

ABSTRACCIÓN

T.D.A. IMAGEN

Estructura de Datos

8.Noviembre.2020

Alberto Llamas González

Elena Ortega Contreras




IMAGEN ORIGINAL	IMAGEN UMBRALIZADA	ICONO IMAGEN
		
Ej. imagen utilizada: cameraman.pgm	Resultado de la función umbralizar sobre imagen cameraman.pgm	Resultado de la función ObtenerIcono sobre imagen cameraman.pgm

TABLA DE CONTENIDO

Contenido

Clase imagen	1
Ejercicios elegidos	2
EJERCICIO 1 → Umbralizar una imagen usando una escala de grises	2
EJERCICIO 4 → Crear un icono a partir de una imagen	2
EJERCICIO 6 → Efectos especiales: simulación de morphing	2
Código	3
Ejemplo de uso:	4
EJERCICIO 2 → Umbralizar una imagen usando una escala de grises	4
EJERCICIO 4 → Crear un icono a partir de una imagen	4
EJERCICIO 6 → Efectos especiales: simulación de morphing	4

Clase imagen

Podemos consultar toda la documentación del T.D.A. Imagen ejecutando:

>\$ make documentacion (generación de la documentación)

Una instancia del tipo de datos abstracto Imagen es un objeto que representa una matriz 2D donde cada elemento corresponde al pixel de una imagen, que representa un valor de luminosidad entre 0 y 255.

Trataremos con imágenes de niveles de gris, que serán de extensión *.pgm*

- Disponemos de los ficheros **imagenES.h** e **imagenES.cpp**, con los que trataremos la entrada y salida de imágenes en el formato PGM.
- Hemos creado los ficheros:
 - **imagen.h** e **imagen.cpp** donde realizamos la implementación de la clase imagen.
 - **usoimagen.cpp** ponemos en práctica las funciones implementadas para el tratamiento de los objetos de tipo imagen.
 - **makefile** para automatizar la compilación de los ficheros, donde además añadiremos la generación de documentación que realizaremos con la herramienta *doxygen*.
 - **Doxyfile** para la documentación con *doxygen*

Ejercicios elegidos

EJERCICIO 1 → UMBRALIZAR UNA IMAGEN USANDO UNA ESCALA DE GRISES

- Consiste en generar a partir de una imagen original, otra imagen con el criterio de que si un pixel de la imagen original tiene un nivel de gris p comprendido en un intervalo definido por 2 umbrales T_1 y T_2 , se deja con ese nivel de gris p y en otro caso, se pone a blanco (nivel de gris 255).

EJERCICIO 4 → CREAR UN ICONO A PARTIR DE UNA IMAGEN

- Consiste en crear una imagen de un tamaño muy reducido a partir de una imagen original. El algoritmo de reducción consiste básicamente tomar cada pixel de la salida como la media de los $n \times n$ píxeles de la entrada si se quiere hacer una reducción de $n \times n$.

