

Graphe des composantes connexes : algorithmique et applications

Équipe d'accueil

Équipe MIV (Modèles, Images et Vision) du Laboratoire des Sciences de l'Ingénieur, de l'Informatique et de l'Imagerie (ICube), Université de Strasbourg

Contexte

En traitement d'images, les opérateurs connexes forment une classe d'opérateurs qui simplifient une image tout en préservant ses contours. Par définition, les opérateurs connexes ne peuvent pas déplacer les contours existants ni en créer de nouveaux, ce qui leur confère un grand intérêt pour filtrer une image.

Dans le cas d'une image dont les valeurs sont ordonnées selon une relation d'ordre totale (ce qui est le cas des images à niveaux de gris par exemple), l'arbre des composantes connexes [1, 2, 3] permet de stocker l'ensemble des composantes connexes des coupes binaires de l'image. La construction de cet arbre est rapide grâce à des algorithmes à la complexité linéaire. Élaguer cet arbre avec un critère croissant revient à appliquer un opérateur connexe anti-extensif. Cette structure a été abondamment étudiée et de nombreuses applications l'utilisent dans la littérature (à des fins de segmentation, détection d'objets, filtrage d'images, etc.).

L'extension de cette structure à des images dont les valeurs ne sont pas ordonnées par une relation d'ordre totale (par exemple les images multivaluées, couleur ou multispectrales) a donné lieu récemment à la notion de graphe des composantes connexes [4, 5]. De nombreuses questions restent à étudier sur cette structure, tant théoriques qu'algorithmiques.

Objectifs

Le graphe des composantes connexes se décline en trois versions : \mathcal{G} , $\hat{\mathcal{G}}$ et $\check{\mathcal{G}}$ de la plus riche à la moins riche en nombre de nœuds (voir figure 1). Les objectifs de ce stage sont les suivants :

1. Rechercher de nouveaux algorithmes ou optimiser les algorithmes actuels de calcul du graphe des composantes-connexes ;
2. Étudier les problématiques et politiques de reconstruction de l'image filtrée à partir du graphe simplifié ;
3. Étudier, concevoir et implanter des algorithmes de construction de l'image filtrée après traitement du graphe ;
4. De manière complémentaire, étudier des stratégies pour réduire le nombre de nœuds du graphe (quantification de l'espace des valeurs par des techniques de réduction de dimensionalité, par exemple) ;
5. Étudier et proposer des techniques de parallélisation (en s'inspirant de [6] ou [5]).
6. Construire de nouveaux opérateurs connexes à base de cette structure et trouver des exemples d'applications (voir figure 2).

Développement logiciel

Le prototypage des algorithmes pourra s'effectuer en Python. Le développement des méthodes finalisées devra s'effectuer en C++ et s'intégrer dans les dépôts Github :

- <https://github.com/bnaegel/libtim>
- <https://github.com/bnaegel/component-graph>

Compétences souhaitables

- Traitement d'images
- Algorithmique
- Programmation (C++, Python)

Encadrement (à contacter pour plus d'informations)

- Benoît NAEGEL - Équipe MIV - ICube Bureau C230 - b.naegel@unistra.fr
- Christian RONSE - Équipe MIV - ICube Bureau C228a - cronse@unistra.fr

