

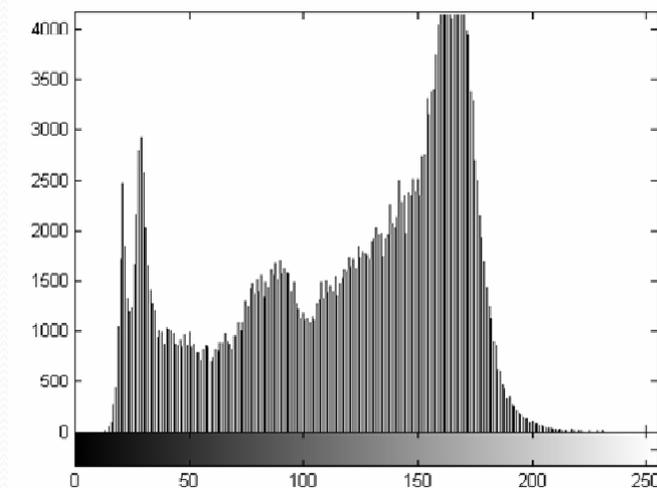
Histogramas

Temario

- Histogramas para imágenes en tonos de gris.
- Histogramas para imágenes a color.
- Ecuación del histograma.
- Histogramas en 2D.
- Histogram Backprojection.

Histograma

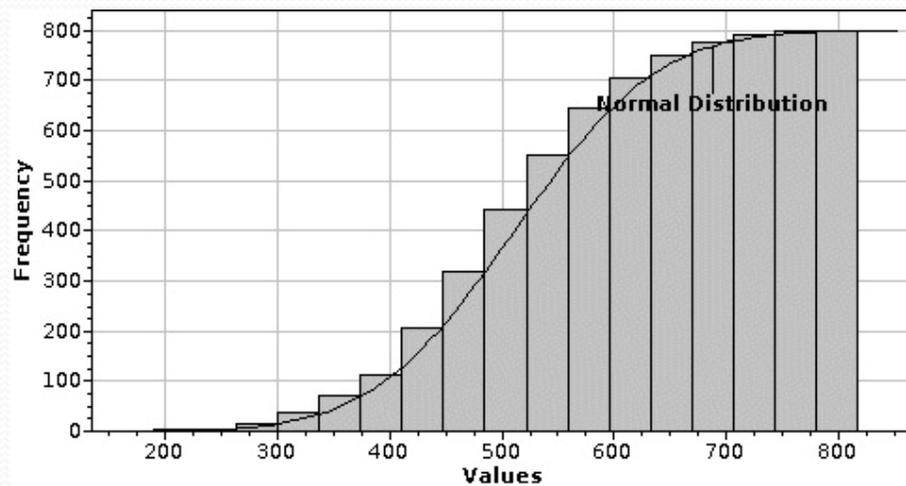
- **Histograma.** Informa cuántos pixeles existen de cada valor.
- Python: `cv2.calcHist()`



Fuente: https://www.researchgate.net/figure/Histogram-Analysis-of-grayscale-image_fig2_228744430

Histograma acumulativo

- **Histograma acumulativo.** Conteo del número de pixeles menores o igual a cada valor.

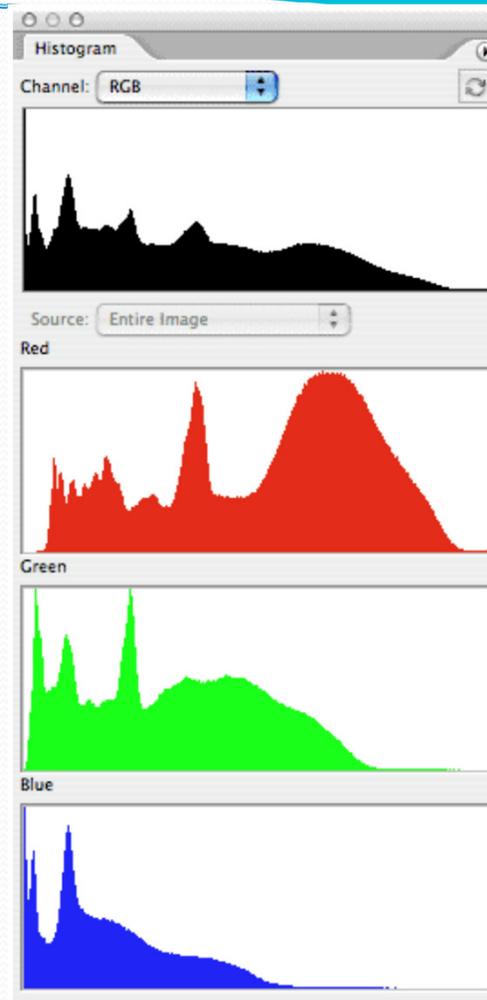


Fuente: <https://mipav.cit.nih.gov/pubwiki/images/1/19/CumulativeHistogramSample.jpg>

Histogramas de imágenes a color

- Dos tipos de histogramas para imágenes a color:
 1. Histograma de intensidad.
 2. Histogramas individuales por cada canal de color.

