

Práctica 01

La biblioteca OpenCV

Visión Computacional Aplicada a la Robótica

UNAM, 2021-2

Resumen

El alumno aprenderá a abrir y manipular imágenes usando la biblioteca OpenCV, así como a hacer funciones personalizadas que permitan manipular las estructuras de datos que maneja OpenCV.

Duración:

Dos semanas

Desarrollo

1. Utilizando las funciones de OpenCV:
 - a) Crear una imagen con un cierto largo y ancho.
 - b) Cargar una imagen a partir de un archivo en disco.
 - c) Crear una copia de una imagen.
 - d) Separar los canales de color de una imagen.
 - e) Definir la región de interés de una imagen y copiarla a una imagen más pequeña.
 - f) Aplicar operaciones lógicas usando dos imágenes diferentes del mismo tamaño (AND, OR, NOT, XOR).
 - g) Convertir la imagen a escala de grises y HSV. Mostrar cada canal por separado en la imagen HSV.
2. Programando a mano:
 - a) Escribir una función que genere una imagen en espacio de color RGB normalizado a partir de una imagen en RGB.

Funciones de referencia

- numpy.zeros
- numpy.ones
- cv2.imread
- cv2.bitwise_and
- cv2.bitwise_or
- cv2.bitwise_not
- cv2.bitwise_xor
- cv2.cvtColor
- cv2.split
- cv2.imshow

Código fuente de referencia

```

1 import cv2
2 import numpy as np
3
4 def do_something(img):
5     result = np.zeros(img.shape, np.uint8)
6     [h,w,c] = img.shape
7     for i in range(h):
8         for j in range(w):
9             b,g,r = img[i,j]
10            sum = float(b) + g + r
11            if sum > 0:
12                result[i,j] = (255*b/sum, 255*g/sum, 255*r/sum)
13    return result
14
15 def main():
16    img_baboon = cv2.imread('baboon.jpg')
17    img_normalized = do_something(img_baboon)
18    cv2.imshow("Original BGR", img_baboon)
19    cv2.imshow("Normalized BGR", img_normalized)
20    cv2.waitKey(0)
21
22 if __name__ == '__main__':
23    main()

```