

Rodrigo Hernández Zavala

03 de mayo de 2025

Ing. Mecatrónica

22310215

Histograma

Visión Artificial

Este código realiza una práctica fundamental en procesamiento digital de imágenes: **ecualización del histograma**. Aquí te explico paso a paso:



¿Qué hace el código?

1. **Carga una imagen en escala de grises.**
 2. **Calcula el histograma de la imagen original.**
 3. **Aplica ecualización del histograma** para mejorar el contraste.
 4. **Calcula el nuevo histograma** (después de la ecualización).
 5. **Muestra en una sola ventana:**
 - Imagen original
 - Histograma original
 - Imagen ecualizada
 - Histograma ecualizado
-



Explicación por bloques:

1. Carga de imagen

```
python
CopiarEditar
img = cv2.imread('watch.jpg', cv2.IMREAD_GRAYSCALE)
```

- Lee la imagen 'watch.jpg' en **modo de escala de grises** (1 canal).
- Esto es necesario para trabajar con histogramas de brillo (no de color).
- Se verifica que se haya cargado correctamente con:

```
python
CopiarEditar
if img is None:
    ...
```

2. Histograma original

```
python
CopiarEditar
hist_orig = cv2.calcHist([img], [0], None, [256], [0, 256])
```

- Calcula el histograma de la imagen.
- [0]: canal de intensidad de gris.
- 256: número de bins (niveles de intensidad de 0 a 255).

- Devuelve una matriz de frecuencia de píxeles para cada nivel de intensidad.
-

3. Ecualización del histograma

```
python
CopiarEditar
img_ecualizada = cv2.equalizeHist(img)
```

- Esta función redistribuye los niveles de brillo para que la imagen tenga **mejor contraste**.
 - Muy útil para imágenes oscuras o con poco rango dinámico.
-

4. Histograma después de la ecualización

```
python
CopiarEditar
hist_eq = cv2.calcHist([img_ecualizada], [0], None, [256], [0, 256])
```

- Se repite el cálculo del histograma pero ahora para la imagen mejorada.
-

5. Visualización con Matplotlib

```
python
CopiarEditar
plt.subplot(...) ...
plt.imshow(...) ...
```

- Se muestran 4 gráficas:
 - **Imagen original** (superior izquierda).
 - **Histograma original** (superior derecha).
 - **Imagen ecualizada** (inferior izquierda).
 - **Histograma ecualizado** (inferior derecha).
 - `plt.tight_layout()` ajusta los elementos para que no se encimen.
-

¿Para qué sirve esta práctica?

- **Mejorar imágenes con poco contraste.**
- Visualizar cómo la **ecualización distribuye mejor los niveles de intensidad**.
- Es un paso esencial en **visión artificial**, por ejemplo para facilitar la detección de bordes o segmentación.