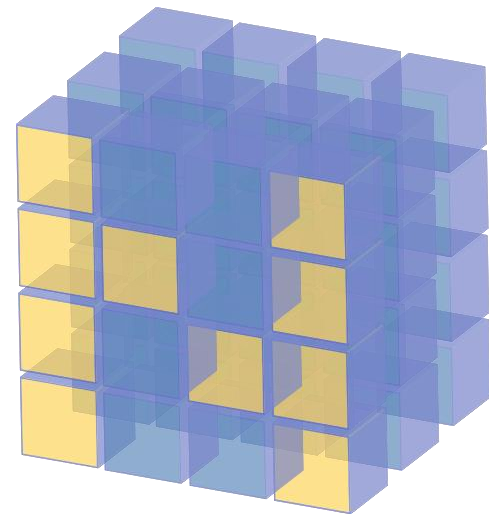


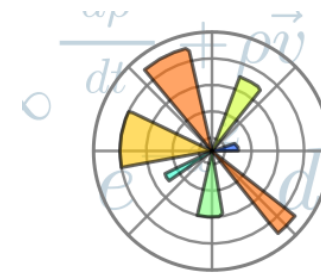
PROCESAMIENTO DIGITAL DE IMAGENES



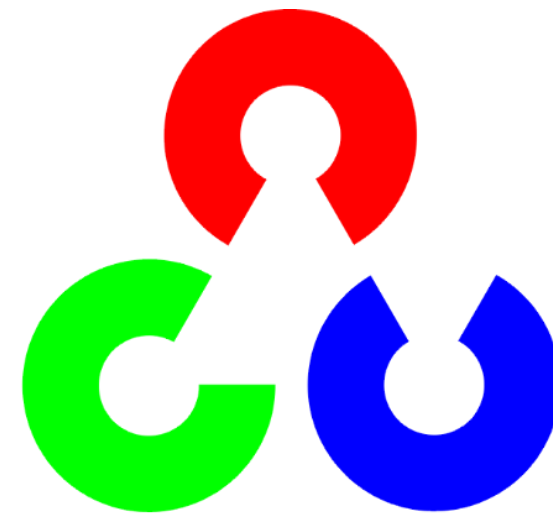
Módulos principales



NumPy



matplotlib



OpenCV

`pip install opencv-python`

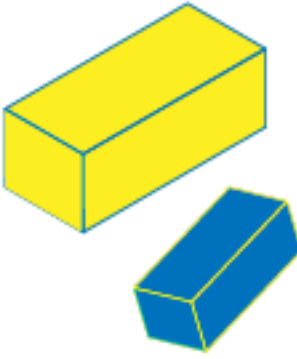


PROCESAMIENTO
DIGITAL DE
IMÁGENES

UMAKER | CENTRO DE CAPACITACIÓN
DE DESARROLLO TECNOLÓGICO

TEMAS QUE VEREMOS HOY

- Umbralización
- Umbralización Simple
- Umbralización Adaptativa
- Umbralización de Otsu



PROCESAMIENTO
DIGITAL DE
IMÁGENES

UMBRALIZACIÓN

- Consiste en convertir los píxeles al color blanco o negro de una imagen en escala de grises
- Es el método de segmentación de imágenes más simple