EJERCICIO 6 → EFECTOS ESPECIALES: SIMULACIÓN DE MORPHING

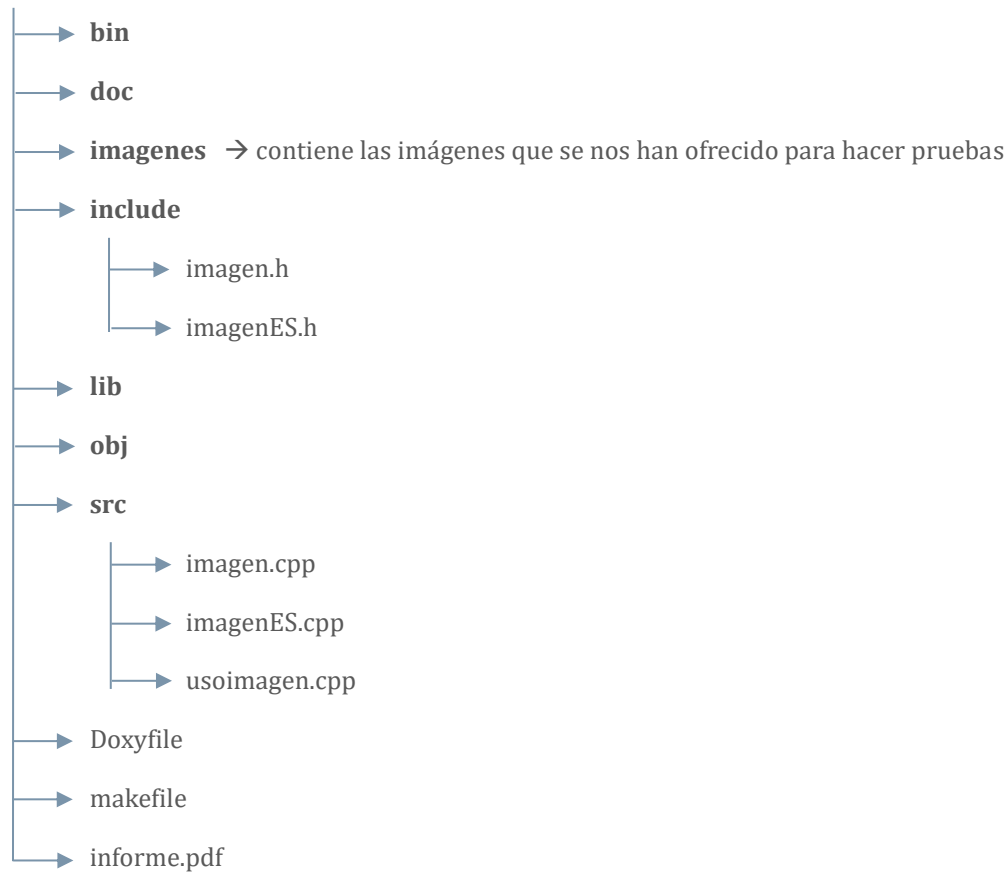
- El *morphing* se usa para cambiar una imagen en otra o para proporcionar una transición suave de una imagen a otra creando la ilusión de una transformación.

Tanto la cabecera como la implementación de estas funciones se incluyen en el fichero `usoimagen.cpp`.

Código

Esquema de los ficheros incluidos en el T.D.A. Imagen.

TDAimagen



Ejemplo de uso:

Situados en el directorio TDAimagen, mostramos un ejemplo de uso para cada función de los ejercicios:

>\$ make (compilación del proyecto)

EJERCICIO 2 → UMBRALIZAR UNA IMAGEN USANDO UNA ESCALA DE GRISES

Crearemos una nueva imagen resultado de umbralizar una imagen (cameraman.pgm):

> \$ bin/usoimagen umbralizar imagenes/cameraman.pgm imagenes/cameramanumbralizar.pgm

> Aplicando funcion umbralizar.

se genera una nueva imagen cameramanumbralizar.pgm, resultado de umbralizar la imagen original .

EJERCICIO 4 → CREAR UN ICONO A PARTIR DE UNA IMAGEN

Crearemos un icono de 128x128 a partir de una imagen (cameraman.pgm) de 256x256:

> \$ bin/usoimagen icono imagenes/cameraman.pgm imagenes/cameramanicono.pgm

> Introduzca el valor del icono a formar de la forma (filas, columnas):

128

128

se genera el icono cameramanicono.pgm, con una altura de 128 píxeles y una anchura de 128 píxeles.

EJERCICIO 6 → EFECTOS ESPECIALES: SIMULACIÓN DE MORPHING

Crearemos la ilusión de una transformación de una imagen a otra (de cameraman.pgm a vacas.pgm):

> \$ bin/usoimagen morphing imagenes/cameraman.pgm imagenes/vacas.pgm

> Introduzca el número de pasos intermedios en el morphing:

6

se generan los ficheros intermedios: intermedio1.pgm, intermedio2.pgm, ... , intermedio6.pgm que muestran la transición de la imagen cameraman a vacas.pgm (en 6 pasos)