Tarea 1: Aplicaciones del Reconocimiento de Patrones

Reconocimiento de Patrones - 0757

Facultad de Ingeniería  
Universidad Nacional Autónoma de México

Murrieta Villegas A.   
Ingeniería en Computación  
Facultad de Ingeniería, UNAM  
Ciudad de México, México  
alfonsomvmx@gmail.com

**Profesores:**  
Dr. Boris Escalante Ramírez  
Dra. Olveres Montiel Jimena  
I.I.M.A.S. - UNAM

Resumen— Esta práctica consiste en realizar manipulaciones básicas a imágenes utilizando bibliotecas de Python, como son Matplotlib, OpenCV, Scikit-Image, PIL y SciPy. Entre estas manipulaciones, se encuentran desplegar, cambiar el espacio de color, redimensionar, rotar y recortar imágenes, además de imprimir su información.

1. Introducción

A lo largo de esta práctica, se leerán, desplegarán y manipularán imágenes utilizando scripts codificados en Python. Una imagen es una matriz, o arreglo, de píxeles cuadrados (elementos de imagen) dispuestos en columnas y filas. En una imagen en escala de grises de 8 bits, cada elemento de la imagen tiene una intensidad asignada que va de 0 a 255 . Las imágenes en escala de grises contienen sólo un canal, aunque otro tipo de imágenes, como las imágenes a color RGB, contienen más.

Las imágenes que se utilizarán son de diferentes resoluciones, formatos y tipos; entre estas imágenes, hay imágenes de 512\*512, 800\*600 y 300\*209 píxeles, imágenes con formato TIFF, RAW y JPG e imágenes en escala de grises y RGB.Desarrollo

La práctica consiste en 5 ejercicios, algunos de los cuales consisten en varias actividades. En todos los casos, se manipulan distintas imágenes con diferentes formatos y resoluciones.

Algunos sitios consultados como apoyo en el desarrollo de la práctica fueron , , y .

1. 4.1 Carpeta de imágenes

Para la primera actividad se desarrolló un script de Python para leer y mostrar las diferentes imágenes a través de diferentes bibliotecas (Matplotlib, OpenCV, Sikit-image, PIL y Scipy), a continuación, se muestran los resultados obtenidos.

Imágenes que originalmente estaban en escala de grises:



Fig. 1 *cameraman.tif*.

1. Conclusiones

En la presente práctica aprendimos a utilizar clases y métodos de distintas bibliotecas de Python que tienen como propósito principal el procesamiento de imágenes. A su vez aprendimos a trabajar con distintos tipos de imágenes desde formatos más tradicionales como ***.tif*** hasta imágenes de formatos crudos como “***.raw*”** o de ambientes médicos “***.dcm*”**, también aprendimos a manejar estas mismas imágenes en distintos espacios de color que están destinados a entornos concretos es el caso del ***HSV*** o ***YUV***.

Por último, aprendimos a realizar distintas transformaciones con las imágenes, desde aspectos básicos como la rotación hasta aspectos más complejos como el escalamiento donde debía contemplarse el muestreo y otros aspectos relacionados con la interpolación o con la “*decimation*”.

Referencias

|  |  |
| --- | --- |
| [1] | Hubble Space Telescope, “Introduction to image processing”, esahubble.org, s. f. [En línea]. Disponible en: https://esahubble.org/static/projects/fits\_liberator/image\_processing.pdf. [Consultado el 6 de Septiembre del 2021]. |
| [2] | A. Ybodon, “[Python In-depth] Image handling in Python with OpenCV (2)”, Medium, 2019. [En línea]. Disponible en: https://financial-engineering.medium.com/python-in-depth-image-handling-in-python-with-opencv-2-5260e56c186f. [Consultado el 4 de Septiembre del 2021]. |
| [3] | A. P. Chazhoor, “Image processing using scikit image”, Towards Data Science, 2019. [En línea]. Disponible en: https://towardsdatascience.com/image-processing-using-scikit-image-cb57ce4321ed. [Consultado el 4 de Septiembre del 2021]. |
| [4] | R. Khandelwal, “Loading and Saving Images in Python”, Towards Data Science, 2020. [En línea]. Disponible en: https://towardsdatascience.com/loading-and-saving-images-in-python-ba5a1f5058fb. [Consultado el 4 de Septiembre del 2021]. |