PROCESAMIENTO
DIGITAL DE
IMÁGENES

UMBRALIZACIÓN

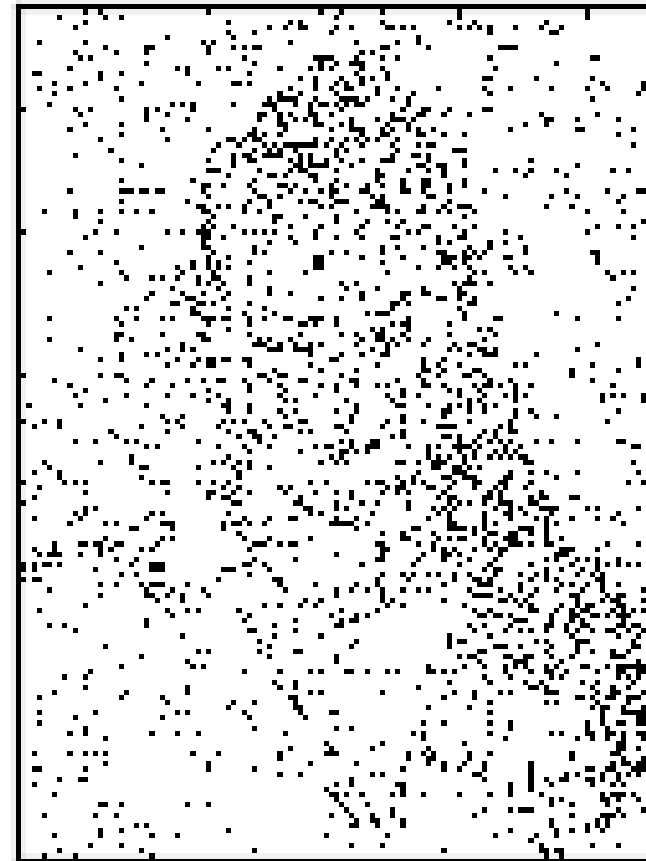
- La conversión de los píxeles se determina mediante los valores de intensidad de cada píxel
- Se utiliza un valor «umbral» a partir del cual se determina si un píxel será del color blanco o negro mediante su intensidad



PROCESAMIENTO
DIGITAL DE
IMÁGENES

TIPOS DE UMBRALIZACIÓN

- La variable “x” indica la posición de un píxel en el eje X
- La variable “y” indica la posición de un píxel en el eje Y
- El valor umbral es representado mediante «umbral»
- La intensidad de un píxel es representado mediante «intensidad»



$$\text{intensidad}(x, y) \in [0, 255]$$



PROCESAMIENTO
DIGITAL DE
IMÁGENES

TIPOS DE UMBRALIZACIÓN (UMBRAL BINARIO)

- Si la intensidad de un píxel es mayor al valor «umbral», entonces el valor del píxel es modificado
- Los píxeles se modifican por un valor predefinido en lugar de cambiarlo solamente por el color blanco
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces el valor del píxel es modificado al color negro

$$\text{intensidad}(x, y) = \begin{cases} \text{valor_predefinido}, & \text{intensidad}(x, y) > \text{umbral} \\ 0, & \text{intensidad}(x, y) \leq \text{umbral} \end{cases}$$



TIPOS DE UMBRALIZACIÓN (UMBRAL BINARIO)

- Si la intensidad de un píxel es mayor al valor «umbral», entonces el valor del píxel es modificado
- Los píxeles se modifican por un valor predefinido en lugar de cambiarlo solamente por el color blanco
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces el valor del píxel es modificado al color negro

valor_predefinido=200

umbral=150

$$\begin{bmatrix} 20 & 210 & 54 & 87 & 108 \\ 57 & 68 & 150 & 213 & 14 \\ 68 & 98 & 24 & 207 & 192 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 200 & 0 & 0 & 0 \\ 0 & 0 & 0 & 200 & 0 \\ 0 & 0 & 0 & 200 & 200 \end{bmatrix}$$



TIPOS DE UMBRALIZACIÓN (UMBRALE BINARIO INVERTIDO)

- Si la intensidad de un píxel es mayor al valor «umbral», entonces el valor del píxel es modificado por el color negro
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces el valor del píxel es modificado
- Los píxeles se modifican por un valor predefinido en lugar de cambiarlo solamente por el color blanco

$$\text{intensidad}(x, y) = \begin{cases} 0 & , \text{intensidad}(x, y) > \text{umbral} \\ \text{valor_predefinido} & , \text{intensidad}(x, y) \leq \text{umbral} \end{cases}$$



TIPOS DE UMBRALIZACIÓN (UMBRAL BINARIO INVERTIDO)

- Si la intensidad de un píxel es mayor al valor «umbral», entonces el valor del píxel es modificado por el color negro
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces el valor del píxel es modificado
- Los píxeles se modifican por un valor predefinido en lugar de cambiarlo solamente por el color blanco

valor_predefinido=200

umbral=150

$$\begin{bmatrix} 20 & 210 & 54 & 87 & 108 \\ 57 & 68 & 150 & 213 & 14 \\ 68 & 98 & 24 & 207 & 192 \end{bmatrix} \rightarrow \begin{bmatrix} 200 & 0 & 200 & 200 & 200 \\ 200 & 200 & 200 & 0 & 200 \\ 200 & 200 & 200 & 0 & 0 \end{bmatrix}$$

