

---

# Interactions segmentation-classification dans un cadre multi-paradigme pour l'analyse d'images de télédétection

Andrés Troya-Galvis<sup>1</sup>, Pierre Gancarski<sup>1</sup>, Laure Berti-Équille<sup>2,3</sup>

1. ICube, Université de Strasbourg

300 bd Sébastien Brant - CS 10413 - F-67412 Illkirch Cedex, France

troyagalvis@unistra.fr, gancarski@unistra.fr

2. Qatar Computing Research Institute, Hamad Bin Khalifa University, Doha, Qatar

lberti@qf.org.qa

3. Espace-Dev UMR 228, IRD - Université de Montpellier,

500 Rue J.F. Breton, 34090 Montpellier, France

Laure.Berti@ird.fr

---

**RÉSUMÉ.** Les tâches de segmentation et de classification d'images sont étroitement liées dans le cadre de l'analyse d'images de télédétection. Les méthodes collaboratives permettent l'interaction entre les approches de segmentation et de classification afin d'améliorer simultanément leurs résultats. Dans cet article nous présentons un cadre collaboratif générique pour la segmentation et la classification d'images de télédétection, et nous faisons une étude exploratoire comparant un grand nombre de stratégies de collaboration possibles permettant de mieux comprendre les interactions entre ces deux paradigmes.

**ABSTRACT.** Segmentation and classification tasks are closely related in the remote sensing image analysis domain. Collaborative approaches allow interactions between segmentation and classification techniques in order to mutually improve both results. In this article we present a generic collaborative framework for segmentation and classification of remote sensing images, and we make an exploratory study comparing a large number of collaboration strategies in order to better understand the interactions between these paradigms.

**MOTS-CLÉS :** segmentation, classification, analyse d'images de télédétection.

**KEYWORDS:** segmentation, classification, remote sensing image analysis.

---

DOI:10.3166/RIA.31.133-152 © 2017 Lavoisier

## 1. Introduction

L'analyse d'images de télédétection consiste à associer une sémantique à chaque pixel d'une image prise par un engin aérien ou satellitaire. La masse de données images disponible ne cesse pas d'augmenter et rend l'analyse par photo-interprétation infaisable. L'interprétation automatique de telles images est une tâche difficile mais nécessaire pour des applications dans différents domaines tels que l'analyse de l'évolution urbaine (Pham *et al.*, 2011), la prévention de désastres naturels (Westen, 2013), ou le suivi d'écosystèmes (Coppin *et al.*, 2004). À basse et moyenne résolution spatiale, les approches par pixel ont prouvé leur efficacité. Néanmoins, elles atteignent leurs limites avec l'apparition des images à très haute résolution spatiale (THRS). En effet, à une telle résolution, inférieure à 5 m, les objets géographiques d'intérêt ne sont pas représentés par un unique pixel mais par un ensemble de ceux-ci. Le paradigme d'analyse d'images basée objets (*Object Based Image Analysis*, OBIA (Blaschke, 2010)) s'est donc imposé comme technique de prédilection pour traiter ce type d'images. Ces approches construisent les objets en groupant des pixels similaires avant de les classer. Une phase de segmentation est appliquée comme pré-traitement. Ensuite, chaque segment est caractérisé par un ensemble d'attributs, notamment des descripteurs géométriques ou texturaux permettant ainsi de ne plus se limiter aux seules propriétés radiométriques des objets. Enfin, une méthode d'apprentissage automatique supervisé ou non est appliquée afin de classer les segments ou pour proposer de nouvelles classes potentiellement intéressantes pour l'expert.

Les étapes de segmentation et de classification sont cruciales dans la chaîne de traitements des approches OBIA et y sont étroitement liées. En effet, il a été montré que la segmentation a un impact sur le résultat de la classification (Gao *et al.*, 2011). Il est donc essentiel de fournir à l'algorithme de classification la segmentation la plus à même d'optimiser celle-ci. Cependant, la segmentation d'images est un problème mal posé, la notion de qualité étant subjective et dépendante du contexte. De fait, il n'existe pas de méthode universelle permettant de segmenter de manière optimale la totalité des objets géographiques d'intérêt. De plus, les approches classiques effectuent la segmentation de l'image de manière indépendante à la classification ; elle n'est jamais remise en cause. Par conséquent, si la segmentation initiale n'est pas adaptée aux critères employés par la méthode de classification, les résultats de cette dernière ont une forte chance d'être sous-optimaux.

