

Optimización de operadores PDI con instrucciones SIMD

1. Descripción

En este proyecto se optimizarán operadores básicos de procesamiento digital de imágenes por medio de instrucciones SIMD. Se deberán utilizar las instrucciones intrínsecas en C/C++ de procesadores Intel para la optimización, utilizando los sets de instrucciones hasta AVX2. Si lo desea puede limitar los sets a SSE2, SSE3 o SSE4.

Debe elegir uno de los siguientes dos operadores:

1. Filtro de máximos/mínimos
2. Operador de componentes conectados

1.1. Filtro de máximos/mínimos

Los filtros de mínimos o de máximos son filtros de rango. Para el presente proyecto se puede suponer que dichos filtros asignan a un pixel central el valor mínimo (o máximo) en una ventana de un tamaño impar dado a su alrededor. Para la optimización puede fijar este valor a 5×5 . La estructura de ambos filtros es idéntica, excepto por el operador de rango empleado, por lo que basta con implementar uno de ellos.

Puede utilizarse el trabajo de Dokládál and Dokladalov [1] como punto de partida.

Debido a que los filtros máximos y mínimos están directamente asociados con los operadores morfológicos de dilatación y erosión, respectivamente, pueden buscarse eventualmente optimizaciones de dichas operaciones morfológicas para elementos estructurales cuadrados.

La LTI Lib-2 tiene los filtros `lti::maximumFilter` o `lti::minimumFilter` en una implementación no optimizada. La OpenCV tiene implementaciones parcialmente optimizadas de la erosión y dilatación. Utilícelos en sus comparaciones de desempeño.

1.2. Detector de componentes conectados

El operador estructural de detección de componentes conectados es un operador básico del análisis de imágenes. Su función es asignar un número consecutivo único para cada región conectada en una imagen de etiquetas. Una región conectada es un conjunto de píxeles adyacentes que comparten un mismo valor de pixel. Esta imagen de etiquetas tiene usualmente píxeles con valores enteros de 8 u 32 bits.

Puede utilizarse el trabajo de Bailey and Johnston [2] como punto de partida.

Existen dos formas del problema: uno asume los valores de entrada binarios (0 el fondo, no-cero es región a re-etiquetar); el otro asume que la entrada es etiquetada (0 el fondo, todos los otros valores representan una identificación de región distinta).

La LTIlib-2 tiene el funtor `lti::fastRelabeling` que puede realizar la detección de componentes conectadas tanto en imágenes binarias como etiquetadas. La OpenCV 2.4 tiene los métodos `cv::findContours` que encuentran componentes conectados en entradas binarias. No hay equivalente para entradas etiquetadas.

2. Objetivos

Al finalizar el proyecto, el estudiante podrá:

1. utilizar principios encontrados tras una revisión de bibliografía científica actualizada para proponer una solución avanzada
2. profundizar en el funcionamiento de un algoritmo de modo que permita la optimización de recursos apropiada para un sistema embebido (sin limitaciones energéticas, pero computacionales)
3. utilizar técnicas avanzadas de programación optimizada utilizando instrucciones intrínsecas SIMD
4. comparar la eficiencia de algoritmos de forma objetiva
5. resumir resultados en un formato científico

3. Metodología

Como pasos para este proyecto:

1. Realice una revisión bibliográfica sobre el tema elegido.
2. Implemente un programa que mida el tiempo de ejecución de forma objetiva y precisa. Mida una vez el tiempo de ejecutar n repeticiones del algoritmo a prueba, de modo que usted pueda entonces luego dividir por n para obtener la duración promedio. Reduzca los costos adicionales (*overhead*) del código, para lograr que dicha medición sea precisa. Idee una forma de medir la varianza en la duración de una ejecución.

Usted puede medir el tiempo de ejecución en función del tamaño de imagen de entrada.

3. Seleccione un algoritmo concreto para desarrollar. Debe buscar tanto estrategias eficientes del algoritmo en general, como estrategias de optimización en una arquitectura SIMD.
4. Implemente el algoritmo seleccionado, utilizando optimización vectorial (SIMD) con SSE x , o AVX x .

Puede utilizar cualquiera de las bibliotecas utilizadas en el curso como plataforma de desarrollo, pero debe tener cuidado de que su memoria sea reservada con los requisitos de alineación de SSE o AVX que emplee.

5. Prepare un artículo en formato científico donde resuma la revisión bibliográfica y los resultados obtenidos.

4. Entregables

1. Los archivos fuente.
2. Un artículo en formato científico con una descripción del método realizado. Para la sección de resultados debe diseñar un método de evaluación cuantitativa comparándolo con el detector sencillo indicado en las líneas anteriores.

Referencias

- [1] Petr Dokládál and Eva Dokladalova. Computationally efficient, one-pass algorithm for morphological filters. *Journal of Visual Communication and Image Representation*, 22(5):411–420, 2011.
- [2] D.G. Bailey and C.T. Johnston. Single pass connected components analysis. In *Proc. Image and Vision Computing New Zealand*, pages 282–287, December 2007.