

CENTRO DE ENSEÑANZA TECNICA INDUSTRIAL



Practica 005

Funciones de umbrales para la recuperación de información

Gonzalez Yañez Leonardo Isaac

29/04/2025

Objetivo de la práctica

Utilizar distintos tipos de **umbrales (thresholds)** aplicados a imágenes para **recuperar información** visual importante, como texto o bordes, mediante técnicas como binarización fija, adaptativa y el método de Otsu.

Desarrollo de la práctica

Se utilizó la imagen `bookpage.jpg` en escala de grises para aplicar diferentes técnicas de umbral.

El código se dividió en tres bloques principales:

1. Umbral fijo (global thresholding)

- `THRESH_BINARY`: Píxeles $\geq t \rightarrow 255$, los demás $\rightarrow 0$
- `THRESH_BINARY_INV`: Inverso del anterior
- `THRESH_TRUNC`: Píxeles $\geq t$ se fijan en t
- `THRESH_TOZERO`: Píxeles $< t \rightarrow 0$
- `THRESH_TOZERO_INV`: Píxeles $\geq t \rightarrow 0$

2. Umbral adaptativo

- `ADAPTIVE_THRESH_MEAN_C`: Calcula el umbral como el promedio de los píxeles vecinos.
- `ADAPTIVE_THRESH_GAUSSIAN_C`: Usa una media ponderada gaussiana de la vecindad.

3. Umbral de Otsu

- `THRESH_OTSU`: Encuentra automáticamente el umbral óptimo para segmentar la imagen cuando tiene una distribución bimodal (oscuro/claro).

Se mostraron **nueve versiones de la imagen** en una sola ventana utilizando `Matplotlib`, para comparar visualmente el efecto de cada umbral.

Tipo de Umbral	Observaciones
THRESH_BINARY	Texto claro sobre fondo negro, buena separación.
THRESH_BINARY_INV	Fondo claro y texto negro, útil en negativo.
THRESH_TRUNC	Suaviza los bordes; no es binario.
THRESH_TOZERO / TOZERO_INV	Conserva tonos en zonas relevantes.
ADAPTIVE MEAN / GAUSSIAN	Excelente para textos con iluminación variable.
OTSU	Resultado automático muy efectivo para segmentación.

Conclusiones

- Las técnicas de **umbral fijo son simples** pero efectivas cuando la iluminación es uniforme.
- **Los umbrales adaptativos** son útiles en entornos con **variación de luz**, como documentos escaneados.
- El método de **Otsu** permite obtener un umbral automático muy preciso en imágenes bimodales.
- El umbral es una herramienta clave en **procesos de segmentación**, OCR (reconocimiento de caracteres), y visión artificial en general.

Código completo

```
import cv2

import numpy as np

from matplotlib import pyplot as plt


# Cargar la imagen en escala de grises

img = cv2.imread('bookpage.jpg', cv2.IMREAD_GRAYSCALE)


# 1. Umbral binario

_, thresh_binary = cv2.threshold(img, 127, 255, cv2.THRESH_BINARY)


# 2. Binario inverso

_, thresh_binary_inv = cv2.threshold(img, 127, 255,
cv2.THRESH_BINARY_INV)


# 3. Truncado

_, thresh_trunc = cv2.threshold(img, 127, 255, cv2.THRESH_TRUNC)
```

4. To Zero

```
_, thresh_tozero = cv2.threshold(img, 127, 255, cv2.THRESH_TOZERO)
```

5. To Zero Inverso

```
_, thresh_tozero_inv = cv2.threshold(img, 127, 255, cv2.THRESH_TOZERO_INV)
```

6. Adaptativo Media

```
thresh_mean = cv2.adaptiveThreshold(img, 255, cv2.ADAPTIVE_THRESH_MEAN_C, cv2.THRESH_BINARY, 11, 2)
```

7. Adaptativo Gaussiano

```
thresh_gauss = cv2.adaptiveThreshold(img, 255, cv2.ADAPTIVE_THRESH_GAUSSIAN_C, cv2.THRESH_BINARY, 11, 2)
```

8. Otsu

```
_, thresh_otsu = cv2.threshold(img, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
```

Mostrar resultados en una sola ventana con Matplotlib

```
titles = ['Original', 'Binary', 'Binary Inv', 'Trunc', 'ToZero', 'ToZero Inv',  
          'Adaptativo Media', 'Adaptativo Gauss', 'Otsu']
```

```
images = [img, thresh_binary, thresh_binary_inv, thresh_trunc,  
          thresh_tozero,  
          thresh_tozero_inv, thresh_mean, thresh_gauss, thresh_otsu]
```

```
plt.figure(figsize=(12, 8))
```

```
for i in range(9):
```

```
    plt.subplot(3, 3, i + 1)
```

```
    plt.imshow(images[i], cmap='gray')
```

```
    plt.title(titles[i])
```

```
    plt.xticks([], plt.yticks([]))
```

```
plt.tight_layout()
```

```
plt.show()
```