Par ailleurs, les approches monoclasses (visant l'extraction d'une seule classe thématique dans une image) permettent de focaliser les efforts sur la conception d'algorithmes exploitant des caractéristiques particulières de la classe recherchée. Notamment, (Musci *et al.*, 2013) affirme que la meilleure stratégie pour travailler avec les images à très haute résolution spatiale consiste à employer des segmentations séparées pour chaque classe étant donnée la grande variabilité des objets présents. Ainsi, chacune d'entre elles peut être paramétrée spécifiquement pour mieux retrouver les objets de la classe, menant ainsi à des meilleures performances.

Plusieurs méthodologies existent pour extraire une classe d'objets, comme des bâtiments (Liasis, Stavrou, 2016) ou des routes (Unsalan, Sirmacek, 2012 ; Lacoste *et al.*, 2010). Cependant, ces approches sont très spécifiques et très dépendantes des données, du type d'objets et du contexte applicatif.

Quelques approches plus générales pour l'extraction monoclasse ont également été proposées. Par exemple dans (Belarte *et al.*, 2013), les auteurs proposent une méthode pour l'apprentissage de règles de classification floues à partir d'exemples d'une classe fournis par un expert.

Une idée de plus en plus étudiée consiste à combiner les deux techniques (classification et segmentation), afin d'améliorer les deux résultats simultanément. (Farmer, Jain, 2005) présente un cadre de segmentation guidée par la classification. La particularité de cette approche est d'effectuer la segmentation de manière conjointe avec la classification, en utilisant la probabilité de classification comme métrique pour guider la segmentation. Les approches classiques effectuent la segmentation puis la classification de manière séquentielle, en supposant que la segmentation obtenue, utilisée pour la classification, est de qualité suffisante. En revanche, l'approche conjointe, effectue des allers-retours entre la segmentation et la classification, en utilisant la qualité de la classification pour mesurer la qualité de la segmentation et ainsi guider le processus. Néanmoins, dans cette approche qui ne recherche qu'un seul objet dans l'image, la position de celui-ci est approximativement connue. De plus, elle suppose qu'il existe un sous-ensemble de segments initiaux pour reconstruire l'objet d'intérêt. Cette idée a également été explorée dans (Sellaouti *et al.*, 2012). Les auteurs emploient un critère basé sur la classification pour sélectionner les graines pour amorcer un processus de segmentation par croissance de régions hiérarchique et itératif. Cette approche se base sur un modèle de classification pouvant être très spécifique au jeu de données et donc difficile à reproduire ou généraliser.

Ces méthodes supposent que la segmentation initiale est sur-segmentée : la classification est utilisée pour réduire celle-ci par fusion sans remettre en cause les segments initiaux. Dans le processus collaboratif de segmentation et de classification (CoSC) proposé par (Troya Galvis *et al.*, 2016), la classification est utilisée afin de modifier (fusion, scission, croissance, rétrécissement) directement les segments pour ainsi augmenter leur qualité et de fait, celle de la classification elle-même. De plus, ce processus est générique et peut être facilement adapté à des applications différentes.

Dans cet article, nous présentons une étude comparative de plusieurs implémentations de CoSC afin de mieux comprendre son fonctionnement, notamment les interactions entre les paradigmes de segmentation et de classification au sein de celui-ci. Nous en déduisons quelques règles pour bien configurer un processus CoSC, ainsi que des configurations intéressantes ayant un comportement régulier et satisfaisant.

La suite de l'article est structurée comme suit. La section 2, rappelle la définition générique du cadre collaboratif CoSC. La section 3, présente différentes manières d'instancier CoSC. La section 4, présente une étude comparative de 192 configurations possibles de CoSC dont les résultats sont analysés en termes d'optimisation, de temps

de calcul et de précision par rapport à des données de référence. La section 5 résume les résultats de cette expérience et énumère les faits les plus remarquables de celle-ci. Finalement, la section 6 apporte quelques perspectives de recherche envisageables.