Ejemplo



Fuente: <https://kenrockwell.com/tech/yrgb.htm>

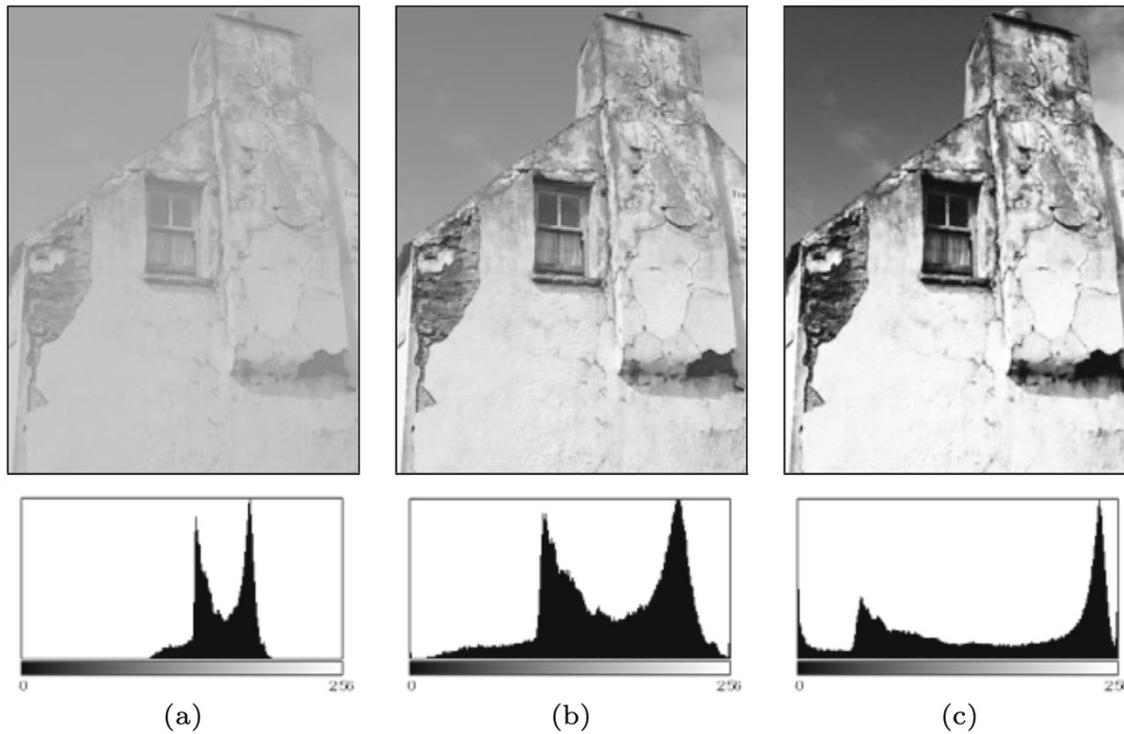
Histogramas

- ¿Para qué sirve el histograma de una imagen?
- Da información sobre el contraste y rango dinámico de una imagen.
- El histograma se puede manipular para mejorar una imagen.

Contraste

- Es la diferencia entre los valores máximo y mínimo de los píxeles de la imagen ($a = a_{max} - a_{min}$).
- El contraste de la imagen se puede ver directamente en el histograma.

Contraste



3.2 INTERPRETING HISTOGRAMS

Fig. 3.7

How changes in contrast affect the histogram: low contrast (a), normal contrast (b), high contrast (c).

Rango dinámico

- **Rango dinámico.** La relación (ratio o razón) entre los valores más grandes y más pequeños de intensidad.
- Se mide como una relación, en decibeles o en stops.

Ejemplo

- Suponer que en una imagen de 8 bits $a_{low} = 10$ y $a_{high} = 250$.
- El rango dinámico es:
- Como relación (ratio) $DR = \frac{250}{10} = 25 = 25:1$.
- En decibeles $DR = 10 \times \log_{10} \left(\frac{250}{10} \right) = 13.98 \text{ dB}$.
- En stops $DR = \log_2 \left(\frac{250}{10} \right) = 4.64 \text{ stops}$.

Rango dinámico

- Fórmulas generales:
- Relación: $\frac{a_{high}}{\max(1, a_{low})}$
- Decibeles: $10 \times \log_{10} \left(\frac{a_{high}}{\max(1, a_{low})} \right)$ dB
- Stops: $\log_2 \left(\frac{a_{high}}{\max(1, a_{low})} \right)$ stops
- En el denominador se usa $\max(1, a_{low})$ porque es posible que a_{low} valga 0.
- Los decibeles se emplean en sonido y los stops en fotografía y en imágenes digitales.

Técnicas para mejorar el contraste

- Autocontraste.
- Operadores logaritmo y exponencial.
- Corrección gamma.
- Ecuación del histograma.
- CLAHE.