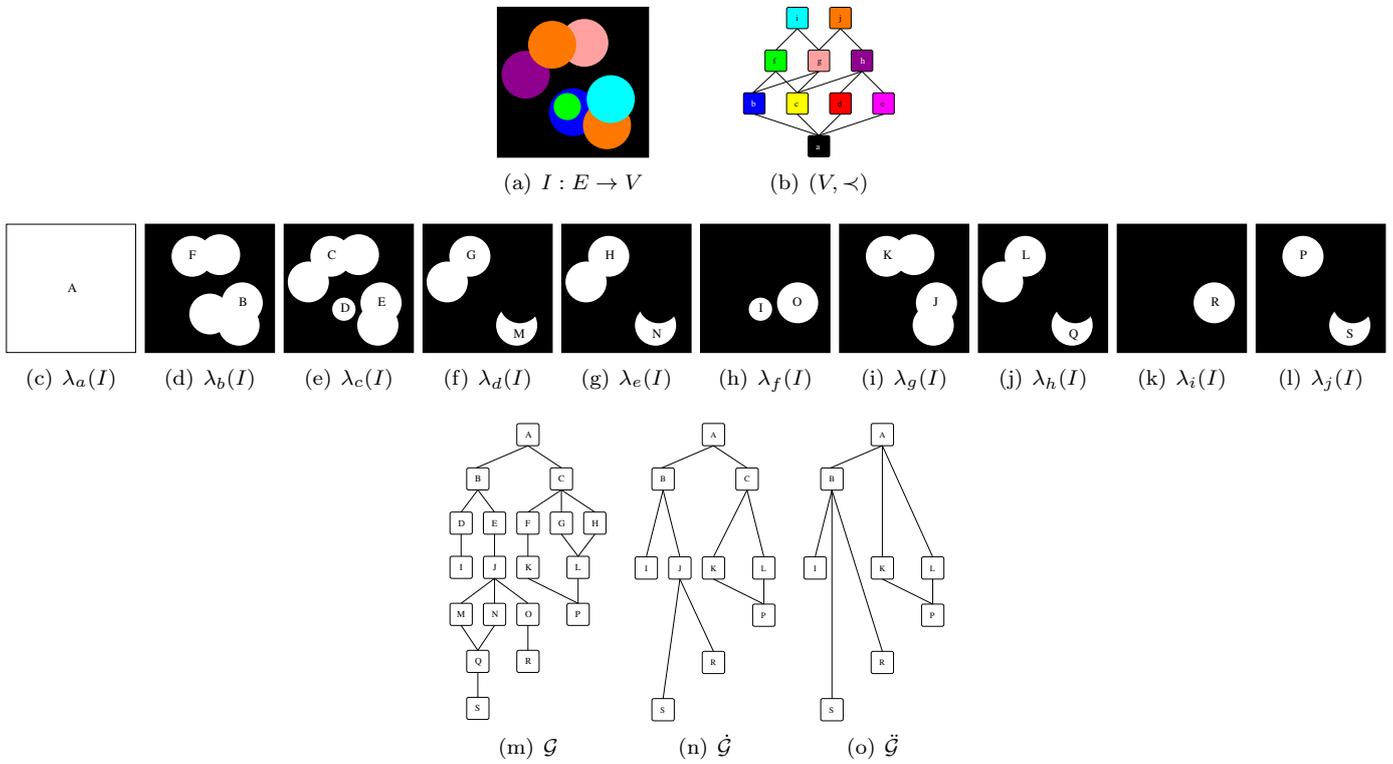


FIGURE 1 – (a) Une image $I : E \rightarrow V$, avec $V = \{a, \dots, j\}$. (b) Le diagramme de Hasse de l'ensemble partiellement ordonné (V, \leq) . (c-l) Coupes binaires $\lambda_v(I)$ de l'image pour l'ensemble des ses valeurs $v \in V$. (m-o) Les graphes de composantes connexes de I . Les lettres (A-S) correspondent aux composantes connexes de (c-l).

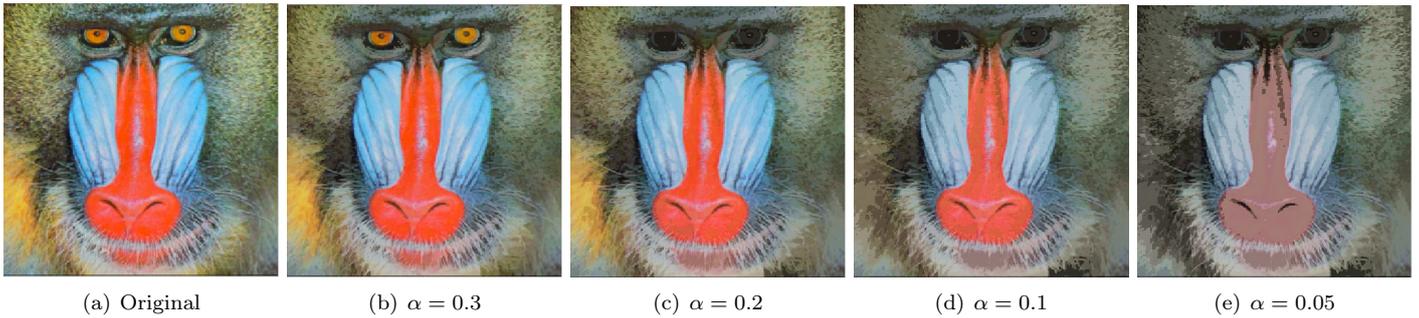


FIGURE 2 – Exemple d'opérateur connexe à base de graphe des composantes connexes.

Références

- [1] P. Salembier, A. Oliveras, L. Garrido Anti-extensive Connected Operators for Image and Sequence Processing IEEE Transactions on Image Processing, Vol.7 Num. 4, p. 555-670, 1998
- [2] L. Najman, M. Couprie Building the component tree in quasi-linear time. IEEE Transactions on Image Processing, Vol.15 Num. 11, p. 3531-3539, 2006
- [3] E. Carlinet, T. Géraud A fair comparison of many max-tree computation algorithms. Computing Research Repository <http://arxiv.org/abs/1212.1819>
- [4] N. Passat, B. Naegel Component-trees and multivalued images : Structural properties. Journal of Mathematical Imaging and Vision, Vol. 49 Num. 1, p. 37-50, 2014
- [5] B. Naegel, N. Passat Colour image filtering with component-graphs. International Conference on Pattern Recognition (ICPR), p. 1621-1626, IEEE, Stockholm, Sweden, 2014.
- [6] G. Palma, I. Bloch, S. Muller Fast fuzzy connected filter implementation using max-tree updates. Fuzzy Sets and Systems, Vol. 161, Num. 1, p. 118-146, 2010.