

PatchMatchGraph : Building a Graph of Dense Patch Correspondences for Label Transfer

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 par patch ; Algorithmes de correspondance ; Graphes ; Segmentation multi-labels ; Classification

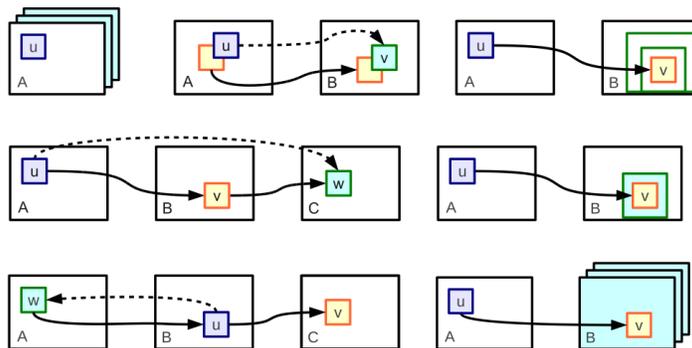


FIGURE 1 – Schéma décrivant une partie de l'algorithme proposé par [2]

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.

Calculer pour chaque pixel, son pixel le plus proche correspond à calculer une carte de correspondance appelée *NNF* (nearest neighbor field). Un des algorithmes les plus efficace et rapide de la littérature est celui proposé par [1], l'algorithme itératif du PATCHMATCH, qui permet de calculer un *NNF* approché (et non exact) en se basant sur une notion de propagation de pixel (patch) voisin et une recherche aléatoire. Étant donné une image *A*, le PATCHMATCH permet d'obtenir la carte de correspondance pour une seconde image *B*, c.à-d., $j = NNF(i)$ où j correspond au pixel de *B* le plus proche (au sens des patches) du pixel i de *A*

Partant de ce principe, Gould et Zang [2] propose une modification de l'algorithme afin de pouvoir calculer un graphe des k plus proches voisins (*au lieu du plus proche*, $k = 1$) d'une part et, d'autre part de ne pas considérer une seule image *B* mais un ensemble d'images cibles. La Figure 1 montre le principe de l'algorithme. Les auteurs proposent d'appliquer l'algorithme pour faire de la segmentation multi-labels (objets) ou de la classification de données. En effet, l'avantage de modéliser le problème sous forme de graphe permet d'appliquer le même algorithme à différent type de données.

Objectifs Les objectifs de ce projet sont

- d'étudier l'article de Gould et Zang [2] et de comprendre l'algorithme proposé ;
- les auteurs proposent une implémentation de la méthode (<http://drwn.anu.edu.au/drwnProjPatchMatch.html>)
Le but est d'étudier et de comprendre le code source fourni, de tester et de confirmer les résultats obtenus par les auteurs. Enfin on proposera une implémentation simplifié de la méthode. Le choix du langage de programmation sera à déterminer entre l'encadrant et le groupe d'étudiants.

Références

- [1] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. PatchMatch : A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 28(3), August 2009.
- [2] Stephen Gould and Yuhang Zhang. Patchmatchgraph : Building a graph of dense patch correspondences for label transfer. In *ECCV*, volume 7576, pages 439–452, 2012.