## 2. Le cadre collaboratif CoSC

Le cadre collaboratif CoSC est spécialisé dans l'extraction d'une seule classe thématique  $C_k$  choisie par l'expert. Dans la suite, nous supposons disposer :

- d'une segmentation initiale  $\mathcal{S} = \{R_i \mid 0 < i < N_R\}$  où  $N_R$  est le nombre de segments et chaque segment  $R_i$  est défini par  $R_i = \{(x_k^i, y_k^i) \mid 0 < k < M_i\}$  avec  $M_i$  le nombre de pixels dans  $R_i$  ;
- d'un classifieur un-contre-tous  $\mathcal{C}_{C_k}$  entraîné pour discriminer la classe  $C_k$ .

Le processus que nous proposons, illustré dans la figure 1 consiste alors à sélectionner un segment selon un critère donné et à le modifier localement en fonction de sa qualité pour tenter d'améliorer sa classification. Ces deux étapes sont répétées jusqu'à convergence. La meilleure segmentation trouvée ainsi que la carte de probabilités qui lui est associée sont alors retournées.

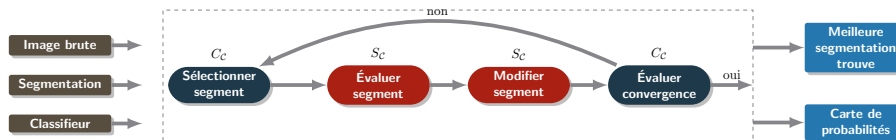


Figure 1. Schéma conceptuel de CoSC

### 2.1. Définitions

Nous posons maintenant quelques définitions nécessaires pour décrire formellement le cadre proposé.

DÉFINITION 1. — Un segmenteur spécialisé, noté  $S_{C_k}$ , est un agent de segmentation capable d'évaluer (par rapport aux caractéristiques de la classe  $C_k$ ) et de modifier localement un segment  $R_i$  donné.

DÉFINITION 2. — Un extracteur de classe, noté  $C_{C_k}$ , est un agent de classification capable de déterminer à partir de  $\mathcal{C}_{C_k}$ , la probabilité  $P_{C_k}(R)$  qu'un segment  $R \in \mathcal{S}$  appartienne à la classe  $C_k$ .

DÉFINITION 3. — Un agent de collaboration, noté  $SC_{C_k}$ , est un agent capable de gérer la collaboration entre un segmenteur spécialisé  $S_C$  et un extracteur de classe  $C_{C_k}$ .

DÉFINITION 4. — Soient deux seuils  $T_{\notin} < T_{\in}$  compris dans  $[0, 1]$ . Un segment  $R_i \in C_k$  (resp.  $R_i \notin C_k$ ) ssi  $P_{C_k}(R_i) \geq T_{\in}$  (resp.  $P_{C_k}(R_i) \leq T_{\notin}$ )<sup>1</sup>.

DÉFINITION 5. — Un segment  $R$  tel que  $T_{\notin} < P_{C_k}(R_i) < T_{\in}$  est dit ambigu.

## 2.2. Le processus CoSC

Le processus collaboratif, réalisé par  $SC_{C_k}$ , consiste à échanger des informations entre un segmenteur spécialisé  $S_{C_k}$  et un extracteur de classe  $C_{C_k}$  de sorte que :

1.  $S_{C_k}$  propose une segmentation initiale  $\mathcal{S}$  quelconque.
2. Une étape de modifications locales<sup>2</sup> est alternée avec une étape d'évaluation jusqu'à ce que la méthode converge :
  - (a) Dans l'étape de modification locale  $t$ ,  $C_{C_k}$  sélectionne un segment  $s_c$  suivant un critère donné et le communique au segmenteur. Le segmenteur  $S_C$  évalue  $s_c$  de manière locale pour déterminer si  $s_c$  est sur-, sous-, ou bien segmenté ; il modifie alors  $s_c$  en conséquence pour essayer d'améliorer sa qualité.
  - (b) Dans l'étape d'évaluation, la segmentation modifiée  $\mathcal{S}_{t+1}$  est transmise à l'extracteur  $C_{C_k}$  qui la classe et l'évalue de façon non-supervisée.
  - (c) Si le processus a convergé alors FIN (le meilleur couple segmentation-classification a été trouvée) sinon, l'étape 2 est ré-itérée.

Dans les sous-sections suivantes nous présentons en détail ces différentes étapes.