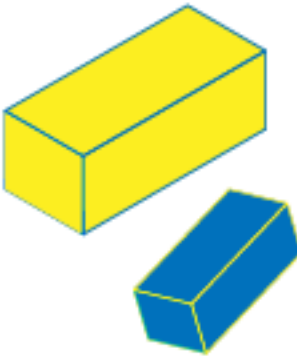


PROCESAMIENTO
DIGITAL DE
IMÁGENES

TIPOS DE UMBRALIZACIÓN (TRUNCAR)

- Si la intensidad de un píxel es mayor al valor «umbral», entonces el valor del píxel es modificado por el valor «umbral»
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces el valor del píxel no es modificado
- Debido que el máximo valor de un píxel es el valor «umbral», se dice que los valores de los píxeles se truncan

$$\text{intensidad}(x, y) = \begin{cases} \text{umbral} & , \text{intensidad}(x, y) > \text{umbral} \\ \text{intensidad}(x, y) & , \text{intensidad}(x, y) \leq \text{umbral} \end{cases}$$



TIPOS DE UMBRALIZACIÓN (TRUNCAR)

- Si la intensidad de un píxel es mayor al valor «umbral», entonces el valor del píxel es modificado por el valor «umbral»
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces el valor del píxel no es modificado
- Debido que el máximo valor de un píxel es el valor «umbral», se dice que los valores de los píxeles se truncan

umbral=150

$$\begin{bmatrix} 20 & 210 & 54 & 87 & 108 \\ 57 & 68 & 150 & 213 & 14 \\ 68 & 98 & 24 & 207 & 192 \end{bmatrix} \rightarrow \begin{bmatrix} 20 & 150 & 54 & 87 & 108 \\ 57 & 68 & 150 & 150 & 14 \\ 68 & 98 & 24 & 150 & 150 \end{bmatrix}$$



TIPOS DE UMBRALIZACIÓN (UMBRALES A CERO)

- Si la intensidad de un píxel es mayor al valor «umbral», entonces el valor del píxel no es modificado
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces el valor del píxel es modificado por el color negro
- Debido que el mínimo valor de un píxel es cero, se dice que los valores de los píxeles se ajustan a cero

$$\text{intensidad}(x, y) = \begin{cases} \text{intensidad}(x, y), & \text{intensidad}(x, y) > \text{umbral} \\ 0, & \text{intensidad}(x, y) \leq \text{umbral} \end{cases}$$



TIPOS DE UMBRALIZACIÓN (UMBRAL A CERO)

- Si la intensidad de un píxel es mayor al valor «umbral», entonces el valor del píxel no es modificado
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces el valor del píxel es modificado por el color negro
- Debido que el mínimo valor de un píxel es cero, se dice que los valores de los píxeles se ajustan a cero

umbral=150

$$\begin{bmatrix} 20 & 210 & 54 & 87 & 108 \\ 57 & 68 & 150 & 213 & 14 \\ 68 & 98 & 24 & 207 & 192 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 210 & 0 & 0 & 0 \\ 0 & 0 & 0 & 213 & 0 \\ 0 & 0 & 0 & 207 & 192 \end{bmatrix}$$



TIPOS DE UMBRALIZACIÓN (UMBRALES A CERO INVERTIDO)

- Si la intensidad de un píxel es mayor al valor «umbral», entonces el valor del píxel es modificado por el color negro
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces el valor del píxel no es modificado
- Debido que los valores mayores al valor «umbral» son iguales a cero, se dice que los valores de los píxeles se ajustan, de manera invertida, a cero

$$\text{intensidad}(x, y) = \begin{cases} 0 & , \text{intensidad}(x, y) > \text{umbral} \\ \text{intensidad}(x, y), & \text{intensidad}(x, y) \leq \text{umbral} \end{cases}$$



TIPOS DE UMBRALIZACIÓN (UMBRAL A CERO INVERTIDO)

