

Programación de computadores

Integrantes: Juan Camilo Rivera Claro

Mateo Rocha Padilla

Sergio Alejandro Meneses Galvis

Cristian Norbey Hernández Gualteros

Taller Grupal

Colab del código del taller:

<https://colab.research.google.com/drive/1CmQ-ZZ8d1M3cdZK4buGr3qfXMcRF1w6Z?usp=sharing>

Explicación

Primer punto

1. Cree una imagen de un canal, es decir con una forma (alto, ancho) del tamaño que desee, donde la diagonal principal hacia arriba sea blanca y desde la diagonal hacia abajo sea negra.

Iniciamos el código importando: random, numpy, Skimage y matplotlib

En la primera celda declaramos una variable "ancho =200"

Luego declaramos una lista vacía "imagen"

Luego declaramos otra variable "contador" iniciará en 0

Luego creamos un "for" que va a iterar de 0 a 200

Establecemos a "fila" esta línea nos ayudará para crear las filas de la matriz

Luego creamos otro "for" que nos servirá para iterar dentro de la matriz

Luego establecemos a "rojo" con random.randint de 0 a 225, esto nos dará unos valores aleatorios entre estos valores para asignar la intensidad

Luego ponemos un ejemplo usando tupla

Le preguntamos a la máquina que, si i es menor a J , y es que en algebra lineal cuando i es menor a j estas en la diagonal de arriba

Si la condición anterior se cumple entonces $\text{Pixel}=225$ esto quiere decir que estarán en la máxima intensidad y cuando los pixeles brillan al máximo, serán de color blanco

Establecemos a `fila.append(pixel)` y esto nos ayudará que en todas las filas de la matriz serán de color blanco arriba de la diagonal principal

Luego ponemos "else" que si no se cumple la condición Pixel=0 esto significa que los pixeles tendrán la mínima intensidad, y serán de color negro

Luego ponemos fila.append(pixel) y esto nos ayudará que en todas las filas de la matriz serán de color negro debajo de la diagonal principal.

Establecemos a contador +=1 y significa la cantidad de pixeles que va a haber en la matriz

Ponemos a imagen.append(fila) y es que en la lista llamada "imagen" va insertar la cantidad de filas que hay en la matriz y por esta razón la cantidad de columnas al estar en el segundo "for" ya que en el segundo "for" recorre toda la matriz en orden fila-columna

Creamos una variable que será "imagen_aleatoria" y esta será igual a "imagen" y acá se guardará como tal la matriz

Establecemos "plt.imshow(imagen_aleatoria)" que acá va mostrar la matriz

Y luego imprimimos "imagen_aleatoria"

Y luego imprimimos a contador que nos dirá la cantidad de pixeles que hay en la matriz.

2. Use una imagen de la librería skimage, en la base de datos "data". e implemente alguno de los siguientes procesos:([link](#))
 - Suavizar y resaltar contornos:
 - Restauración
 - Eliminación de ruido
 - Mejoramiento del contraste
 - Detección de bordes

En el segundo punto de la actividad escogimos selección de bordes

Iniciamos el código importando las librerías necesarias para hacer funcionar el código

Abrimos la imagen con "imagen" donde le daremos la ruta a la imagen

Y luego con "imagen_g" establecemos el color de la imagen

Luego creamos una lista que serán los filtros de la imagen, la lista será "filtros"

Creamos un "for" y se utiliza el iterador filtros y se recorre la lista filtros

Luego a nuestra imagen aplicamos los filtros a través de una lista

Luego con "plt.imshow" se muestra el resultado y como es la función

Y luego con "plt.show()" le decimos a la maquina que muestre la imagen

