

Segmentation d'images

BELOT Tom, GAMIETTE Didier

Telecom Physique Strasbourg

18 janvier 2025

Table des matières

1	Résumé	2
2	Description du projet	3
2.1	Objectif principal	3
2.2	Contexte et enjeux	3
2.3	Contraintes et défis	3
2.4	Résultats attendus et perspectives	4
3	Méthodologie et conception	5
3.1	Représentation de l'image en un graphe	5
3.2	Création de l'arbre couvrant de poids minimal	5
3.3	Seuillage et extraction des zones de l'image	6
3.4	Création de la nouvelle image segmentée	6
3.5	Résumé du processus	7
4	Tests et résultats	8
5	Conclusion	9
	Bibliographie	11

1. Résumé

Le projet a pour objectif de créer un programme informatique permettant la segmentation d'images, une technique essentielle en traitement d'images, en biométrie ou en analyse scientifique pour isoler des zones d'intérêt dans des fichiers visuels. En utilisant des algorithmes basés sur la théorie des graphes, le programme développé permettra d'extraire et d'analyser des régions pertinentes au sein de ces images. Le projet vise à fournir un outil avec des fonctionnalités telles que la segmentation automatique, l'analyse des propriétés du graphe associé à l'image et l'export des résultats obtenus.

2. Description du projet

2.1 Objectif principal

Le projet a pour objectif de développer un programme de segmentation d'images basé sur la théorie des graphes. L'idée est d'extraire des zones d'intérêt homogènes à partir d'images, en tenant compte des différences de couleur entre les pixels, et de produire une version segmentée de ces images où chaque région d'intérêt est représentée par une couleur distincte.

2.2 Contexte et enjeux

La segmentation d'images est une étape fondamentale dans des domaines variés tels que la vision par ordinateur, l'imagerie médicale et la reconnaissance d'objets. Ce projet est réalisé dans un cadre pédagogique afin d'appliquer des concepts de programmation et de gestion de projet tout en travaillant avec des données complexes comme les images.

2.3 Contraintes et défis

Le projet doit répondre aux spécifications suivantes :

- Supporter les formats JPEG, PNG et TIFF.
- Implémenter des tests automatisés pour valider les fonctions du programme.
- Produire un code propre, documenté, sans erreurs.
- Optimiser les performances pour traiter efficacement un corpus d'images varié.

2.4 Résultats attendus et perspectives

Le programme final devra être capable de produire des images segmentées où les zones d'intérêt sont colorées différemment. Si besoin, des extensions comme l'ajout de nouvelles techniques de segmentation ou l'optimisation pour des images ayant une segmentation difficile pourraient être envisagées.

3. Méthodologie et conception

3.1 Représentation de l'image en un graphe

L'idée principale de la segmentation d'une image repose sur sa transformation en un graphe avec lequel chaque pixel de l'image est représenté par un sommet.

Chaque sommet possède des valeurs RGB (Rouge, Vert, Bleu) du pixel qu'il représente dans l'image. Ainsi, pour chaque pixel de l'image, nous pouvons lui associer un sommet du graphe.

Les arêtes du graphe représentent les connexions entre pixels adjacents. Le poids associé à une arête entre deux pixels correspond à la distance euclidienne dans l'espace de couleurs CIELAB (espace obtenu au préalable grâce valeurs RGB des pixels de l'image), un espace de couleurs permettant de représenter au mieux les différents éléments observables par l'œil humain.

Le poids p_{ij} entre deux sommets adjacents i et j est défini comme suit :

$$p_{ij} = \sqrt{(L_i - L_j)^2 + (a_i - a_j)^2 + (b_i - b_j)^2}$$

où :

- (L_i, a_i, b_i) représente les composantes de la couleur du sommet i dans l'espace LAB ;
- (L_j, a_j, b_j) représente les composantes de la couleur du sommet j dans l'espace LAB.

3.2 Création de l'arbre couvrant de poids minimal

Une fois l'image représentée sous forme de graphe, l'objectif est de réduire le graphe afin de segmenter l'image en zones uniformes. Pour ce faire, nous utilisons l'algorithme de Kruskal pour construire un arbre couvrant de poids minimal à partir du graphe initial.

L'algorithme de Kruskal est bien adapté à ce problème, car il permet de construire un arbre couvrant de poids minimal en ajoutant successivement les arêtes de poids minimal, tout en assurant qu'il n'y ait pas de cycle. Les étapes principales sont les suivantes :

- Trier toutes les arêtes du graphe par poids croissant.
- Ajouter les arêtes les plus légères au graphe, tout en vérifiant qu'elles ne forment pas de cycle.
- Répéter ce processus jusqu'à ce que l'arbre couvrant soit complet, c'est-à-dire jusqu'à ce qu'il relie tous les sommets sans créer un cycle.

