

Curso práctico multiplataforma de Tratamiento Digital de la Imagen con Jupyter

Ana Cuevas Bravo

December 9, 2020

Resumen

Your abstract goes here... ...

Contents

1	Introducción	1
2	Objetivos	2
2.1	Objetivos	2
2.2	Requisitos	2
2.3	Metodología	2
3	Infraestructura	3
3.1	Python y tratamiento de imagen	3
3.1.1	OpenCV	4
3.1.2	Numpy	4
3.1.3	Matplotlib	4
3.1.4	Scikit-learn	4
3.2	Jupyter	4
3.3	Matlab	4
3.4	Estructura de las prácticas de TDI	4
4	Prácticas de TDI	8
5	Conclusiones	9

Capítulo 1

Introducción

Capítulo 2

Objetivos

2.1 Objetivos

2.2 Requisitos

2.3 Metodología

Capítulo 3

Infraestructura

En esta sección se detallan tanto el software empleado para la realización del trabajo, como la estructura previa que existía en la asignatura y que ha servido como base para las nuevas prácticas.

El lenguaje de programación utilizado ha sido Python, profundizaremos en las bibliotecas que tiene este lenguaje para el tratamiento de imagen que han sido necesarias para el desarrollo del trabajo. Esta sección también explicará en que consiste la plataforma Jupyter notebook y las ventajas que tiene en su uso para la docencia. Por último, se hablará de Matlab, la plataforma original que se usa en la asignatura de Tratamiento Digital de la Imagen y la estructura general que tienen las prácticas de esta asignatura.

3.1 Python y tratamiento de imagen

Python es un lenguaje de programación interpretado de alto nivel orientado a objetos creado en los años 80 por Guido van Rossum. Se caracteriza por tener una sintaxis sencilla, fácil de leer, lo que facilita el mantenimiento de programas y lo hace accesible para principiantes. Tiene una amplia biblioteca base y además permite el uso de módulos y paquetes. Al ser interpretado en lugar de compilado el proceso de debugging es más rápido.

Python se está usando cada día más para el tratamiento de imagen, esto se debe a que es un lenguaje muy accesible, siendo gratuito y con una sintaxis sencilla. Con el tiempo han ido surgiendo bibliotecas específicas para el tratamiento de la imagen en Python como Pil (Python Imaging Library) también llamada Pillow, Scikit-image y Open-CV, esta última será la que más se use en este proyecto.

Otras bibliotecas que facilitan el tratamiento de imagen en Python son Numpy, una biblioteca para facilitar el uso de matrices y arrays en Python así como las funciones matemáticas relacionadas y Matplotlib que permite la representación de datos en forma de gráficos.

3.1.1 OpenCV

Como ya hemos mencionado antes, OpenCV(Open Source Computer Vision Library) es una biblioteca centrada en el tratamiento de imagen y video (Computer Vision) y en aprendizaje de máquina, con interfaces para varios lenguajes de programación como C++, Python o Java. OpenCV es la biblioteca de procesamiento de imagen más usada en el mundo con más de 18 millones de descargas. Originalmente programada en C++

OpenCV contiene desde funciones de bajo nivel para el procesado de imagen hasta algoritmos complejos para detección de caras. En este trabajo, ya que trata de dar una base de tratamiento de imagen, usaremos funciones de bajo nivel como filtros o funciones para cambiar de espacio de color.

3.1.2 Numpy

3.1.3 Matplotlib

3.1.4 Scikit-learn

3.2 Jupyter

3.3 Matlab

3.4 Estructura de las prácticas de TDI

Las prácticas originales para la asignatura de Tratamiento Digital de la Imagen consisten en una carpeta zip que se ponía a disposición de los alumnos en el aula virtual de la asignatura. Dentro de esta carpeta están las imágenes que se van a utilizar en la práctica (cuando no eran imágenes internas de MatLab), funciones adicionales necesarias en caso de que no existieran pre-

viamente y un enunciado en pdf. En la imagen 3.1 se puede ver un ejemplo de una de estas carpetas.

