plot_img(images, titles)
```

