

TP 3 : Utilisation des instructions SIMD en graphique

0 Introduction

Le but de ce TP est d'étudier l'utilisation des *intrinsics* pour les instructions SIMD en graphique sur PC (jeu d'instructions IA-32)

On utilisera l'environnement disponible dans la page Web du cours « Architectures Avancées » : <http://www.lri.fr/~de/ArchiM1-1011.htm>

Faire un compte rendu des TP 2 et 3 avec fichiers contenant les programmes et envoyer par courrier électronique à de@lri.fr. Ceci constitue le **devoir 1** à rendre pour le **25/10/2010**.

1 Filtre Laplacien

On applique sur une image X le noyau de convolution Laplacien 3 x 3 suivant pour obtenir une image Y :

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \text{ qui correspond au calcul suivant}$$

$$Y[i][j] = (4*X[i][j] - X[i][j-1] - X[i][j+1] - X[i-1][j] - X[i+1][j]) + 128;$$

1° Exécuter la fonction lap1 (version scalaire) sur lena512 et lena1024 et mesurer le temps d'exécution (cpp)

2° Exécuter la fonction lap1_simd (version simd) sur lena512 et lena1024 et mesurer le temps d'exécution. Quelle est l'accélération résultant de l'utilisation des instructions SIMD ?

NB : les fonctions lapl et lapl_simd sont fournies (site web du cours)

2 Filtre gaussien

Pour le filtre gaussien ci-dessous,

$$(1/16) * \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

1° Ecrire la fonction scalaire Gauss et mesurer les temps d'exécution (cpp) sur lena512 et lena1024

2° Ecrire la fonction simd Gauss_simd et mesurer les temps d'exécution (cpp) sur lena512 lena1024

3 Gradient de Sobel

$$G_x = \frac{1}{8} \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad G_y = \frac{1}{8} \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$$

$$G = |G_x| + |G_y|$$

Les noyaux G_x et G_y sont les filtres de Sobel horizontaux et verticaux. La norme L1 du gradient est G .

1° Ecrire une fonction `sobel0` qui prenne en entrée une image X et qui renvoie en sortie trois images G_x , G_y et G qui est la norme L1 de l'image X . Ce calcul se fera via l'appel en interne de trois fonctions `sobelx`, `sobely` et `normeL1`. Les images devront être allouées au préalable.

Mesurer le temps d'exécution (cpp) pour `lena512` et `lena1024` de la fonction `sobel0`.

2° Ecrire une fonction `sobel1` qui calcule directement la norme L1 en prenant comme paramètre l'image X et retourne la norme dans l'image G , sans calcul intermédiaire.

Mesurer le temps d'exécution (cpp) pour `lena512` et `lena1024` de la fonction `sobel1`.

3° Ecrire une version SIMD de la fonction `sobel1`. Mesurer le temps d'exécution (cpp) pour `lena512` et `lena1024` de cette version `sobel1_simd`.