### 2.2.1. Sélection du segment candidat

À chaque itération, un segment  $s_c$  est choisi par l'extracteur  $C_{C_k}$  pour être candidat à modification. Il est crucial de sélectionner un "bon" segment candidat, autrement dit, un segment dont la modification pourrait réellement améliorer la classification. Nous verrons dans la section 3 que l'on peut mettre en œuvre plusieurs stratégies comme, par exemple, sélectionner un segment de manière aléatoire ou sélectionner un segment avec une probabilité très proche de  $T_{\in}$  ou de  $T_{\notin}$ .

### 2.2.2. Évaluation de la segmentation

En vue de pouvoir modifier le segment de sorte à minimiser les erreurs de segmentation, il est nécessaire d'évaluer le segment candidat en termes de segmentation. Pour déterminer si un segment est sur-, sous-, ou bien segmenté, nous employons la fonction d'évaluation locale proposée dans (Troya-Galvis *et al.*, 2015). Elle est définie par :

1. Ces deux seuils définissent donc une zone de rejet (Chow, 1970) sur  $\mathcal{C}_{C_k}$ .  
 2. Lors de la première itération, l'étape 2.a n'est pas effectuée :  $S_1 = S_0$ .

$$\phi_{\delta}(R_i) = \begin{cases} -1 & \text{si } H(R_i) > \delta \\ 1 & \text{si } H(R_i) \leq \delta \text{ et } \exists R_j \in \mathcal{N}(R_i) \mid H(R_i \cup R_j) \leq \delta \\ 0 & \text{sinon} \end{cases} \quad (1)$$

où  $R_i$  est le segment évalué,  $H$  est une fonction quantifiant l'homogénéité de  $R_i$ ,  $\delta$  est un seuil sur les valeurs de  $H$  et  $\mathcal{N}(R_i)$  est l'ensemble de segments adjacents à  $R_i$ . Dans nos expérimentations, l'indice d'homogénéité utilisé est l'entropie (Zhang *et al.*, 2003), et le seuil  $\delta$  est appris pour chaque classe thématique, à partir de quelques exemples donnés par un expert.

### 2.2.3. Modifications locales

On définit un opérateur de modification locale  $\mathcal{O}$  comme une fonction  $\mathcal{O} : \mathcal{D}_i \rightarrow \mathcal{D}_i$  où  $\mathcal{D}_i = R_i \cup \mathfrak{N}_{R_i}$  où  $\mathfrak{N}_{R_i}$  est l'ensemble de points appartenant aux segments adjacents à  $R_i$  (voisinage). Un opérateur peut consister par exemple à scinder un segment en deux sous-segments, ou à fusionner deux segments adjacents. Soient  $O_L$ ,  $U_L$ , et  $W_L$ , trois listes de tels opérateurs  $\mathcal{O}$ . Lors de l'étape de modification, le segment  $s_c$  est évalué. Selon qu'il soit sur- (resp. sous-, resp. bien) segmenté, un opérateur choisi aléatoirement parmi les opérateurs  $\mathcal{O}$  dans la liste  $O_L$ , (resp.  $U_L$ , resp.  $W_L$ ) est appliqué (cf. Algo. 1 et 2).

#### Algorithme 1. Appliquer modifications

---

```

Data: Segment:  $s_c$ , List of  $\mathcal{O}$ :  $L$ 
Result: Segment:  $R_m$ 
begin
   $L.shuffle()$ 
  while  $L.iterator.has\_next()$  do
     $R_m \leftarrow modify(s_c, L.iterator.next())$ 
    if  $R_m \neq s_c$  then
      | return  $R_m$ 
  return  $s_c$ 

```

---

### 2.2.4. Convergence

La fonction objectif à optimiser par CoSC quantifie la qualité de la solution courante de façon non-supervisée, c'est-à-dire, sans l'utilisation de connaissances externes (données de référence) qui sont souvent manquantes, incertaines ou incomplètes en télédétection. Cette fonction objectif peut être définie de plusieurs façons et prendre en compte, par exemple, des critères basés sur la classification, sur la segmentation, ou sur les deux, comme nous le montrons dans la section 3. Pour éviter une convergence prématurée, le processus autorise un nombre  $D$  de modifications dégradantes, en espérant sortir des minima locaux avec :

$$D = \frac{\mathcal{U} \times N_R}{3} \quad (2)$$

*Algorithme 2. Modification locale*


---

**Data:** Segment:  $s_c$   
**Result:** Segment:  $R_m$   
**begin**  
  **if** *sur-segmentation*( $s_c$ ) **then**  
    |  $R_m \leftarrow$  Appliquer modifications( $s_c, O_L$ )  
  **else if** *sous-segmentation*( $s_c$ ) **then**  
    |  $R_m \leftarrow$  Appliquer modifications( $s_c, U_L$ )  
  **else**  
    |  $R_m \leftarrow$  Appliquer modifications( $s_c, W_L$ )  
  **return**  $R_m$

