

Proyecto Individual - Desarrollo de una aplicación para la generación de gráficos y texto

Gustavo Gamboa-Mora

2907@estudiantec.cr

Área académica de Ingeniería en Computadores

Instituto Tecnológico de Costa Rica

I. REQUERIMIENTOS DEL SISTEMA

Para la realización de este proyecto se enlistaron los siguientes requerimientos:

- Procesamiento de imágenes en lenguaje de bajo nivel (ensamblador), para esto se debe escoger ya sea ARM, RISC-V o X86. Es decir, la interpolación bilineal requerida debe realizarse en ensamblador, no es permitido realizarlo en un lenguaje de alto nivel.
- La imagen de entrada debe tener una dimensión mínima de 390x390 para este proyecto.
- Se debe realizar en escala de grises (píxeles de 0 a 255).
- Se debe permitir al usuario escoger uno de los 16 cuadrantes de la imagen original para realizarle la interpolación.
- El programa en ensamblador debe generar un archivo de salida, con formato .img, que contenga la imagen interpolada.
- El archivo de salida con formato .img debe ser visualizado en un software de alto nivel.
- Se debe generar un framework que realice todo el proceso, (escogencia de cuadrante, interpolación y visualización de cuadrante e imagen interpolada).

II. OPCIONES DE SOLUCIÓN

II-A. Implementación Secuencial Directa en Ensamblador

Se desarrolla un algoritmo en ensamblador que sigue de forma directa los pasos de la interpolación bilineal. Dividiéndose en:

- Carga y lectura de la imagen.
- Selección de cuadrante.
- Cálculo de interpolación.
- Escritura del archivo de salida.

En la siguiente imagen se muestra un diagrama de bloques de la solución.

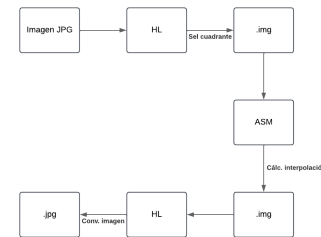


Figura 1. Diagrama de bloques propuesta 1

Como ventajas se puede decir que tiene una simplicidad y un control completo del problema a solucionar.

Como desventajas el rendimiento, ya que puede ser lento a la hora de generar la interpolación si se utilizan imágenes grandes.

II-B. Implementación Optimizada con Paralelismo e Instrucciones SIMD

Esta solución tiene bastantes similitudes con la primera solución pero se enfoca en aprovechar las capacidades de procesamiento paralelo del ISA a escoger. Esta solución tiene la siguiente estructura.

- Carga y lectura de la imagen.
- Selección de cuadrante.
- Cálculo de interpolación de forma paralela subdividiendo el cuadrante en bloques y utilizar instrucciones SIMD para procesar múltiples operaciones.
- Sincronización y reensamblado.
- Escritura del archivo de salida.

Como ventajas se tiene un rendimiento superior y mayor escalabilidad, mientras que sus desventajas se tiene una mayor complejidad y problemas con la compatibilidad.

II-C. Comparación de Soluciones

Realizando una comparación entre ambas soluciones se puede ver una mayor simplicidad en la propuesta 1 además de tener una mayor amplitud para la utilización de cualquier ISA ya que en la propuesta 2 las instrucciones SIMD dependerán de la arquitectura específica y las instrucciones SIMD que posea, además que varios simuladores no soportan este tipo de instrucciones.

Además de esto para el proyecto específico y las dimensiones mínimas requeridas para que el proyecto funcione ambas soluciones son válidas.

III. PROPUESTA FINAL

Para el proyecto como propuesta final se escoge la propuesta 1, ya que es simple y se adapta fácilmente a los requerimientos que el proyecto demanda, además de esto es fácil de verificar a la hora de defenderlo.

A continuación se muestra nuevamente el diagrama de bloques de la propuesta escogida.

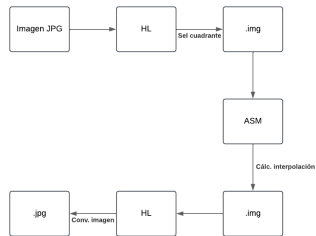


Diagrama de bloques propuesta final

Donde en un solo framework se obtiene el cuadrante, se llama al ensamblador para que realice la interpolación y muestra en pantalla el cuadrante y la interpolación realizada.