

Taller 3: Procesamiento de video en algoritmos paralelos — Transformación a escala de grises

Portada

Taller 3 — Procesamiento de video: conversión a escala de grises (secuencial vs paralelo)

Asignatura: HPC

1. Objetivo del laboratorio

El objetivo de este laboratorio es implementar y comparar la eficiencia de un algoritmo secuencial y uno paralelo multihilos al procesar un video, transforman el video originales a uno a escala de grises, El resultado de este laboratorio será un informe que demuestre el speedup (aceleración) logrado con el paralelismo.

2. Marco teórico

El procesamiento de video es un área fundamental dentro de la computación visual y el cómputo de alto rendimiento (HPC). Un video está compuesto por una secuencia de fotogramas (frames), generalmente mostrados entre 24 y 60 veces por segundo, lo que permite la ilusión de movimiento. Cada frame es esencialmente una imagen matricial que contiene información de color, brillo y estructura, y puede ser procesada individualmente. Esta independencia entre fotogramas es lo que permite aplicar técnicas de paralelismo de manera eficiente.

2.1. Representación digital del video

Un video digital está conformado por:

- **Frames:** imágenes estáticas ordenadas en el tiempo.
- **FPS (Frames por segundo):** ritmo de reproducción.
- **Resolución:** número de píxeles en cada frame.
- **Codificación y compresión:** formatos como MP4, AVI o MKV utilizan códecs como H.264 para almacenar de forma comprimida.

Para manipular un video, normalmente este debe ser decodificado en memoria para obtener cada uno de sus fotogramas. Bibliotecas como OpenCV facilitan este proceso mediante `cv2.VideoCapture()`.

2.2. Conversión de color: RGB a escala de grises

La conversión a escala de grises es una operación común en visión por computador. El objetivo es reducir la complejidad de la imagen eliminando la información de color, conservando únicamente la intensidad luminosa. En OpenCV, la conversión se realiza mediante la función cvtColor, que transforma cada píxel RGB en un valor de intensidad en escala de grises.

2.3. Paralelismo en procesamiento de video

Cada frame de un video puede procesarse de forma independiente, por lo que este tipo de problema es *altamente paralelizable*. Esto permite dividir el trabajo entre múltiples hilos o procesos.

2.4. Speedup y eficiencia

El *speedup* es una métrica clave en HPC usada para comparar el desempeño entre una versión secuencial y una paralela:

Speedup = Tiempo Secuencial / Tiempo Paralelo.

Un speedup mayor a 1 indica mejora en desempeño; valores cercanos al número de núcleos disponibles indican una paralelización eficiente.

2.5. Consideraciones sobre Python y el GIL

Python posee el *Global Interpreter Lock* (GIL), que limita la ejecución concurrente de código Python puro. Sin embargo, muchas funciones de OpenCV están implementadas en C y liberan el GIL, lo que permite que el uso de múltiples hilos sí mejore el rendimiento.

Cuando el trabajo es estrictamente CPU-bound en Python puro, los procesos suelen ofrecer mejor rendimiento que los hilos.

3. Metodología

La metodología implementada en el laboratorio consistió en:

- Extraer los fotogramas del video original.
- Procesar cada imagen para convertirla a escala de grises en dos versiones:
- Secuencial.
- Paralela usando múltiples hilos.
- Guardar los resultados en carpetas separadas.
- Reconstruir el video final a partir de los frames procesados.
- Medir y comparar tiempos de ejecución.

4. Análisis del código

4.1. Estructura

Se utilizó una única función para realizar la conversión a escala de grises, reutilizada tanto en la versión secuencial como en la paralela.

4.2. Rendimiento

Se identificó que el sistema de lectura y escritura en disco puede convertirse en cuello de botella. Según el hardware, la E/S puede limitar los beneficios del paralelismo.

4.3. Medición

Se recomienda utilizar mediciones repetidas y métricas estadísticas (media y desviación estándar) para obtener conclusiones más precisas.

5. Comparaciones

Se deben comparar:

- Tiempo secuencial vs paralelo.
- Speedup obtenido.
- Número de hilos utilizados.
- Influencia del tamaño del video (FPS, resolución y cantidad de frames).

6. Conclusiones

Las conclusiones deben basarse en los tiempos obtenidos. En general, se espera observar el GIL. Las limitaciones detectadas deben ser incluidas, como saturación del disco o cuellos de botella de memoria.