---

où  $\mathcal{U}$  est le pourcentage de segments ambigus (cf. Def. 5). Quand le processus atteint  $D$  étapes consécutives sans amélioration, le processus effectue un retour en arrière vers la meilleure solution trouvée. Le processus collaboratif s'arrête lorsque plus aucune amélioration n'est apportée à la meilleure solution courante. Il est à noter que  $D$  diminue proportionnellement au nombre de segments ambigus. C'est en quelque sorte une procédure d'optimisation par recuit simulé simplifiée et adaptée aux particularités des données de télédétection.

### 2.3. Données en sortie

À l'issue du processus, le résultat se compose de la segmentation modifiée et d'une image de probabilité où chaque pixel représente la probabilité d'appartenance de celui-ci à la classe  $C_k$ .

En résumé, CoSC est par construction une méthode générique pouvant être appliquée facilement pour extraire les objets d'un type donné (par un expert généralement). Le processus repose sur des critères de qualité afin de choisir le segment candidat et pour appliquer les modifications locales à celui-ci.

## 3. Implémentation du processus CoSC

Afin d'étudier le comportement de CoSC, nous avons proposé et implémenté plusieurs variantes. Ainsi, nous avons étudié l'influence des différents paramètres sur le comportement de la méthode, ainsi que sur les résultats de celle-ci. Pour cela, nous avons testé l'ensemble des combinaisons possibles des différents paramètres sur quatre classes thématiques : Eau, Végétation, Bâtiments et Routes. Les résultats sont présentés en section 4.

### 3.1. Les différents paramètres

Une instance de CoSC est définie par quatre paramètres :

- le modèle de classification, noté  $P_c$  ;
- la stratégie de sélection du segment candidat, notée  $P_s$  ;
- la stratégie de modification, notée  $P_m$  ;
- la fonction objectif, notée  $P_e$ .

Dans ce qui suit nous rappelons le rôle de chaque paramètre et nous présentons les choix implémentés.

#### 3.1.1. Modèle de classement $P_c$

Le modèle de classement est défini par un apprentissage supervisé préalable et indépendant de CoSC. Il calcule la probabilité qu'un segment appartienne à la classe thématique recherchée. Deux types de modèles ont été testés durant cette étude :

1. Arbre de Hoefding ( $P_c = \text{VFDT}$ ) : des arbres incrémentaux de Hoefding ou VFDT (Hulten *et al.*, 2001) ont été entraînés à partir d'exemples et contre-exemples (entre 10 et 15) construits de manière interactive par un expert pour chaque classe considérée.
2. Perceptron multicouche ( $P_c = \text{MLP}$ ) : des perceptrons multicouches ou MLP (Haykin, 1998), avec deux couches cachées et 15 nœuds par couche, ont été entraînés à partir de données de référence. L'objectif à prédire est le pourcentage de la classe en question à l'intérieur de chaque segment.

Les segments sont caractérisés par 30 attributs différents, dont certains sont basés sur des propriétés spectrales telles que la moyenne ou l'écart type des valeurs sur chaque bande ; et d'autres sont basés sur des propriétés spatiales comme des indices de compacité ou d'élongation. Aucune procédure d'optimisation sur les méta-paramètres de ces algorithmes ni de sélection d'attributs n'a été effectuée. Les modèles de prédiction obtenus ne sont donc probablement pas optimaux, mais ils ont été suffisants pour démontrer l'applicabilité de la méthode proposée.

#### 3.1.2. Sélection du segment candidat $P_s$

À chaque étape de la collaboration, un segment  $s_c$  est sélectionné pour être modifié. Nous avons implémenté quatre stratégies de sélection différentes.

