



DESARROLLO DE ALGORITMOS CON FICHEROS

Evaluación

UCO
ONLINE

Desarrollo de algoritmos con ficheros

Objetivos

Desarrollar un programa en Python diseñado con orientación a objetos que haga uso de ficheros de texto y binario

Temporización

180 minutos

Enunciado

Descripción

Una imagen digital es una matriz bidimensional de puntos (pixels) en la que cada uno tiene asociado un nivel de luminosidad cuyos valores están en el conjunto $\{0, 1, \dots, 255\}$ de forma que el 0 indica la menor luminosidad (negro) y el 255 la mayor luminosidad (blanco). Los restantes valores indican niveles intermedios de luminosidad (grises), siendo más oscuros cuanto menor sea su valor. Con esta representación cada píxel requiere únicamente un byte.

Cada casilla de esta matriz representa a un punto de la imagen y el valor guardado en ésta indica

- En imágenes de niveles de gris: su nivel de luminosidad.
- En imágenes en color: su código de color (representación por tabla de color) o la representación del color en el esquema usado (RGB, IHV, etc).

Nuestro interés se centra únicamente en imágenes de niveles de gris, por lo que cada casilla contiene niveles de luminosidad que se representan con valores del conjunto $\{0, 1, \dots, 255\}$ con la convención explicada anteriormente.

Las imágenes se almacenan en ficheros con un determinado formato. Nosotros vamos a trabajar con imágenes en formato PGM.

Objetivo

Implementar un programa que permita trabajar con imágenes en formato PGM. El programa deberá definir una clase Imagen y:

- Cargar en memoria una imagen PGM
- Almacenar con formato PGM una imagen que está en memoria
- Calcular el negativo de una imagen
- Calcular el espejo horizontal de una imagen

Descripción del formato PGM

El formato PGM constituye un ejemplo de los llamados formatos con cabecera. Un fichero PGM tiene, desde el punto de vista del programador, un formato mixto texto-binario: la cabecera se almacena en formato texto y la imagen en sí en formato binario.

Con más detalle, la descripción del formato PGM es:

1. Cabecera. La cabecera está en formato texto y consta de:
 - Un "número mágico" para identificar el tipo de fichero. Un fichero PGM que contiene una imagen de niveles de gris (nuestro caso) tiene asignado como identificador los caracteres P5.
 - Un número indeterminado de comentarios (marcados con #).
 - Número de columnas (c) y número de filas (f), separados por un número indeterminado de espacios.
 - Valor del mayor nivel de gris que puede tener la imagen (m).
 - Cada una de estas informaciones está terminada, por un salto de línea.
2. Contenido de la imagen
 - Una secuencia binaria de $f \times c$ bytes, con valores entre 0 y m. Cada uno de estos valores representa el nivel de gris de un píxel. La imagen se almacena por filas, es decir, el primer píxel es la esquina superior izquierda, el segundo el que está a su derecha, etc.

Algunas aclaraciones respecto a este formato

- En Python, los ficheros que tienen mezcla de texto y binario se abren en binario y la parte de texto se lee en binario y se decodifica con el encode.
- Los comentarios son de línea completa y están precedidos por el carácter #. La longitud máxima es de 70 caracteres.

- Aunque el mayor nivel de gris sea m , no tiene porqué haber algún píxel con este valor.

En la Figura 1 se muestra, a modo de ejemplo, como se almacena una imagen de 4 filas y 3 columnas en este formato:

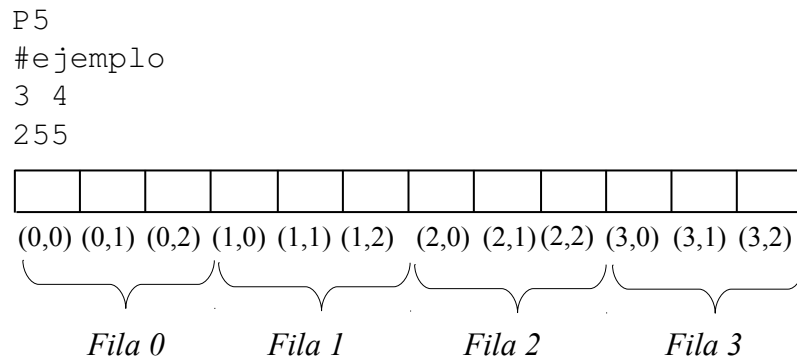


Figura 1: Formato imagen PGM

Descripción de los métodos a implementar

Los siguientes métodos son los básicos que hay que implementar. Además, se podrá implementar cualquier otra función y/o método que se crea necesario

Cargar una imagen en formato PGM desde un fichero

Dado un fichero que contiene una imagen digital en el formato PGM anteriormente descrito, el método deberá leer la cabecera del fichero, para obtener las características de la imagen (número de columnas, número de filas y niveles de gris) y, con estos datos, leer los píxeles de la imagen y guardarlos en un objeto Imagen.

Guardar una imagen en un fichero con formato PGM

Este método deberá crear un fichero que contenga la imagen almacenada en un objeto Imagen. El fichero deberá tener formato PGM. Para ello, en primer lugar habrá que grabar la cabecera del fichero en formato texto, de acuerdo a la especificación dada, y, a continuación, la imagen en formato binario.

Negativo

Se trata de una operación píxel a píxel en la que se genera una nueva imagen similar al negativo de una fotografía. Para ello:

$$\text{valorPixelNegativo} = 255 - \text{valorPixel}$$

*Imagen original**Negativo*

Espejo Horizontal

Consiste en simular el efecto de colocar un espejo sobre el eje vertical de la imagen, de forma que la primera columna de la imagen pase a ser la última columna de la imagen resultado, la segunda pasa a ser la penúltima, y así sucesivamente.

*Imagen original**Imagen reflejado horizontalmente*

Invocación del script

El programa recibirá 2 parámetros:

- Imagen con la que se va a trabajar
- Operación a realizar: n (negativo), h (espejo horizontal)

y creará un fichero con la imagen generada. El nombre del fichero será:

- ficheroOriginal_negativo.pgm si se elige el negativo
- ficheroOriginal_espejo.pgm si se elige espejo horizontal

Ejemplos

```
python3 procesaimagen.py vacas.pgm n
```

Salida: fichero vacas_negativo.pgm con la imagen en negativo

NOTA

Para simplificar la cabecera de los ficheros PGM, los ejemplos que se dejan para probar el programa, no contienen ningún comentario (no es necesario comprobar si existen y leerlos). No obstante, se debe tener en cuenta que si se utilizan otros ficheros de imágenes, éstas pueden presentar comentarios en la cabecera.