- Si la intensidad de un píxel es mayor al valor «umbral», entonces el valor del píxel es modificado por el color negro
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces el valor del píxel no es modificado
- Debido que los valores mayores al valor «umbral» son iguales a cero, se dice que los valores de los píxeles se ajustan, de manera invertida, a cero

umbral=150

$$\begin{bmatrix} 20 & 210 & 54 & 87 & 108 \\ 57 & 68 & 150 & 213 & 14 \\ 68 & 98 & 24 & 207 & 192 \end{bmatrix} \rightarrow \begin{bmatrix} 20 & 0 & 54 & 87 & 108 \\ 57 & 68 & 150 & 0 & 14 \\ 68 & 98 & 24 & 0 & 0 \end{bmatrix}$$



UMBRALIZACIÓN SIMPLE

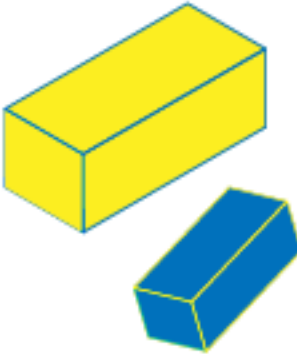
- Si la intensidad de un píxel es mayor al valor «umbral», entonces se le asigna un valor (normalmente el color blanco)
- Si la intensidad de un píxel es menor o igual al valor «umbral», entonces se le asigna un valor distinto (normalmente el color negro)
- La imagen a la cual aplicar la umbralización debe estar en escala de grises
- Los valores umbrales, en OpenCV, pueden ser:

THRESH_BINARY
THRESH_BINARY_INV
THRESH_TRUNC
THRESH_TOZERO
THRESH_TOZERO_INV



PROCESAMIENTO
DIGITAL DE
IMÁGENES

Umbralizar una imagen



threshold : Aplica un valor umbral pre-establecido a cada píxel de la imagen

Sintaxis: cv2.threshold(imagen, umbral, valor_máximo, tipo)

Parametros:

imagen : Imagen a la cual aplicar la umbralización

umbral : El valor «umbral» a aplicar

valor_máximo : Es el valor predefinido para THRESH_BINARY y THRESH_BINARY_INV

tipo : Tipo de umbralización a usar

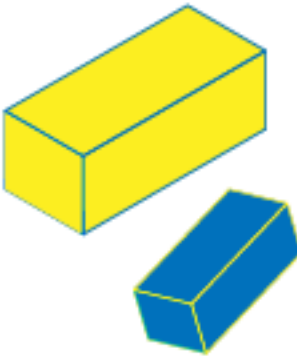
Nota El parámetro “valor_máximo” normalmente es definido como 255 para representar el color blanco



UMBRALIZACIÓN SIMPLE (THRESH_BINARY)

- Hace referencia al «Umbral Binario»
- Por defecto, si la intensidad es menor o igual al valor «umbral», entonces el valor del píxel es cero
- Está definido por:

$$\text{intensidad}(x, y) = \begin{cases} \text{valor_predefinido}, & \text{intensidad}(x, y) > \text{umbral} \\ 0, & \text{intensidad}(x, y) \leq \text{umbral} \end{cases}$$



Ejemplo:

```
import cv2

image_as_BGR = cv2.imread("random_scene.jpg")
image_as_GRAY = cv2.cvtColor(image_as_BGR, cv2.COLOR_BGR2GRAY)

valor_umbral = 127
valor_maximo = 255
tipo = cv2.THRESH_BINARY

return_value, image = cv2.threshold(image_as_GRAY, valor_umbral, valor_maximo, tipo)

cv2.imshow("Imagen", image_as_GRAY)
cv2.imshow("Imagen como la Umbralización", image)

cv2.waitKey(0)

cv2.destroyAllWindows()
```

UMBRALIZACIÓN SIMPLE (THRESH_BINARY_INV)

- Hace referencia al «Umbral Binario Invertido»
- Por defecto, si la intensidad es mayor al valor «umbral», entonces el valor del píxel es cero
- Está definido por:

$$\text{intensidad}(x, y) = \begin{cases} 0 & , \text{intensidad}(x, y) > \text{umbral} \\ \text{valor_predefinido} & , \text{intensidad}(x, y) \leq \text{umbral} \end{cases}$$