Ecualización del histograma

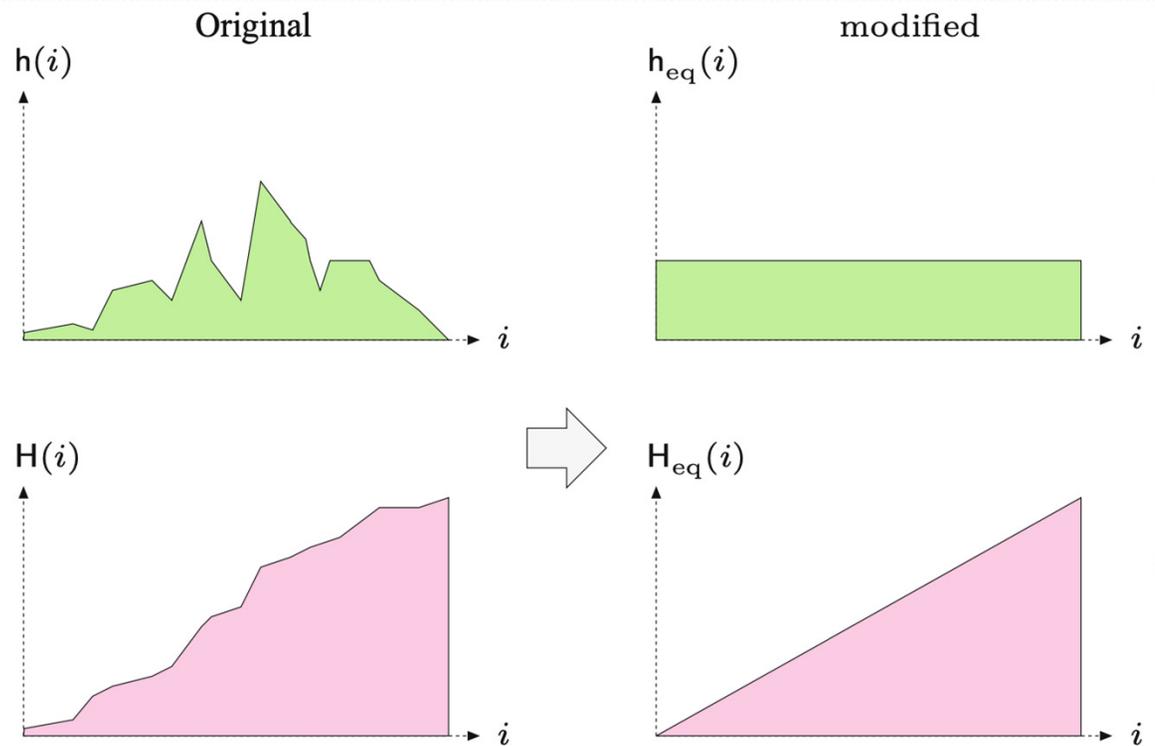
- El objetivo es modificar una imagen de tal forma que el histograma se aproxime a una distribución uniforme.

Ecuación del histograma

4 POINT OPERATIONS

Fig. 4.8

Histogram equalization. The idea is to find and apply a point operation to the image (with original histogram h) such that the histogram h_{eq} of the modified image approximates a *uniform* distribution (top). The cumulative target histogram H_{eq} must thus be approximately wedge-shaped (bottom).