1. Par sélection aléatoire ( $P_s = s_{alea}$ ): consiste à choisir  $s_c$  de manière aléatoire parmi tous les segments de  $\mathcal{S}$ .

$$s_{alea} = \mathcal{S}[\mathcal{U}\{0, N\}] \quad (3)$$

où  $\mathcal{U}$  dénote une distribution uniforme et  $N$  est le nombre de segments dans  $\mathcal{S}$ .

2. Au moins ambigu ( $P_s = s_{amb}^-$ ): consiste à choisir le segment le plus proche de  $T_\epsilon$  ou de  $T_\zeta$ .

$$s_{amb}^- = \arg \min_R (|P_C(R) - T_\epsilon|, |P_C(R) - T_\zeta|) \quad (4)$$



3. Au plus ambigu ( $P_s = s_{amb}^+$ ): consiste à choisir le segment le plus éloigné de  $T_\in$  et  $T_\notin$ .

$$s_{amb}^+ = \arg \min_R P_C(R) - \frac{T_\in + T_\notin}{2} \quad (5)$$

4. Au pire segmenté ( $P_s = s_{seg}$ ): consiste à employer l'évaluation locale à granularité fine de  $UOA$  (Eq. 7).

$$s_{seg} = \arg \max_R \phi_{\delta_{fine}}(R) \quad (6)$$

La fonction  $\phi_{\delta_{fine}}$  est définie par :

$$\phi_{\delta_{fine}}(R) = \begin{cases} \text{under}(R) & \text{si } \phi_\delta(R) = -1 \\ \text{over}(R) & \text{si } \phi_\delta(R) = 1 \\ 1 - \text{isolation}(R) & \text{sinon} \end{cases} \quad (7)$$

avec

$$\text{over} = \frac{N_v}{|\mathcal{N}(R_i)|} \quad (8)$$

$$N_v = |\{R_j \in \mathcal{N}(R_i) \mid H(R_i \cup R_j) \leq \delta\}|$$

où  $N_v$  est le nombre de segments voisins considérés comme trop similaires à  $R_i$  par rapport à  $\delta$ .

$$\text{under} = \frac{H(R_i) - \delta}{1 - \delta} \quad (9)$$

et

$$\text{isolation} = 1 - \frac{H(R_i)}{\delta} \quad (10)$$

### 3.1.3. Stratégie de modification $P_m$

Une stratégie de modification consiste à définir les trois listes d'opérateurs de modification  $O_L$ ,  $U_L$  et  $W_L$  qui seront appliqués au segment candidat  $s_c$  à chaque étape de la collaboration. Pour rappel,  $O_L$  doit contenir des opérateurs visant à réduire la sur-segmentation,  $U_L$  doit contenir des opérateurs visant à réduire la sous-segmentation et  $W_L$  peut contenir des opérateurs quelconques qui sont appliqués lorsque  $s_c$  n'est pas détecté comme sur- ou sous-segmenté. À noter, que même si  $s_c$  apparaît comme "bien" segmenté, il s'avère tout de même intéressant de le perturber pour tenter d'améliorer le résultat global, notamment, si la sélection est faite en fonction de la probabilité de classification, dans ce cas,  $s_c$  est *a priori* difficile à classifier. Nous avons implémenté quatre opérateurs de modification.

**Fusion** : cet opérateur vise principalement à réduire la sur-segmentation. Il cherche à fusionner  $s_c$  avec le segment adjacent dont la distance euclidienne calculée sur les moyennes de bandes radiométriques est minimale.

**Scission** : cet opérateur vise principalement à réduire la sous-segmentation. Il consiste à appliquer la méthode de super-pixels SLIC (Achanta *et al.*, 2012) contrainte

aux pixels de  $s_c$ . SLIC favorise la création de segments plutôt compacts et qui adhèrent aux contours présents dans l'image. Le nombre de nouveaux segments créés est calculé en fonction de la taille de  $s_c$  : plus il est grand, plus le nombre de nouveaux segments sera élevé.

**Rétrécissement** : cet opérateur vise à réduire la sous-segmentation. Il consiste à appliquer une opération d'érosion morphologique sur  $s_c$  en utilisant comme élément structurant un carré  $3 \times 3$ . Ces pixels érodés sont redistribués aux voisins spatialement plus proches, afin de conserver une partition de l'image. Si l'érosion entraîne une rupture de la connexité de  $s_c$  l'opération est refusée afin de respecter la définition de la segmentation.

