



Universidad Autónoma de San Luis Potosí  
Facultad de ingeniería  
Tratamiento de Imágenes



#### Practica 4

**Nombre Práctica:** Manipulaciones Básicas de Imágenes

**Nombre del Alumno:** Manuel Ramírez Galván

**Fecha:** 18/02/2025

## Procedimiento

3.1. Diseñe un programa en Python en el cual se cumplan los siguientes requisitos:

- Cargue una imagen del disco duro.
- Aplique dos operaciones de transformación a fin a la imagen
- La imagen debe de ser mostrada en la ventana creada con el nombre "Ejercicio 4.1".

## Resultados

```
1 import cv2
2
3 img1 = cv2.imread('Recursos\Hollow Knight.jpg', cv2.IMREAD_COLOR)
4
5 alto, ancho = img1.shape[:2]
6
7 tx, ty = ancho/2, alto/2
8
9 centro = (ancho/2, alto/2)
10 rot_matriz = cv2.getRotationMatrix2D(centro, 45, 1.0)
11
12 rot_img1 = cv2.warpAffine(img1, rot_matriz, (ancho, alto))
13
14 nuevo_ancho = 560
15 nuevo_alto = 540
16 nuevo_dsize = (nuevo_ancho, nuevo_alto)
17 img_rsize = cv2.resize(rot_img1, nuevo_dsize)
18
19 cv2.imshow("Ejercicio 4.1", img_rsize)
20 cv2.waitKey(0)
21 cv2.destroyAllWindows()
```

Imagen 1.- Código Ejercicio 1

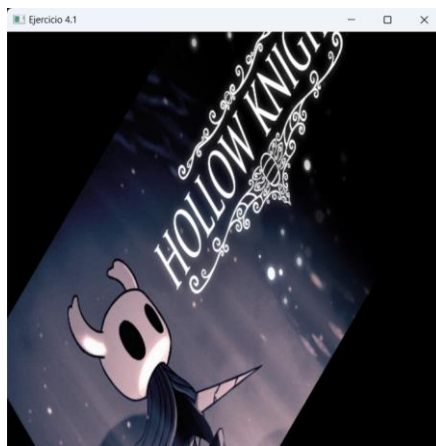


Imagen 2.- Imagen con Operaciones

## Procedimiento

3.2. Diseñe un programa en Python en el cual se cumplan los siguientes requisitos:

- Cargue una imagen del disco duro.
- Crear una ROI que abarque elementos de interés en la imagen.
- La imagen debe de ser mostrada en la ventana creada con el nombre "Ejercicio 4.2" hasta que se presione la tecla ESC.

## Resultados

```
1  import cv2
2
3  img2 = cv2.imread('Recursos\BOTW.jpg', cv2.IMREAD_COLOR)
4
5  escala = 0.7
6  ancho = int(img2.shape[1] * escala)
7  alto = int(img2.shape[0] * escala)
8  img2 = cv2.resize(img2, (ancho, alto), interpolation=cv2.INTER_AREA)
9
10 retval = cv2.selectROI("Imagen", img2)
11
12 img2 = img2[int(retval[1]):int(retval[1]+retval[3]), int(retval[0]):int(retval[0]+retval[2])]
13
14 mostrar = True
15 while mostrar:
16     cv2.imshow("Ejercicio 4.2", img2)
17     keypress = cv2.waitKey(1)
18     if keypress == 27:
19         mostrar = False
20 cv2.destroyAllWindows()
```

Imagen 3.- Código Ejercicio 2



Imagen 4.- Imagen Original e Imagen al que se le hizo ROI

## Procedimiento

3.3. Diseñe un programa en Python en el cual se cumplan los siguientes requisitos:

- Cree una ventana con dos trackbars, uno para modificar el contraste y otro para modificar el brillo de una imagen.
- Cargue una imagen del disco duro.
- Utilice la fórmula  $g(i,j) = \alpha \cdot f(i,j) + \beta$  para cambiar el contraste y brillo de la imagen, donde  $\alpha$  es contraste y  $\beta$  es brillo.

- La imagen debe de ser mostrada en la ventana creada con el nombre "Ejercicio 4.3".
- Las imágenes se mostrarán indefinidamente hasta que se presione una tecla.