Ejemplo:

```
import cv2

image_as_BGR = cv2.imread("random_scene.jpg")
image_as_GRAY = cv2.cvtColor(image_as_BGR, cv2.COLOR_BGR2GRAY)

valor_umbral = 127
valor_maximo = 255
tipo = cv2.THRESH_BINARY_INV

return_value, image = cv2.threshold(image_as_GRAY, valor_umbral, valor_maximo, tipo)

cv2.imshow("Imagen", image_as_GRAY)
cv2.imshow("Imagen como la Umbralización", image)

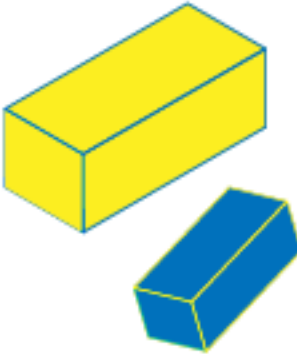
cv2.waitKey(0)

cv2.destroyAllWindows()
```


UMBRALIZACIÓN SIMPLE (THRESH_TRUNC)

- Hace referencia al «Truncar»
- Por defecto, si la intensidad es mayor al valor «umbral», entonces el valor del píxel es igual al valor «umbral»
- Está definido por:

$$\text{intensidad}(x, y) = \begin{cases} \text{umbral} & , \text{intensidad}(x, y) > \text{umbral} \\ \text{intensidad}(x, y) & , \text{intensidad}(x, y) \leq \text{umbral} \end{cases}$$



PROCESAMIENTO
DIGITAL DE
IMÁGENES

Ejemplo:

```
import cv2

image_as_BGR = cv2.imread("random_scene.jpg")
image_as_GRAY = cv2.cvtColor(image_as_BGR, cv2.COLOR_BGR2GRAY)

valor_umbral = 127
valor_maximo = 255
tipo = cv2.THRESH_TRUNC

return_value, image = cv2.threshold(image_as_GRAY, valor_umbral, valor_maximo, tipo)

cv2.imshow("Imagen", image_as_GRAY)
cv2.imshow("Imagen como la Umbralización", image)

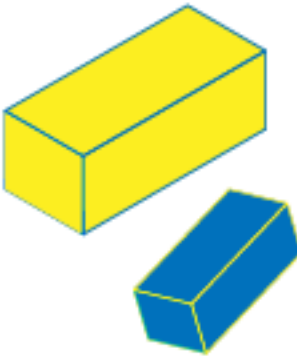
cv2.waitKey(0)

cv2.destroyAllWindows()
```

UMBRALIZACIÓN SIMPLE (THRESH_TOZERO)

- Hace referencia al «Umbral a Cero»
- Por defecto, si la intensidad es mayor al valor «umbral», entonces el valor del píxel es cero
- Está definido por:

$$\text{intensidad}(x, y) = \begin{cases} \text{intensidad}(x, y), & \text{intensidad}(x, y) > \text{umbral} \\ 0, & \text{intensidad}(x, y) \leq \text{umbral} \end{cases}$$



PROCESAMIENTO
DIGITAL DE
IMÁGENES

Ejemplo:

```
import cv2

image_as_BGR = cv2.imread("random_scene.jpg")
image_as_GRAY = cv2.cvtColor(image_as_BGR, cv2.COLOR_BGR2GRAY)

valor_umbral = 127
valor_maximo = 255
tipo = cv2.THRESH_TOZERO

return_value, image = cv2.threshold(image_as_GRAY, valor_umbral, valor_maximo, tipo)

cv2.imshow("Imagen", image_as_GRAY)
cv2.imshow("Imagen como la Umbralización", image)

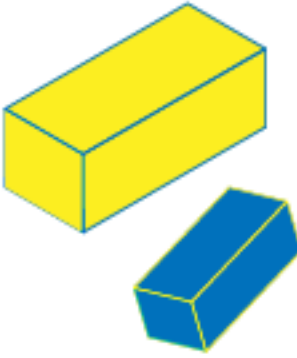
cv2.waitKey(0)

cv2.destroyAllWindows()
```

UMBRALIZACIÓN SIMPLE (THRESH_TOZERO_INV)

- Hace referencia al «Umbral a Cero Invertido»
- Por defecto, si la intensidad es mayor al valor «umbral», entonces el valor del píxel es igual a cero
- Está definido por:

$$\text{intensidad}(x, y) = \begin{cases} 0 & , \text{intensidad}(x, y) > \text{umbral} \\ \text{intensidad}(x, y), & \text{intensidad}(x, y) \leq \text{umbral} \end{cases}$$