**Croissance** : cet opérateur vise principalement à réduire la sur-segmentation. Il consiste à appliquer une opération de dilatation morphologique sur  $s_c$  en utilisant comme élément structurant un carré  $3 \times 3$ . Ces pixels dilatés sont retranchés aux segments auxquels ils appartenaient, afin de conserver une partition de l'image. Si la dilatation entraîne un changement dans la topologie de  $s_c$  ou de ses voisins, l'opération est refusée afin de toujours respecter la définition de la segmentation.

En partant de ces opérateurs de base, nous implémentons les deux stratégies de modification en définissant deux ensembles de listes différents.

1. Stratégie sans scission ( $P_m = M_F$ ):
  - $O_L$  : Fusion ;
  - $U_L$  : Rétrécissement ;
  - $W_L$  : Fusion, Rétrécissent, Croissance.
2. Stratégie avec scission ( $P_m = M_S$ ):
  - $O_L$  : Fusion ;
  - $U_L$  : Scission ;
  - $W_L$  : Rétrécissent, Croissance.

L'application de ces stratégies sont équivalentes aux algorithmes 3 et 4.

#### 3.1.4. Fonction objectif $P_e$

Nous avons défini et étudié trois fonctions différentes.

1. Critère basé sur la classification ( $P_e = Q_{clsf}$ ) : cette fonction se base sur la probabilité de classification des segments et cherche intuitivement à réduire le nombre de segments ambigus (cf. Déf. 5), c'est-à-dire, les segments pour lesquels le classifieur n'arrive pas à déterminer avec certitude l'appartenance ou non à la classe en question. Ce critère est la probabilité moyenne qu'un segment appartienne ou non à la classe de manière sûre. Il est défini par :

*Algorithme 3.* Stratégie de modification sans scission.

---

**Data:** Segment  $s$   
**Data:** Liste de Segments  $neighbours$   
**Result:** Segment modifié  $s'$   
**begin**  
  **if**  $s$  est sur-segmenté **then**  
    └ Fusionner  $s$  avec son voisin le plus similaire  
  **else if**  $s$  est sous-segmenté **then**  
    └ Rétrécir  $s$   
  **else**  
    └ Appliquer à  $s$  un opérateur pris au hasard parmi fusion, rétrécissement ou croissance

---

*Algorithme 4.* Stratégie de modification avec scission.

---

**Data:** Segment  $s$   
**Data:** List of Segment  $neighbours$   
**Result:** Modified Segment  $s'$   
**begin**  
  **if**  $s$  est sur-segmenté **then**  
    └ Fusionner  $s$  avec son voisin le plus similaire  
  **else if**  $s$  est sous-segmenté **then**  
    └ Diviser  $s$  en appliquant la méthode SLIC  
  **else**  
    └ Appliquer à  $s$  un opérateur au hasard parmi rétrécissement ou croissance

---

$$Q_{clsf} = \frac{1}{N_R} \left( \sum_{i|P_{C_k}(R_i) > T_\epsilon} P_{C_k}(R_i) + \sum_{i|P_{C_k}(R_i) < T_\zeta} (1 - P_{C_k}(R_i)) \right) \quad (11)$$

où  $N_R$  est le nombre de segments dans la segmentation courante. Ainsi, plus la classe d'un segment  $R_i$  est certaine, plus  $R_i$  contribue au score. *A contrario*, les segments ambigus sont pénalisés de manière implicite puisqu'ils ne contribuent pas au score, mais ils sont pris en compte dans le facteur  $\frac{1}{N_R}$ . Le cas idéal où tous les segments sont classés avec 100% de certitude donne  $Q_{clsf} = 1,0$ . Dans le pire des cas, où tous les segments sont ambigus, alors  $Q_{clsf} = 0,0$ .

