Índice

1. Introducción **2**

2. Desarrollo **3**
2.1. Smalltiles . . . 3
2.1.1. Descripcion . . . 3
2.1.2. Codigo C . . . 3
2.1.3. Codigo ASM . . . 3
2.2. Rotar . . . 3
2.2.1. Descripcion . . . 3
2.2.2. Codigo C . . . 3
2.2.3. Codgio ASM . . . 3

3. Resultados **4**

4. Conclusiones **5**

1. Introducción

tu vieja e re gata wachen

2. Desarrollo

2.1. Smalltiles

2.1.1. Descripcion

El filtro Smalltiles consiste en repetir imagen de n*n en cuatro imagenes de n/2*n/2 Para eso los codigos consisten en agarrar una de cada dos filas, recorrerla y copiar uno de cada dos pixeles de la misma sobre cada sub imagen. Esto porque justamente al tratar de achicar la imagen a un cuarto, vamos a querer la mitad de altura de la imagen, asi que solo tenemos encuenta la mitad de las filas, y la otra mitad se descarta, al igual que cuando recorremos las columnas, solo la mitad se tiene en cuenta y la otra se descarta.

2.1.2. Codigo C

El codigo en C consiste en implementar dos for que nos permitan recorrer toda la imagen de a un pixel por cada dos pixeles que se encuentran en cada fila, y una fila de acada dos filas que hay en la imagen. Y los pixeles seleccionados se imprimen por cada iteracion del for en cada sub imagen.

2.1.3. Codigo ASM

El codigo en ASM se intenta implementar una idea parecida al implmentado en C, pero en este se intenta explotar lo maximo posible el conjunto de instrucciones SSE de procesamiento simultaneo. Aca para recorrer, se utiliza la misma idea de un ciclo principial que recorra las filas, y uno interior a este que recorra las columnas de cada fila. En el codigo utilizamos los registros xmm de 128 bits, que al hacer una lectura de memoria leerian 4 pixeles, y como mencionamos anteriormente nuestra idea se basaba en solo tener en cuenta uno de cada dos pixeles, por lo que dos de esos 4 pixeles no son de nuestra utilidad, asi que para poder aprovechar el tamaño de los estos registros hacemos dos lecturas por iteracion con dos registros diferentes, unimos los pixeles utiles en un solo registro, y los imprimimos debidamente en cada sub imagen, esto siempre por cada iteracion del sub ciclo.

2.2. Rotar

2.2.1. Descripcion

En este filtro se pretende reproducir la imagen src en la imagen destino con los mismos tamnos, pero con la unica diferencia de rotar los canales de la siguiente forma:

- CanalRojoDest ← CanalAzulSrc
- CanalVerdeDest ← CanalRojoSrc
- CanalAzulDest ← CanalVerdeSrc

2.2.2. Codigo C

El codigo C simplemente consiste en dos ciclos for acoplados, uno para columnas y otro para filas, cosa de recorrer uno por uno los pixeles de toda la imagen y cambiando directamente los canales de la manera ya planteada e imprimirlos por cada iteracion en la iamgen src

2.2.3. Codgio ASM

En el codigo asm la idea es similar a la implementada en C, de dos ciclos combinados para recorrer toda la image, solo que aca utilizamos los registros xmm para levantar 4 pixeles por cada iteracion, una instruccion del conjunto de instruciciones SSE, para hacer la rotacion conjunta de los 4 pixeles, e imprimimos directamente por cada iteracion los 4 pixeles en la imagen Dest.

3. Resultados

4. Conclusiones