ÉCOLE POLYTECH 'NICE SOPHIA
SI4
2020-2021

Julien Deantoni

Programmation orientée objet : CPP

Gestion simple de la concurrence !

Objectif

Basé sur le TD de la semaine précédente, il faut que vous rajoutiez de la concurrence dans la manipulation d'image. Essayez de déterminer par vous même le niveau de concurrence souhaitable et implémentez les algorithmes de la semaine précédente ainsi que les nouveaux. Vous pourrez tester les performances sur la grosse image ajoutée aux images de la semaine dernière (voir sur bigImage.ppm pour les tests de performance).

Faites vos développements dans une classe Image séparée afin de pouvoir comparer les performances.

Pour la suite de traitements suivante:

```
ImageSeq grosseImage("../pict/grosse_imageP3.ppm");
grosseImage.makeItGrey();
grosseImage.returnIt();
ImageSeq grosseImage2 = grosseImage;
grosseImage2.blurIt(40);
ImageSeq grosseImage3 = grosseImage;
grosseImage3.sortIt();
```

Quel est votre temps record pour les différentes versions parallèles et séquentielles que vous avez faites ? Pourquoi ? Comment améliorer ceci ?

Personnellement, sans y passer réellement du temps j'ai des résultats du type:

> Time spent sequential mode = 5197[ms]
> Time spent concurrent fine mode = 3424[ms]
> Time spent concurrent fine and coarse mode = 3127[ms]
> Time spent concurrent coarse mode = 4449[ms]

et un record de « Time spent concurrent fine and coarse mode = 946[ms] »

Ferez-vous mieux ?

Négatif

Votre but est maintenant de faire un négatif de votre image. Pour cela vous devrez appliquer un traitement à chacune des valeurs qui encodent les couleurs. À vous de trouver le traitement à appliquer.

Noir et blanc

Maintenant que la manipulation d'image n'a plus de secret pour vous, vous allez faire une fonction membre dont le but est de mettre en noir et blanc une image. Le seuil qui distingue le blanc du noir doit pouvoir être choisi par l'utilisateur.

Retournement de situation

Après avoir joué sur les couleurs, nous allons maintenant faire une symètrie. Codez une fonction membre qui fait la symètrie de l'image de sorte que le coin en haut à gauche se retrouve en bas à doite.

Sort it

Pour une raison obscure, écrivez une fonction membre qui trie les pixels de l'image de façon croissance.

Le monde est flou !

Écrivez une fonction membre qui remplace la valeur d'un ensemble de pixels successifs de taille N (N étant un paramètre) par la valeur moyenne des pixels dans l'ensemble.

Facilities?

Écrivez une fonction membre qui permet d'appliquer une des fonctions précédente à l'image et qui sauvegarde le résultat dans un fichier dont le nom est donné en paramètre.