2. Critère basé sur la segmentation ( $P_e = Q_{seg}$ ) : cette fonction se base sur la métrique  $UOA_{L_2}$  (Troya-Galvis *et al.*, 2015). Elle ne tient compte que de la

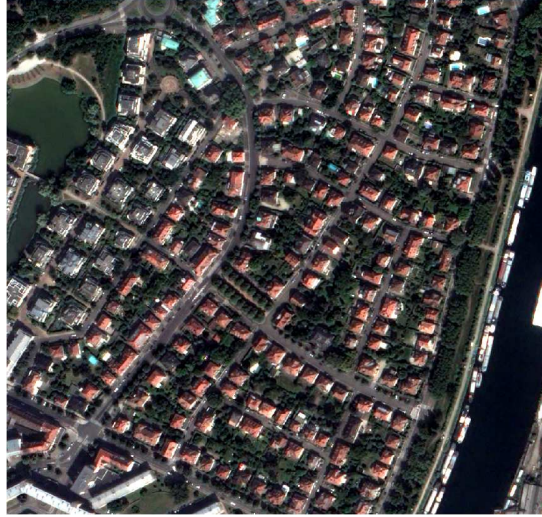


Figure 2. Image étudiée

segmentation et favorise donc la réduction des erreurs de segmentation. Elle est définie par :

$$Q_{seg} = 1 - UOAL_2 \quad (12)$$

$$UOAL_2 = \sqrt{\Psi^2 + \Theta^2} \quad (13)$$

où  $\Psi$  et  $\Theta$  représentent les taux de sous- et sur-segmentation respectivement.

3. Critère de qualité mixte ( $P_e = Q_{mix}$ ): cette fonction cherche à améliorer simultanément la classification et la segmentation en combinant les deux critères définis précédemment. C'est la moyenne harmonique de  $Q_{clsf}$  et  $Q_{seg}$ . Elle est définie comme suit :

$$Q_{mix} = 2 \cdot \frac{Q_{clsf} \cdot Q_{seg}}{Q_{clsf} + Q_{seg}} \quad (14)$$

#### 4. Étude du comportement de CoSC en fonction de ses paramètres

Dans cette étude, nous avons employé l'image présentée dans la figure 2, c'est un extrait d'image Pléiades de 1122 par 1085 pixels et 4 bandes radiométriques. Les données de référence (figure 3) ont été fournies par un expert géographe. Chaque pixel représente la probabilité que celui-ci appartienne à la classe recherchée. Ainsi, nous avons choisi d'utiliser la métrique de précision floue (*fuzzy accuracy*) (Binaghi *et al.*, 1999) pour mesurer les performances de chaque variante testée.

La méthodologie expérimentale est la suivante. Pour chaque permutation possible des différents paramètres (deux pour  $P_c$ , deux pour  $P_m$ , trois pour  $P_e$  et quatre pour

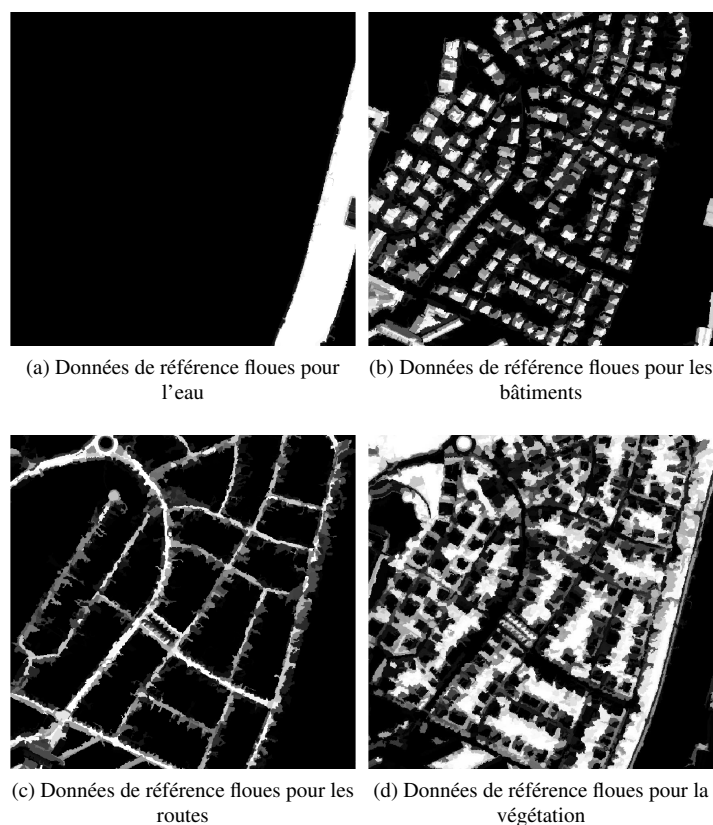


Figure 3. