

Fast Exact Nearest Patch Matching for Patch-based Image Editing and Processing

Supervision : Vinh-Thong Ta
(ta@labri.fr, <http://www.labri.fr/~ta>)

Affiliation : LaBRI, équipe Image & Son,
Pictura Research Group (<http://pictura.labri.fr>)

Mots-clés : Traitements d'images ; Édition d'images ; Traitements par patch ; Algorithmes efficaces ; Accélération ; Structures de données, Images 2D, 3D, GPU ;

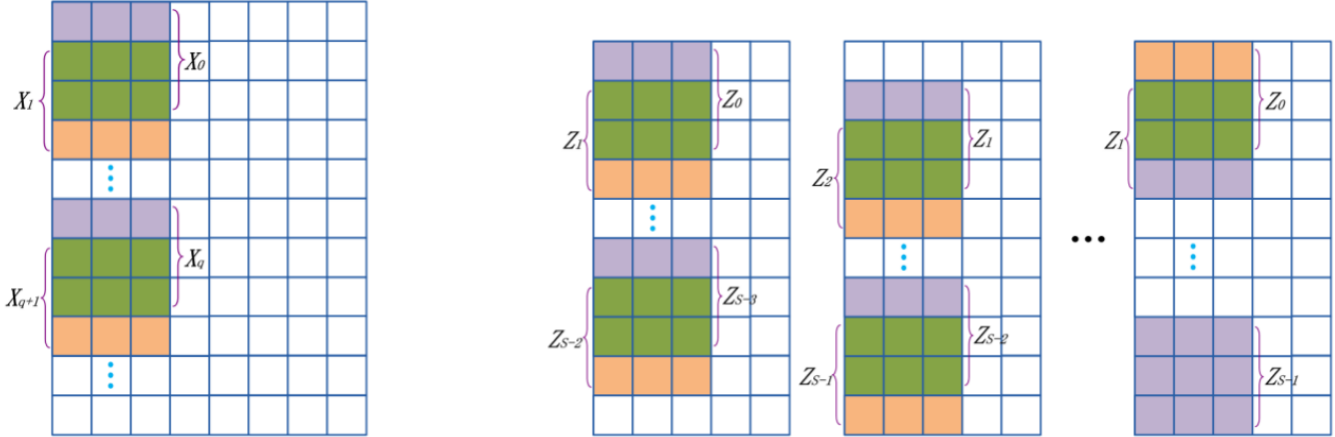


FIGURE 1 – Schéma décrivant l'algorithme proposé par [1] dans le cas d'une image 2D

Description À l'heure actuelle de nombreuses méthodes de traitements et d'édition d'image utilisent la notion de patch (petit voisinage autour du pixel à traiter) afin de caractériser de manière plus riche le pixel en lui-même (au lieu de juste utiliser la couleur ou le niveau de gris associé) dans le traitement à accomplir (filtrage, inpainting,...). Une étape du traitement est de déterminer le patch *le plus proche*, parmi tous les patches possible de l'image, du patch courant.

Afin de déterminer ce plus proche candidat, une manière classique est de calculer la différence entre les patches en utilisant une *SSD* (sum of square difference) calculée de manière *rowscan*, c.-à-d. pixel à pixel, et ceci pour l'ensemble des patches de l'image. Ce processus induit une **complexité très grande** et on peut faire remarquer que la plupart des algorithmes (dans leur version *brute-force*) utilisant des patches, **passent environ 90% de leur temps à réaliser ce calcul** interdisant de ce fait des applications en temps réel pour de grand volume de données.

On peut remarquer que, dans le calcul des distances entre patches, de nombreux calculs sont redondants car de nombreux patches se chevauchent. Partant de ce constat, [1] propose un algorithme permettant de profiter de ce chevauchement afin de rendre le calcul du **plus proche patch exact** moins coûteux. La figure 1 illustre une partie de l'algorithme proposé.

Objectifs Les objectifs de ce projet sont :

- d'étudier l'article de Xiao *et al.* [1] et de comprendre l'algorithme proposé dans sa version initiale ;
- de comprendre l'extension de l'algorithme à des images 3D ;
- de comprendre l'optimisation réalisée via programmation parallèle (GPU) ;
- de proposer une implémentation afin de tester et confirmer les résultats de l'article, on commencera par la version 2D puis on envisagera les différentes extensions. Le choix du langage de programmation sera à déterminer entre l'encadrant et le groupe d'étudiants.

Références

- [1] Chunxia Xiao, Meng Liu, Nie Yongwei, and Zhao Dong. Fast exact nearest patch matching for patch-based image editing and processing. *Visualization and Computer Graphics, IEEE Transactions on*, 17(8) :1122–1134, 2011.