Ejemplo:

```
import cv2

image_as_BGR = cv2.imread("random_scene.jpg")
image_as_GRAY = cv2.cvtColor(image_as_BGR, cv2.COLOR_BGR2GRAY)

valor_umbral = 127
valor_maximo = 255
tipo = cv2.THRESH_TOZERO_INV

return_value, image = cv2.threshold(image_as_GRAY, valor_umbral, valor_maximo, tipo)

cv2.imshow("Imagen", image_as_GRAY)
cv2.imshow("Imagen como la Umbralización", image)

cv2.waitKey(0)

cv2.destroyAllWindows()
```

UMBRALIZACIÓN ADAPTATIVA

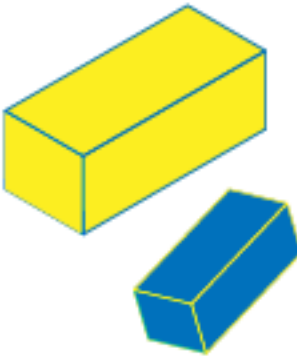
- Debido que las imágenes difieren en las condiciones de luz dependiendo del área, en algunos casos no se pueden obtener los resultados deseados al usar un valor «umbral»
- En lugar de usar un valor «umbral» fijo, se usa una umbralización adaptativa
- Se calcula el valor «umbral» en pequeñas regiones de la imagen
- Para diferentes regiones se obtiene diferentes valores umbrales
- Se obtiene mejores resultados para imágenes con una iluminación variante
- Los valores umbrales, en OpenCV, pueden ser:

ADAPTIVE_THRESH_MEAN_C
ADAPTIVE_THRESH_GAUSSIAN_C



PROCESAMIENTO
DIGITAL DE
IMÁGENES

Umbralizar una imagen



adaptiveThreshold: Aplica un valor umbral adaptativo a una imagen

Sintaxis: `cv2.adaptiveThreshold(imagen, valor_máximo, método, tipo, N[, C])`

Parametros:

imagen : Imagen a la cual aplicar la umbralización

valor_máximo : Valor distinto de cero asignado a los píxeles cuando se cumple la condición

método : Es el método de umbralización adaptativo a usar

tipo : Tipo de umbralización a usar. Debe ser THRESH_BINARY ó THRESH_BINARY_INV

N : Es el tamaño del bloque para la vecindad. Debe ser impar

C : Es el valor de la constante

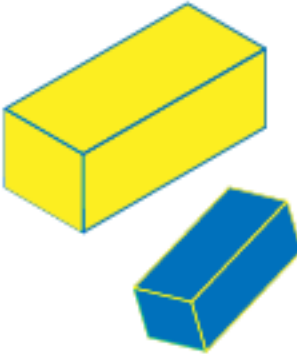
Nota Funciona para imágenes de 8 bits con un sólo canal



PROCESAMIENTO
DIGITAL DE
IMÁGENES

UMBRALIZACIÓN ADAPTATIVA (ADAPTIVE_THRESH_MEAN_C)

- El valor «umbral» se obtiene a partir de la vecindad
- El valor «umbral» es una media de la vecindad $N \times N$ para la coordenada (x, y)
- N es definido como tamaño del bloque
- A la media se le resta un valor constante
- El valor constante es definido como C



