



Centro de Investigación en Matemáticas, A.C.

Reporte tarea 1 - Programación y algoritmos

Erick Salvador Alvarez Valencia

16 de Agosto de 2017

1. Introducción

Este reporte presenta el desarrollo y los resultados en torno a las ejecuciones realizadas al programa entregado para la tarea 1. Se describirá a grandes rasgos la forma en que se programó el algoritmo, posteriormente se brindarán datos sobre las ejecuciones hechas con diferentes imágenes, y finalmente, se darán algunas conclusiones.

2. Desarrollo del programa

La elaboración del programa recayó en unas cuantas tareas: Lectura de la imagen, elaboración del histograma, obtención de la moda, y finalmente crear las imágenes de salida.

Primero se lee la imagen de entrada que está en formato PGM y creamos unas instancias de las imágenes de salida, para lo cual se usan objetos tipo *FILE*. Cabe destacar que se usa una matriz con memoria dinámica para almacenar los datos de la imagen ya que esta misma puede ser muy grande y podemos obtener un error por reservar bastante memoria estática.

Para crear el histograma se crea un arreglo de tamaño 256 y por cada pixel de la imagen, se suma 1 al arreglo en la posición del pixel. La moda se crea en base al histograma, solo se recorre el arreglo que representa al mismo y se busca el valor más grande, pero debido a que pueden existir varias modas se emplea la estrategia de un segundo recorrido al arreglo, solo que esta vez ya conocemos el mayor valor del mismo, y por lo tanto se buscan los pixeles asociados al valor y se saca la media aritmética con ellos.

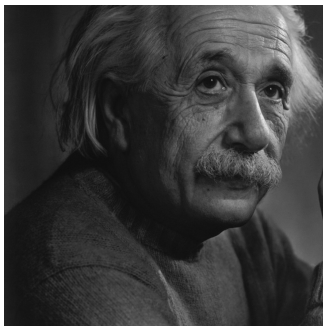
La imagen binaria se genera conociendo la moda, solo hay que iterar la matriz dinámica generada anteriormente y evaluar cada pixel con la moda, si es menor que la misma se coloca un pixel

negro, en caso contrario, se coloca uno blanco.

Finalmente se debe generar la imagen del histograma, para ello se crea una matriz estática de un tamaño fijo, en el cual nos aseguramos que la anchura sea mayor o igual a 256, esto para abarcar todo el rango de colores. Posteriormente en dos ciclos se realiza el llenado de la matriz con los datos del histograma obtenidos anteriormente. Cabe destacar que el recorrido de llenado se realiza de izquierda a derecha (anchura) y de abajo hacia arriba (altura).

3. Pruebas realizadas

El programa se probó con varias imágenes (incluida la adjunta con la tarea) de varios tamaños, todas en formato PGM ASCII. A continuación, se mostrarán los resultados de 2 imágenes que procesó el programa, así como su tiempo de ejecución.



(a) Imagen 1 (entrada)



(b) Imagen 1 (binaria)

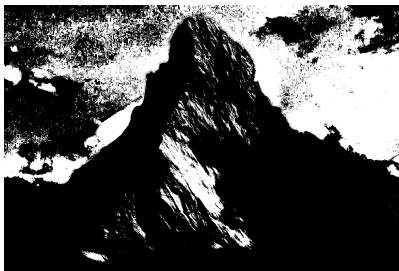


(c) Histograma de la imagen 1

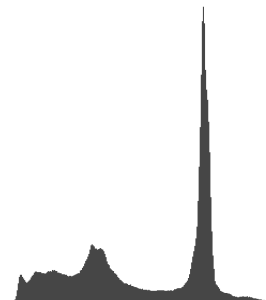
Para esta ejecución el programa tardó un tiempo de 0.209 s en concluir.



(d) Imagen 2 (entrada)



(e) Imagen 2 (binaria)



(f) Histograma de la imagen 2

Para esta ejecución el programa tardó 0.215 s en concluir.

Como se puede observar, el programa creó los modelos correctos de las imágenes binarias, y así mismo, se usó la siguiente página para comparar los histogramas generados por el programa: <http://www169.lunapic.com/editor/?action=histogram>. En la cual se verificó que dichos histogramas son correctos.

En general, después de hacer las pruebas con todas las imágenes, se obtuvo una media de 0.303 s. lo cual muestra que el programa se funciona a una buena velocidad, destacando que una de las imágenes procesadas es de tamaño 3478 x 2310 px y por lo cual requiere más tiempo de ejecución.

4. Conclusiones

La realización de este programa fue bastante intuitiva y no hay mucho que proponer como mejora, aunque cabe destacar que para almacenar la información de la imagen es mejor usar una matriz (o arreglo) con memoria dinámica que estática ya que dicha imagen puede ser bastante grande y el compilador GCC no nos deja reservar tanta memoria estática, lo cual puede producir un segmentation fault.

Otra mejora a considerar es la posibilidad que el programa procese comentarios ya que no lo hace actualmente, y muchos editores de imágenes como GIMP escriben comentarios en las mismas al convertirlas o editarlas.