



Práctico Final Integrador

Objetivos del Práctico:

A través de esta actividad se pretende que el alumno comprenda la problemática asociada al diseño e implementación de buenos sistemas paralelos, debiendo integrar los conceptos vistos en la materia, justificando las decisiones de diseño e implementación y los resultados obtenidos.

Temas a tratar:

Computación Paralela. Motivación. Características. Ventajas y Desventajas. Características de las aplicaciones. Estrategias de división de problemas. Comunicaciones.

Metodología:

Resolución del práctico de máquina y presentación de informe.

Ejercicio:

Cross-Fading de imágenes

El análisis y procesamiento de imágenes se realiza a través de computadoras, debido a la complejidad y/o el número de cálculos necesarios para realizarlo. Si bien la formulación matemática necesaria para su realización data de varios siglos atrás, la posibilidad real de utilizarlo ha sido posible recién en las últimas décadas, gracias al avance en las tecnologías del hardware.

Un Bitmap es un modo elemental para representar imágenes digitales. Consiste, básicamente, en formar arreglos de elementos (vectores, matrices, tensores) ordenados de alguna manera. En general, para imágenes 2D, se realiza un ordenamiento por filas de elementos de matriz: pixels, asignando a cada uno un valor para determinar "el color" en dicha posición de la imagen.

En el caso de imágenes a color el valor de cada elemento de matriz es un vector de tres coordenadas, cada una de las cuales especifica "el grado de influencia" de los colores rojo (Red "R"), verde (Green "G") y azul (Blue "B"), de modo que se denomina representación RGB). Existen otros modos de representación a color, como por ejemplo CMYK (cian, magenta, amarillo y negro). En este trabajo utilizaremos RGB.

Cada color tiene su tonalidad. Típicamente se emplean escalas, la más común es de 8-bits: valores en $[0,255]$. Para una rerepresentación RGB, es posible producir 16.8 millones de diferentes colores, más que los distinguibles por el ojo humano.

Todos los colores en el rango visible pueden representarse como combinaciones RGB, variando desde el negro (0,0,0) al blanco (255,255,255). Por lo tanto, una imagen RGB es representada por un arreglo bidimensional de *pixels* de tres elementos, cada uno con su tonalidad.



La técnica de cross-fading es una técnica sencilla, aplicada en la edición de audio y video. Un ejemplo de su utilización es el video *Black Or White* de Michael Jackson, Caminata Lunar, donde se aplicó por primera vez en 1991.

Cross-fading produce una transición suave entre dos objetos, imágenes en nuestro caso. La idea es evitar la transición abrupta, permitiendo que una se desvanezca gradualmente mientras va apareciendo la siguiente. Esto crea una transición continua y sin interrupciones.

Dada una representación de imágenes RGB, para aplicar cross-fading entre dos imágenes se realizan las siguientes operaciones en cada pixel:

```
result[x][y].r = int(image1[x][y].r * P + image2[x][y].r * (1-P);  
result[x][y].g = int(image1[x][y].g * P + image2[x][y].g * (1-P);  
result[x][y].b = int(image1[x][y].b * P + image2[x][y].b * (1-P);
```

Se busca producir un video donde se muestre la transición entre dos imágenes. Para un segundo de video es necesario contar con al menos 24 imágenes.

Se pide:

1. Realizar un programa secuencial en C el cual sea capaz de generar las imágenes para un video de 4 segundos donde se muestre el cross-fading de dos imágenes.
2. Confeccionar una versión paralela del algoritmo antes planteado en memoria distribuida utilizando pasaje de mensajes (MPI).
3. Desarrollar una versión paralela del algoritmo antes planteado en memoria compartida utilizando Threads (OpenMP). Usar distintas configuraciones de threads.
4. Realizar un análisis del desempeño de cada una de las aplicaciones. Para ello considere el Speedup y la Eficiencia como medidas de performance. Realizar las pruebas con imágenes de 800x800, 2000x2000 y 5000x5000, con:
5. Realice un informe acerca de la implementación y el análisis comparativo del desempeño de los algoritmos desarrollados.

Nota:

- Puede trabajar con cualquier tipo de imágenes: JPEG, TIFF, RAW, PGM, PNG, BMP, debiéndolas convertir a una matriz de pixeles, Para ello puede utilizar herramientas provistas por bibliotecas como STL, SFML, entre otras.
- Para la producción del video final, utilice algún editor de video como CapCut, Inshot, Canva, etc.