Ejemplo:

```
import cv2

image_as_BGR = cv2.imread("random_scene.jpg")
image_as_GRAY = cv2.cvtColor(image_as_BGR, cv2.COLOR_BGR2GRAY)

image_as_GRAY = cv2.medianBlur(image_as_GRAY, 5)

valor_umbral = 127
valor_maximo = 255
tipo = cv2.THRESH_BINARY

return_value, image_with_THRESH_BINARY = cv2.threshold(image_as_GRAY, valor_umbral, valor_maximo, tipo)

valor_maximo = 255
metodo = cv2.ADAPTIVE_THRESH_MEAN_C
tipo = cv2.THRESH_BINARY
N = 5
C = 2

image_with_ADAPTIVE_THRESH_MEAN_C = cv2.adaptiveThreshold(image_as_GRAY, valor_maximo, metodo, tipo, N, C)

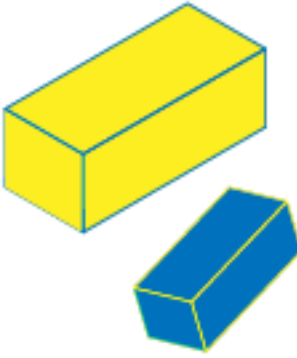
cv2.imshow("Imagen", image_as_GRAY)
cv2.imshow("Imagen con THRESH_BINARY", image_with_THRESH_BINARY)
cv2.imshow("Imagen con ADAPTIVE_THRESH_MEAN_C", image_with_ADAPTIVE_THRESH_MEAN_C)

cv2.waitKey(0)

cv2.destroyAllWindows()
```

UMBRALIZACIÓN ADAPTATIVA (ADAPTIVE_THRESH_GAUSSIAN_C)

- El valor «umbral» se obtiene a partir de la vecindad
- El valor «umbral» es una suma ponderada de la vecindad $N \times N$ para la coordenada (x, y)
- N es definido como tamaño del bloque
- A la suma ponderada se le resta un valor constante
- El valor constante es definido como C



PROCESAMIENTO
DIGITAL DE
IMÁGENES

Ejemplo:

```
import cv2

image_as_BGR = cv2.imread("random_scene.jpg")
image_as_GRAY = cv2.cvtColor(image_as_BGR, cv2.COLOR_BGR2GRAY)

image_as_GRAY = cv2.medianBlur(image_as_GRAY, 5)

valor_umbral = 127
valor_maximo = 255
tipo = cv2.THRESH_BINARY

return_value, image_with_THRESH_BINARY = cv2.threshold(image_as_GRAY, valor_umbral, valor_maximo, tipo)

valor_maximo = 255
metodo = cv2.ADAPTIVE_THRESH_GAUSSIAN_C
tipo = cv2.THRESH_BINARY
N = 5
C = 2

image_with_ADAPTIVE_THRESH_GAUSSIAN_C = cv2.adaptiveThreshold(image_as_GRAY, valor_maximo, metodo, tipo, N, C)

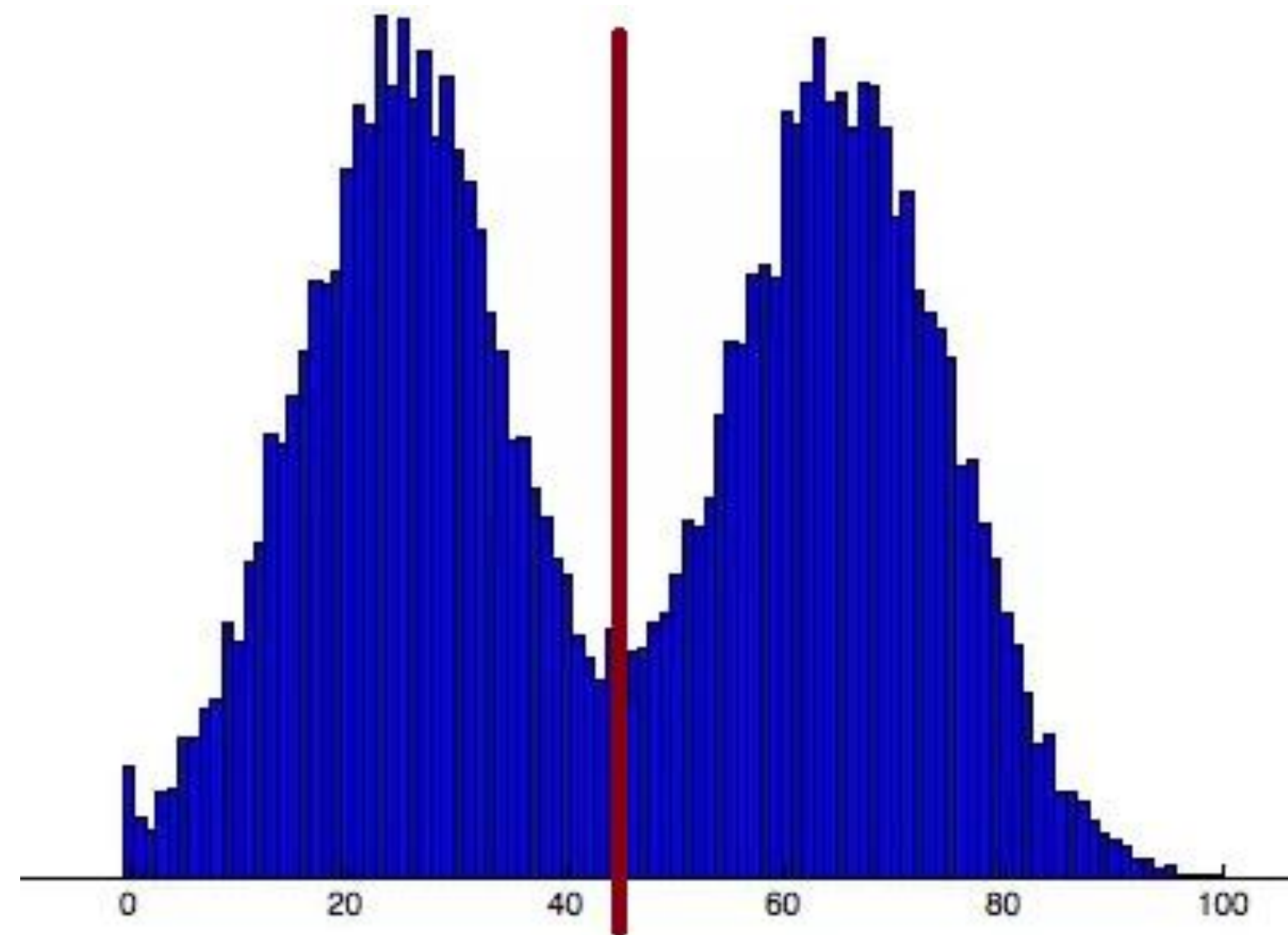
cv2.imshow("Imagen", image_as_GRAY)
cv2.imshow("Imagen con THRESH_BINARY", image_with_THRESH_BINARY)
cv2.imshow("Imagen con ADAPTIVE_THRESH_GAUSSIAN_C", image_with_ADAPTIVE_THRESH_GAUSSIAN_C)

cv2.waitKey(0)

cv2.destroyAllWindows()
```

