

Détection et identification d'arbre à partir d'imagerie satellite/aérienne

Augustin Albert

30 avril 2021

Table des matières

1	Détection des houppiers	2
1.1	Laplacien du gaussien et approche multi échelle	2
1.2	Mise en place de l'algorithme	3
1.3	Évaluation des résultats	4
2	Identification des espèces	4
2.1	Méthodologie de construction d'une base de données fiable	4
2.2	entraînement et quelle type de modèle	4
3	Prolongements envisageables	4
	Références	5
A	Compléments	5
B	Résultats	5
C	algorithmes	5

Introduction

Position du problème

-pourquoi vouloir faire ça, utilité/contexte -1 pb extraction de données :
différentes méthodes qui requièrent plus ou moins de matériel/images de qualité
-2 pb traitement des données

État actuel de la recherche

-voir papiers

Objectifs du TIPE

-limitation à des images aériennes : pourquoi(moins couteux, accessible sur internet, différents modes d'acquisitions, enjeux/difficultés -d'une part à concevoir ... pour detecter les a -d'une autre part à l'utiliser pour contruire une base de donn   permettant identification ulterierure sur la base du machin learning -application au site du parc regional...

1 Détection des houpriers

Intro : Méthode naive (les preentations) de detection des zones plus lumineuses. La luminosité des arbres peut beaucoup varier sur une meme image (à moins d'avoir des images de haute qualité "prise en une seule fois" (ex papier). Une solution = Détection de blobs. 2 problemes : -diff  rence de luminosité et diff  rence d'échelle. Citer algo pour. de pixels plus lumineux que leur voisins correspondants aux houpriers

1.1 Laplacien du gaussien et approche multi échelle

Afin de prendre en compte des houpriers dont le diamètre peut varier consid  rablement entre les diff  rentes esp  ces mais aussi au sein d'une m  me esp  ce, une approche multi-  chelle est n  cessaire. Nous utiliserons la th  orie   chelle-temps d  velopp  e par (??indbergh). Une pyramide d'echelle est r  alis  e en lisant successivement l'image originelle avec un filtre gaussien de param  tre σ : le param  tre d'  chelle. L'image originelle est convol  e autant de fois que n  cessaire par la fonction gaussienne suivante, le param  tre σ   tant multipli   par un ratio fix     !!!! important!!!!    chaque   tape :

$$G_{\sigma} := \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

On applique alors un op  rateur laplacien normalis   aux images r  sultante afin d'obtenir la pyramide d'  chelle du laplacien du Gaussien (LoG). L'image obtenue    chaque niveau est alors la convolution de l'image originelle par la fonction Log :

$$LoG_{\sigma} := -\frac{1}{\pi\sigma^4} \left(1 - \frac{x^2 + y^2}{2\sigma^2}\right) \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

En pratique, le LoG est approxim   par la diff  rence du gaussien (DoG) obtenue en r  alisant la difference des images flout  es entre chaque niveau de la pyramide d'  chelle.

Cette pyramide d'  chelle du LoG permet d'extraire des caract  ristiques ind  pendamment de leur   chelle en exploitant la r  ponse de l'op  rateur LoG appliqu      un signal   chelon. (figure 5). Lorsque le rayon caract  ristique du blob r varie, le minimum (maximum en valeur absolue) du LoG est atteint au centre du blob. Lorsque le param  tre σ varie, la r  ponse au centre est minimale lorsque r est

- (a) 1
(b) 2
(c) 3

FIGURE 1 – test2

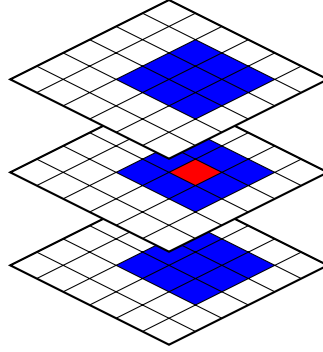


FIGURE 2 – test un deux

relié à σ par la relation $\sigma = \frac{r}{\sqrt{2}}$. La réponse du LoG non normalisé s'atténue lorsque σ augmente, l'opérateur a été multiplié par σ^2 pour que la réponse soit indépendante de l'échelle.