## Resultados

```

1  import cv2
2
3  img3 = cv2.imread('Recursos\Hollow Knight.jpg', cv2.IMREAD_COLOR)
4
5  escala = 0.6
6  ancho = int(img3.shape[1] * escala)
7  alto = int(img3.shape[0] * escala)
8  img3 = cv2.resize(img3, (ancho, alto), interpolation=cv2.INTER_AREA)
9
10 window = 'Ejercicio 4.3'
11 contrast = 10
12 max_contrast = 100
13 brightness = 0
14 max_brightness = 100
15
16 def change_contrast(val):
17     global contrast
18     contrast = val/1000
19     perform_operation()
20
21 def change_brightness(val):
22     global brightness
23     brightness = val/100
24     perform_operation()
25
26 def perform_operation():
27     img = img3*contrast + brightness
28     cv2.imshow(window, img)
29
30 cv2.imshow(window, img3)
31 cv2.createTrackbar('Contrast', window, contrast, max_contrast, change_contrast)
32 cv2.createTrackbar('Brightness', window, brightness, max_brightness, change_brightness)
33 cv2.waitKey(0)

```

Imagen 5.- Código Ejercicio 3

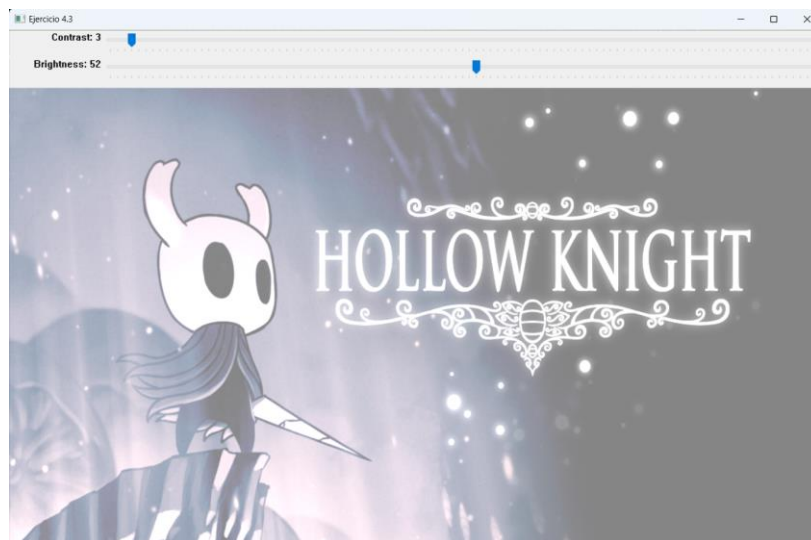


Imagen 6.- Imagen con Trackbars para Brillo y Contraste

## Comprensión

1. ¿Cuáles son las transformaciones afines y cuál es su característica más importante?

**Traslación:** Desplaza un objeto en el espacio sin cambiar su forma o tamaño

**Rotación:** Gira un objeto alrededor de un punto sin cambiar su tamaño

**Escalado:** Cambia el tamaño de un objeto

**Sesgado:** Deforma un objeto inclinando sus ejes

## **2. ¿En qué ayuda en escalamiento en las imágenes?**

Reduce el tamaño de la imagen para que ocupe menos memoria y procesamiento en aplicaciones.

Útil para que todas las imágenes tengan el mismo tamaño al ser comparadas o analizadas fácilmente.

Se adapta la imagen para distintos tamaños de pantalla.

Permite corregir imágenes que han sido capturadas en diferentes resoluciones o proporciones.

Mejoramiento de imágenes al escalarlas sin perder detalles

## **3. ¿Qué es la región de interés y para que se utiliza?**

Es una porción específica de una imagen que se selecciona para analizar o procesar ignorando el resto de la imagen.

Se utiliza para la identificación de objetos como rostros, matrículas, textos en imágenes.

Para reducir el tamaño de los datos a analizar y evitar procesar información irrelevante.

Al definir un ROI alrededor de un objeto y rastrear su movimiento.

Detención de áreas específicas en radiografías, tomografías y análisis de piel.

## **Conclusiones**

Existen manipulaciones básicas de imágenes, como traslación, rotación y escalado, que nos pueden ayudar a cambiar su posición, orientación y tamaño para algún propósito en específico como en el proyecto de la materia al sobreponer las imágenes.

La región de interés nos sirve para enfocar zonas específicas y solo trabajar en ellas evitando que se trabaje más y sea más tardado el trabajo que se quiera realizar como detección de zonas de interés, procesamiento de patrones, detección de movimiento, etc.