UMBRALIZACIÓN DE OTSU

- Usado para imágenes bimodales
- Una imagen bimodal es aquella cuyo histograma tiene 2 picos
- El valor «umbral» es calculado a partir del histograma de la imagen bimodal
- No se recomienda su uso para imágenes no bimodales
- Se usa en combinación de la umbralización simple



PROCESAMIENTO
DIGITAL DE
IMÁGENES

Ejemplo:

```
import cv2
import matplotlib.pyplot as plt

image_as_BGR = cv2.imread("image_with_noise.png")
image_as_GRAY = cv2.cvtColor(image_as_BGR, cv2.COLOR_BGR2GRAY)

valor_umbral = 127
valor_maximo = 255
tipo = cv2.THRESH_BINARY

return_value, image_with_THRESH_BINARY = cv2.threshold(image_as_GRAY, valor_umbral, valor_maximo, tipo)

image_as_GRAY = cv2.GaussianBlur(image_as_GRAY, (7, 9), 27)

valor_umbral = 0
valor_maximo = 255
tipo = cv2.THRESH_BINARY + cv2.THRESH_OTSU

return_value, image_with_THRESH_BINARY_and_THRESH_OTSU = cv2.threshold(image_as_GRAY, valor_umbral, valor_maximo, tipo)

histogram_for_gray = cv2.calcHist([image_as_GRAY], [0], None, [256], [0, 256])

cv2.imshow("Imagen", image_as_GRAY)
cv2.imshow("Imagen con THRESH_BINARY", image_with_THRESH_BINARY)
cv2.imshow("Imagen con THRESH_BINARY y THRESH_OTSU", image_with_THRESH_BINARY_and_THRESH_OTSU)

plt.plot(histogram_for_gray, color="gray")

plt.show()

cv2.waitKey(0)

cv2.destroyAllWindows()
```


UMAKER | CENTRO DE CAPACITACIÓN
DE DESARROLLO TECNOLÓGICO