Cela permet de créer un arbre couvrant qui relie tous les pixels de l'image, mais avec une priorité donnée aux pixels ayant des couleurs proches, créant ainsi des groupes de pixels similaires.

3.3 Seuillage et extraction des zones de l'image

Une fois l'arbre couvrant de poids minimal construit, il est nécessaire de le seuiller pour obtenir les zones distinctes de l'image. Le seuillage consiste à supprimer les arêtes dont le poids (la distance des couleurs des pixels dans l'espace CIELAB) dépasse un certain seuil. Ce seuil est un paramètre important, car il contrôle la segmentation : un seuil bas donnera des zones plus petites, tandis qu'un seuil élevé donnera des zones plus grandes. Une fois le seuil appliqué, le graphe devient un ensemble de composantes connexes, où chaque composant représente une zone de l'image.

Dans notre cas, le seuil est défini tel que 20% des arêtes les plus lourdes soient supérieures à celui-ci.

Les composantes connexes de ce graphe correspondent aux différentes zones de l'image, qui peuvent être interprétées comme des régions uniformes en termes de couleur. Ces zones sont ensuite extraites comme les régions d'intérêt de l'image segmentée.

3.4 Création de la nouvelle image segmentée

À partir des composantes connexes obtenues, nous générons une nouvelle image où chaque zone segmentée est colorée de manière distincte. A chaque composante connexe est attribuée une couleur unique, permettant ainsi de visualiser les différentes régions

segmentées de l'image. Ce processus consiste à assigner une couleur différente à chaque composante connexe et à générer une nouvelle image avec ces couleurs.

Ainsi, la nouvelle image segmentée met en évidence les différentes zones d'intérêt en les coloriant de manière uniforme, en fonction de leur appartenance à une composante connexe. Cela permet de visualiser de manière claire les zones homogènes de l'image, facilitant ainsi l'analyse et l'extraction d'informations.

3.5 Résumé du processus

Le processus de segmentation peut être résumé en les étapes suivantes :

- Représentation de l'image en un graphe où chaque pixel est un sommet et les arêtes sont les différences de couleur CIELAB entre les pixels voisins.
- Construction d'un arbre couvrant de poids minimal à l'aide de l'algorithme de Kruskal.
- Seuillage de l'arbre pour supprimer les arêtes trop lourdes et obtenir un graphe composé de composantes connexes.
- Extraction des zones segmentées de l'image en attribuant une couleur unique à chaque composante connexe, générant ainsi une nouvelle image segmentée.

4. Tests et résultats

De nombreux tests ont été effectués pour contrôler les différentes fonctionnalités du projet, notamment pour vérifier le comportement des fonctions seules et entre elles, vérifier les performances sur plusieurs images dont des cas "extrêmes".

Nous pouvons en conclure que la segmentation est plutôt fluide (une image de taille 3000x4000 est traitée en une trentaine de secondes) et relativement pertinente dans le cas général.

Cependant, la segmentation devient moins précise lorsque l'image dispose d'éléments ayant des couleurs très similaires, un bruit important, une résolution faible ou des objets partiellement visibles ou déformés.

5. Conclusion

L'objectif de ce projet était de développer un programme permettant de segmenter des images en zones homogènes à l'aide d'une approche basée sur la théorie des graphes.

Le programme final répond aux spécifications initiales, notamment la prise en charge des formats JPEG, PNG et TIFF, ainsi que la production d'images segmentées. Des tests ont validé le fonctionnement des différentes fonctions implémentées et confirmé l'approche adoptée dans une grande partie des images segmentées.

Cependant, certaines limites ont été rencontrées et pourraient être corrigées dans de futures itérations.

Ce projet offre de nombreuses perspectives d'amélioration, telles que l'intégration d'autres méthodes de segmentation, ou encore l'optimisation pour des images où la segmentation actuelle est "compliquée".

Enfin, ce projet a permis de mettre en pratique des concepts théoriques de manière concrète, tout en développant des compétences en programmation, en gestion de projet et en résolution de problèmes.

Bibliographie

- [1] Damien Grosgeorge and Caroline Petitjean et Su Ruan. Segmentation d'images par coupe de graphe avec a priori de forme. In *Actes de RFIA 2012 (Reconnaissance des Formes et Intelligence Artificielle)*, pages 1–9, Lyon, France, January 2012. HAL. <https://hal.science/hal-00656508v1>.
- [2] Inconnu. Graphes en traitement d'images et en reconnaissance des formes, 2017. <https://perso.telecom-paristech.fr/bloch/AIV/polyGraphes.pdf>.
- [3] Romain Lerallut. Modélisation et interprétation d'images à l'aide de graphes, August 2006. <https://hal.science/pastel-00003298v1>.
- [4] Florence Tupin. Graphes en analyse d'images, visions, reconnaissance, July 2017. <https://perso.telecom-paristech.fr/bloch/AIV/2graphes.pdf>.