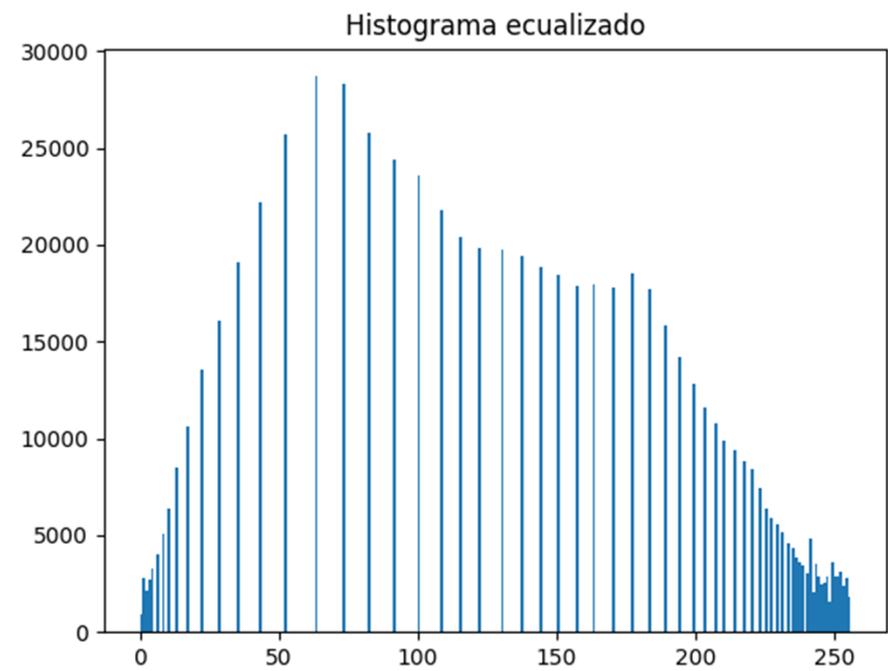
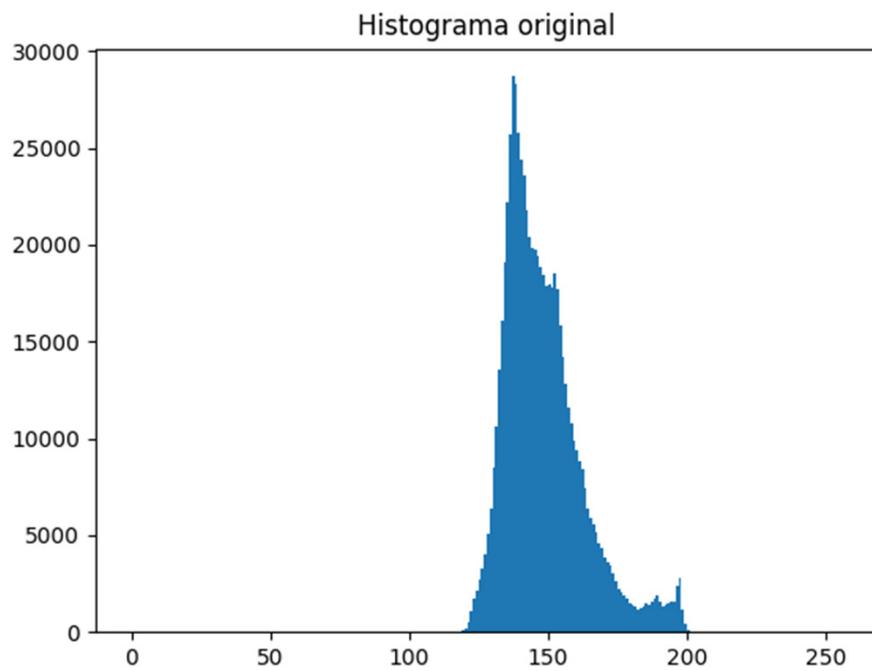
Ecualización del histograma en OpenCV

`cv2.equalizeHist()`

Resultado



Resultado



CLAHE

- Ecuación de histograma adaptativo limitado de contraste (Contrast Limited Adaptive Histogram Equalization).
- La ecuación del histograma considera el contraste global de la imagen.
- Esto puede no ser conveniente en algunas imágenes.

Ecualización global



CLAHE

- Divide la imagen en bloques (8 x 8 por default).
- A cada bloque se le aplica la ecualización del histograma.
- A cada bloque se le limita el contraste.

CLAHE en OpenCV

- Paso 1: Crear un objeto CLAHE (los argumentos son opcionales).
`clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8, 8))`
- Paso 2: Invocar el método `apply`.
`image_clahe = clahe.apply(image)`

Resultado



Ecualización del histograma en color

- Aplicar una ecualización a cada canal de una imagen RGB puede distorsionar el color.



Ecualización del histograma en color

- Solución: convertir la imagen a un espacio en donde la intensidad esté separada de la información del color, por ejemplo YUV o YCbCr, y ecualizar solo en canal Y.

Ecualización YUV



Aplicación: histogram backprojection

- **Histogram Backprojection.** Técnica utilizada para encontrar objetos de interés en una imagen.
- El objeto se busca en base a sus colores.
- Para entender esta técnica, se necesita revisar el concepto de histogramas 2D.

Histogramas 2D

- **Histograma 2D.** Herramienta para encontrar la correlación entre los canales de una imagen.
- Ejemplo: un histograma 2D para los canales rojo y verde, cuenta cuántas veces aparece en la imagen el par (r, g) , donde $r \in [0, 255]$ y $g \in [0, 255]$.

Construcción de un histograma 2D

- Ejemplo tomado de <https://theailearner.com/2019/02/12/2d-histogram/>
- Suponer una imagen de 2 bits de 4 x 4 con los siguientes canales rojo y verde:

3	2	0	3
0	2	0	1
1	3	0	2
3	0	2	2

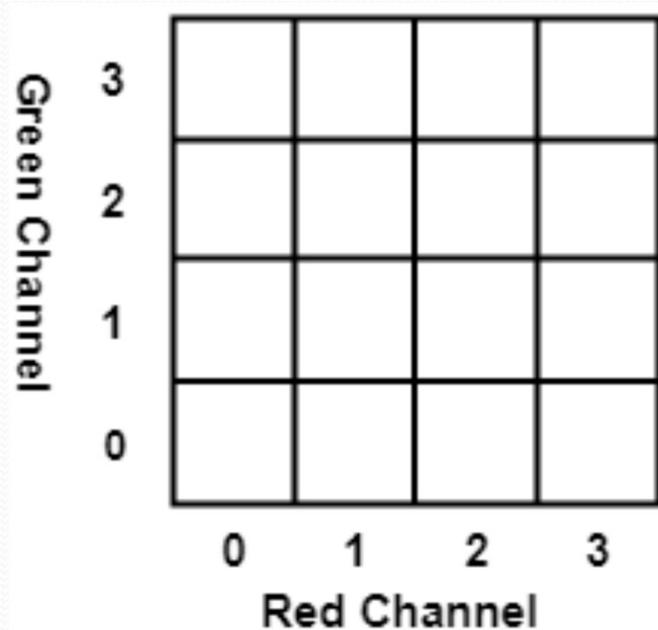
Red Channel

0	1	3	2
3	1	2	0
2	3	0	1
0	1	3	1

Green Channel

Construcción de un histograma 2D

- El histograma 2D se guarda en una matriz de 4 x 4:



Construcción de un histograma 2D

- Se recorre cada posición en los canales contando las parejas de intensidades:

Corresponding Intensity Pairs	Count
(0,0)	1
(0,1)	1
(0,2)	1
●	
●	
●	
(3,2)	1
(3,3)	1



Green Channel	3	2	0	1	1
	2	1	1	0	1
	1	1	0	4	0
	0	1	1	0	2
		0	1	2	3
		Red Channel			

Fill the 2D histogram with count

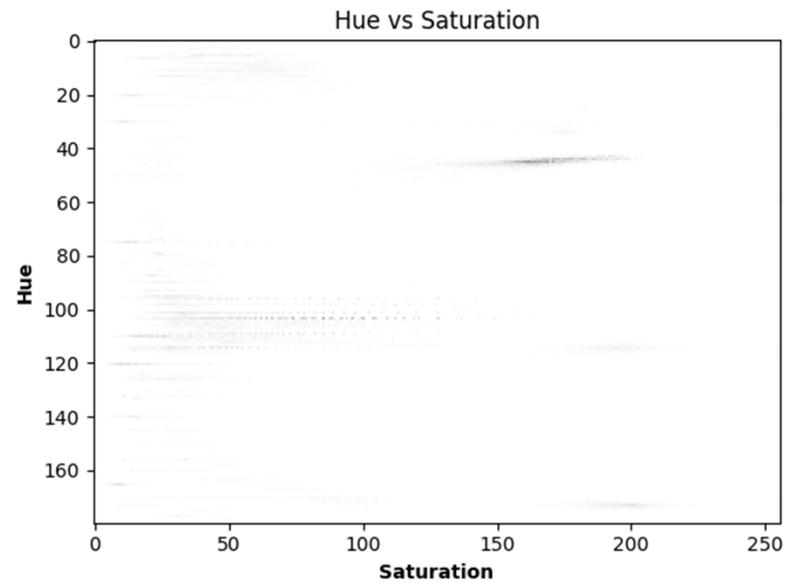
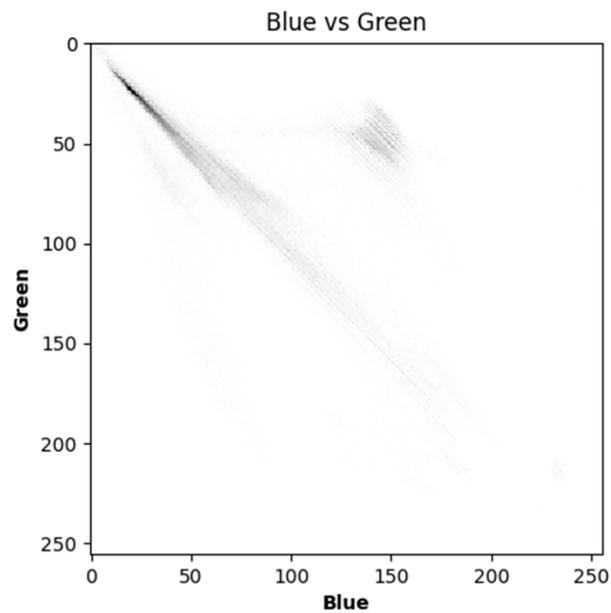
Ejemplo

- Imagen original:



Ejemplo

- Histogramas 2D:



Histogramas 2D en OpenCV

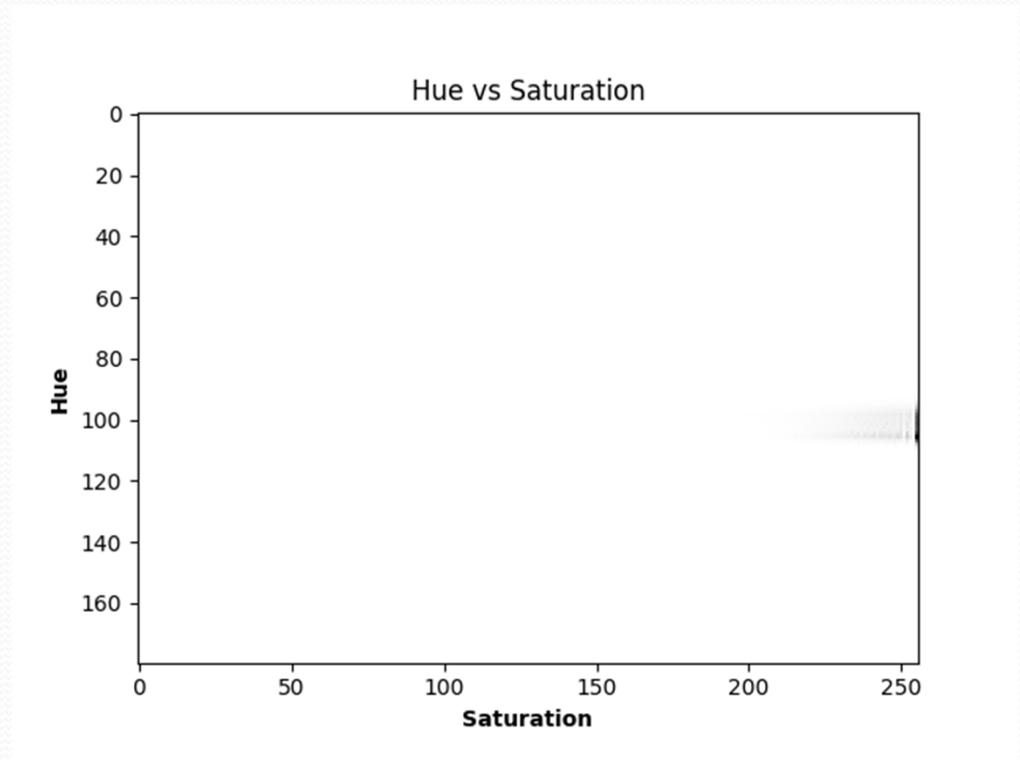
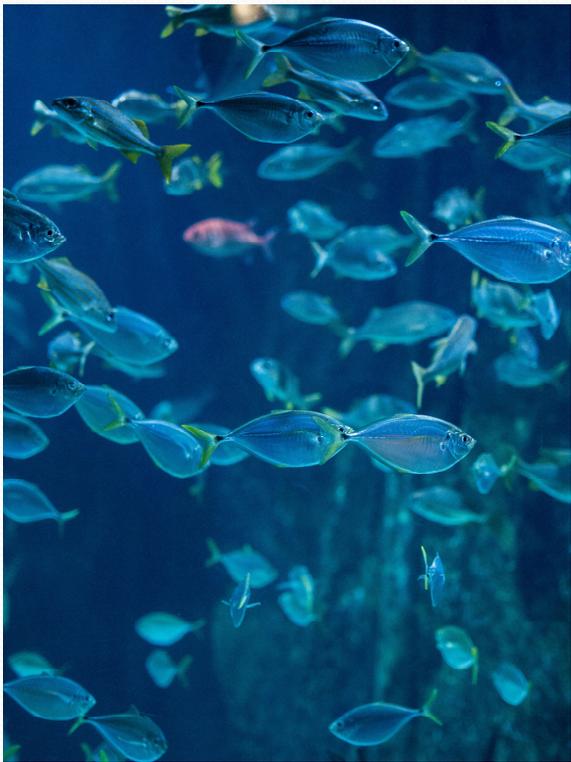
- Se usa la misma función `cv2.calcHist()` que se utiliza para los histogramas convencionales.

- Ejemplos:

```
hist_bg = cv2.calcHist([image_bgr], channels=[0, 1], mask=None,  
    histSize=[256, 256], ranges=[0, 256, 0, 256])
```

```
hist_hsv = cv2.calcHist([image_hsv], channels=[0, 1], mask=None,  
    histSize=[180, 256], ranges=[0, 180, 0, 256])
```

Ejemplo



Histogram Backprojection

- Responde a la pregunta: “¿Dónde están los colores en la imagen que pertenecen al objeto que se está buscando?”
- El algoritmo calcula el histograma del objeto y luego lo usa para hacer una proyección de vuelta.
- Lo siguiente está basado en <https://theailearner.com/2019/04/18/histogram-backprojection/>