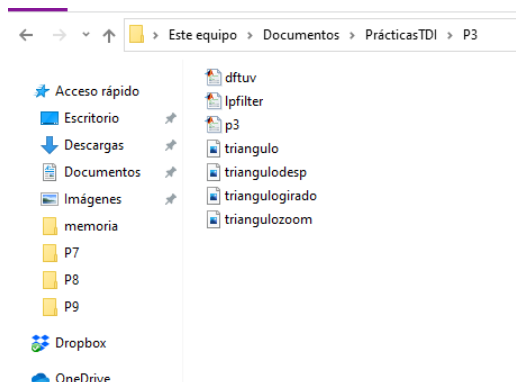


Figure 3.1: ejemplo de carpeta de prácticas

Las prácticas se realizaban en hora de clase en uno de los laboratorios de Windows del campus. Se podían realizar enteras en la duración de la clase.

La profesora daba una breve explicación inicial sobre el tema de la práctica, siempre relacionado con lo que se hubiera dado en las clases de teoría recientes, el resto de la práctica se podía seguir de forma libre con el enunciado, con la profesora respondiendo las dudas que surgieran en el desarrollo del ejercicio.

Los enunciados son instrucciones detalladas sobre los pasos a seguir para realizar la práctica, con poco o nada de teoría ya que se asume que el alumno ha recibido las clases de teoría previas. El alumno debe crear un nuevo documento en Matlab e ir ejecutando paso a paso lo que pide el enunciado.

Los enunciados suelen comenzar con un breve párrafo explicando el objetivo de la práctica, por ejemplo en la práctica 4 que trata el tema de segmentación de Imagen la introducción dice: "El objetivo de esta práctica es comenzar a familiarizar al alumno con las herramientas básicas de segmentación de imagen en entorno MATLAB. Para ello se trabajará con la imagen en escala de grises 'calculadora.tif', que acompaña al material de esta práctica. "

A partir de esta breve introducción la práctica suele estar dividida en secciones normalmente relacionadas con diferentes puntos tratados en la teoría y como llevarlos a cabo.

Los pasos a seguir en MatLab, por lo general, vienen dados en el enunciado que indica qué función usar y cómo llamar a las diferentes variables para mantener una consistencia en los nombres a lo largo de la práctica. En algunos casos se indica las variables necesarias para llamar a la función que se va a usar en esa sección del ejercicio, mientras que en otros se pide al alumno que use el comando *+help* para informarse sobre el funcionamiento de la función. Del mismo modo si, una función requiere de un valor a decidir, dependiendo de lo que se quiera conseguir con su uso, unas veces se da con el enunciado y otras se deja a criterio del alumno para que pruebe como cambian los resultados con diferentes variables y cuál sería la mejor solución para el problema planteado.

Otra característica común de los enunciados de TDI son las preguntas para responder, aunque no se pide una memoria en sí de las prácticas en los enunciados hay preguntas con el objetivo de que el alumno se plantee las respuestas y las razones de los pasos que se han ido realizando durante el ejercicio. La siguiente imagen 3.2 muestra un ejemplo de estas preguntas del principio de la práctica 4.

- ¿son todas las teclas del mismo tamaño?, ¿cuál es la de mayor tamaño?
- ¿tienen todas las teclas alguna letra/número en su interior?, ¿existe conectividad entre ellas?
- ¿aparecen letras/números en el exterior de las teclas?
- ¿qué letras/números presentan mayor intensidad luminosa, los del interior de las teclas o los del exterior?

Figure 3.2: ejemplo de preguntas realizadas en los enunciados de TDI

Algunos de los enunciados tienen imágenes del aspecto que tendría que tener la imagen sobre la que se está trabajando después de ciertos pasos para que el alumno pueda ver si el ejercicio progresa adecuadamente o si debería modificar alguna variable o revisar el código.

Si se pide un algoritmo un poco más complejo en los enunciados aparece dado un trozo de ese código como ejemplo o directamente para copiar en matlab ya que el objetivo de estas prácticas no era aprender a programar en MatLab si no ilustrar de forma práctica lo dado en teoría de la asignatura.

En total la asignatura consiste de nueve prácticas, 7 de imagen y 2 de video.

Capítulo 4

Prácticas de TDI

Capítulo 5

Conclusiones