La détection des blobs se ramène ainsi à la recherche d'un minimum local relativement à l'espace et global relativement à l'échelle pour identifier à la fois les centres des houppiers et la taille caractéristique de leur rayon.

-figure 1 réponse à un échelon -figure 2 un blob mais différents paramètres de largeur et de paramètre!!

1.2 Mise en place de l'algorithme

L'algorithme envisagé a été implémenté à l'aide du langage Python et utilise la bibliothèque Numpy afin d'accélérer le traitement des tableaux. Il comporte trois étapes.

- L'image à traiter est convertie en nuance de gris et éventuellement inversée afin que le fond soit plus clair que les houppiers. (une intervention humaine est nécessaire).

- La pyramide d'échelle du DoG est générée. Les noyaux de Gauss et la convolution ont été implémentés en exploitant la séparabilité du filtre de Gauss pour accélérer le calcul, mais en définitive, la convolution fournie par le module scipy a été utilisée. Les résultats sont stockés dans un tableau Numpy 3D.

- Les minimums sont calculés en parcourant le tableau par profondeur croissante afin de ne conserver que les houppiers de rayon maximum lors d'éventuels chevauchements.

3 paramètres permettent d'ajuster les résultats. La sélection des paramètres dépend fortement du (très heuristique comme méthode, nécessite des essais => la méthode n'est pas complètement automatique. (dépend de la taille caractéristique des arbres, de l'échelle choisie) donner les paramètres pour l'échelle et tout.

facteur limitant i choix final des paramètres : + images utilisées (taille des arbres sur le terrain et échelle de l'image)

1.3 Évaluation des résultats

-évaluation de la complexité -propres résultats faire tableau 3 colonnes pour les trois images différentes -comparaison avec les résultats des papiers

2 Identification des espèces

Choix du machine learning comme dans de nombreux papiers (même si pas les mêmes données de base) Propre implementation développé mais pas complet et ne permet pas à ce jour de traiter efficacement des images (manque le truc de convolution et reste lent en comparaison des systèmes professionnels) Le choix s'est donc porté sur Tensorflow.

2.1 Méthodologie de construction d'une base de données fiable

Il aurait été fastidieux d'étiqueter à la main un nombre important d'images, d'autant plus qu'elles sont de basse résolution. Afin - géoportail (vérifier autorisation... côté) et extraction sur des zones où la couverture d'espèce est uniforme : res — images triées en 2

2.2 entraînement et quelle type de modèle

-citer papier, à la main car ici le cas est plus simple mais l'on pourrait généraliser. (différence avec eux)

3 Prolongements envisageables

Différents prolongements seraient envisageables : -Sensible aux ... séparer préalablement et éventuellement grossièrement les zones forestières des zones d'habitation ou industrielle. Même une route bétonnée peut éventuellement altérer les résultats. Bâtiment/construction qui fausserait les résultats. De plus obtenir qu'un cercle autour des arbres -une méthode watershed segmentation avec marqueurs que l'on a trouvés pourrait être envisageable pour délimiter parfaitement les arbres (voir papier)

Références

- [NAV19] Sowmya Natesan, Costas Armenakis, and Udaya Vepakomma. Resnet-based tree species classification using uav images. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W13 :475–481, 06 2019.

A Compléments

Lien avec filtre basse fréquence (downsampling, pourquoi?) le filtre de gauss retire les hautes fréquences donc permet d'augmenter la fréquence d'échantillonnage 'théorème de shannon sans qu'il y ait d'artefacts.

B Résultats

-résultats intermédiaire (pyramide de gauss) -les deux -douglas seul -feuillus seul -échantillon banque fourni pour feuillus -échantillon banque fourni pour douglas -échantillon aléatoire parmi des images non déjà vus

C algorithmes

a faire : réaliser les tableaux de comparaison, mettre en forme le code écrire les preuves