Ejemplo



ROI



Target

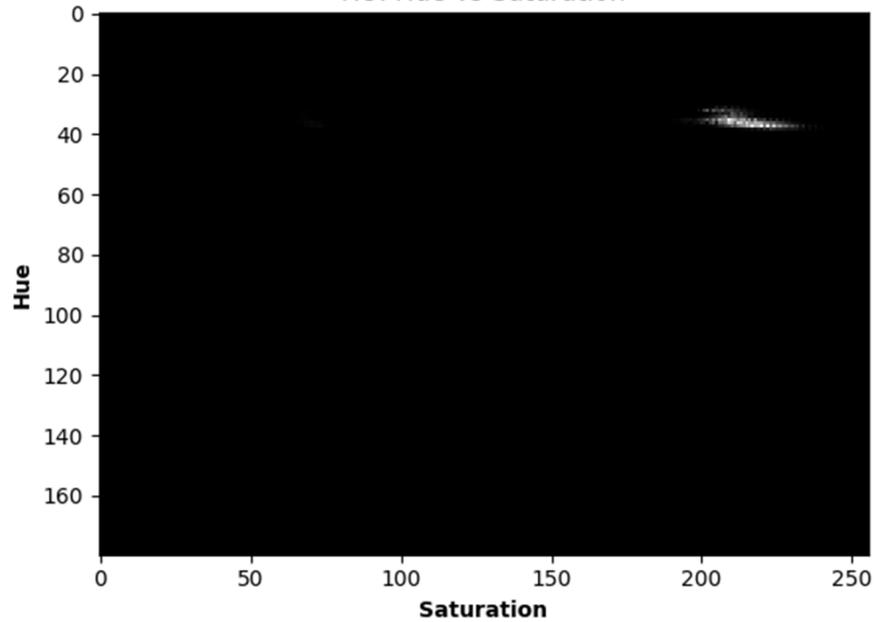


Primeros pasos

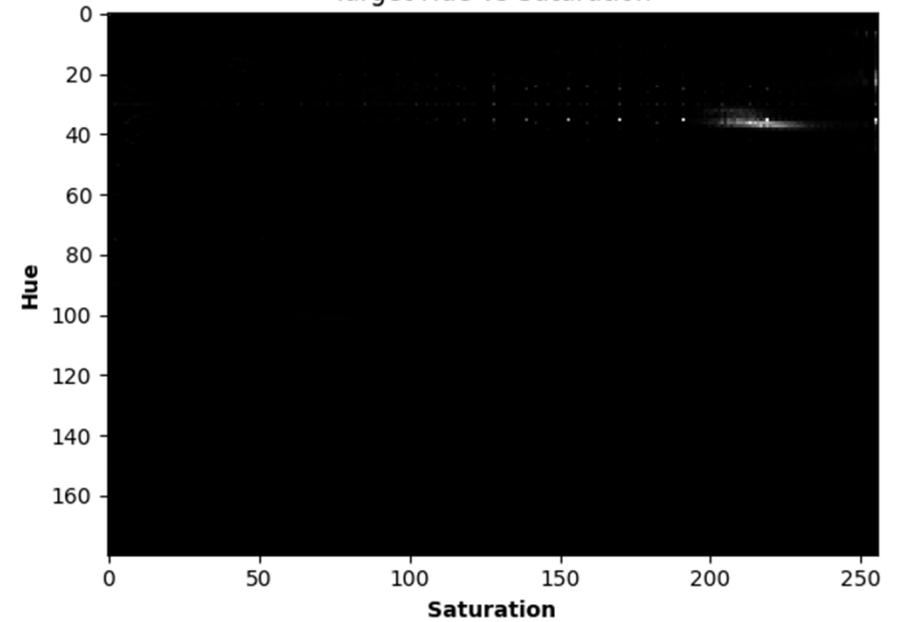
- Leer las imágenes.
- Obtener los histogramas 2D.

Histogramas 2D

ROI Hue vs Saturation



Target Hue vs Saturation



Explicación

- La imagen ROI tiene tonos verdes y su histograma está concentrado principalmente en los valores de H y S que representan el verde.
- Se puede decir lo mismo de la imagen destino.

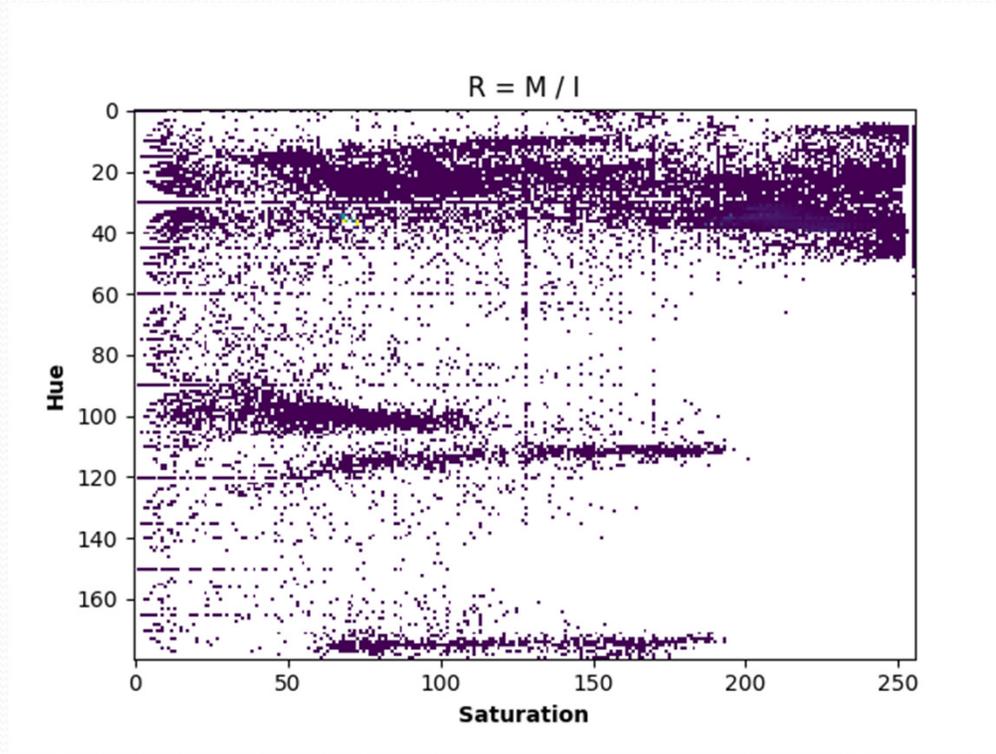
Proyección de regreso

- Se busca extraer los colores de la imagen ROI en la imagen destino.
- Una solución es dividir M (el histograma 2D de la imagen ROI) entre I (el histograma 2D de la imagen destino).

$$R = M / I$$

- Los valores de R son:
- Mayores que 0 si $M > 0$ e $I > 0$
- 0 o nan en otro caso.

Matriz R



Explicación

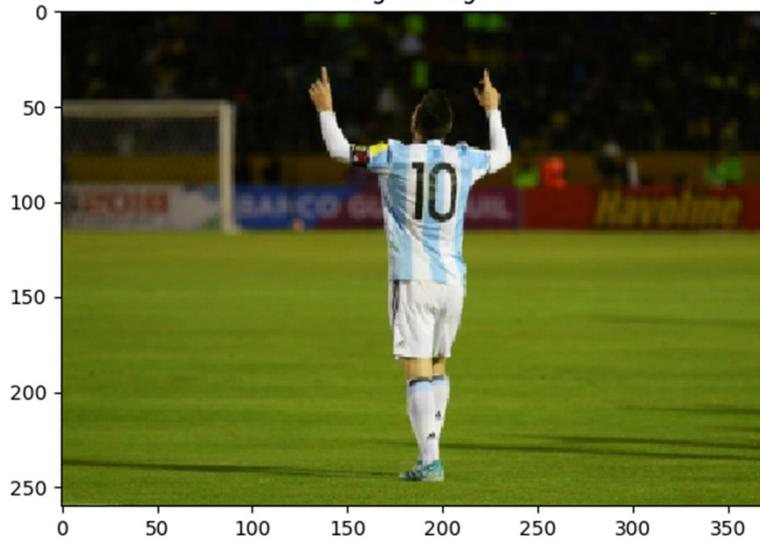
- El color cyan (alrededor de $H = 30$ y $190 \leq S \leq 200$) son valores mayores a 0.
- El color púrpura son valores iguales a 0.
- El color blanco son valores nan.

Proyección de regreso

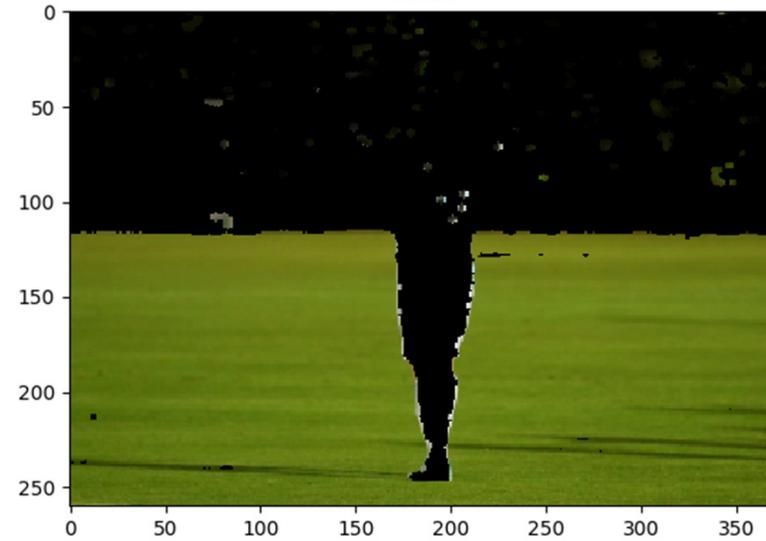
- Ahora R se usa como paleta y se proyecta de regreso.
- Es decir, $B(x, y) = R[h(x, y), s(x, y)]$
- Donde h es el hue y s la saturación del pixel en (x, y) en la imagen destino.
- B tiene valores distintos de 0 en la región de la imagen destino que corresponde a los colores de la imagen ROI.

Resultado

Target image



Result



Histogram backprojection in OpenCV

- Método `cv2.calcBackProject()`.

Resumen

- Técnicas para ampliar el contraste: autocontraste, corrección gamma, ecualización del histograma.
- En imágenes en color conviene separar la luma y la crominancia para evitar cambios de color no deseados.
- Al ampliar el contraste en una imagen en color es mejor operar en el canal de luma.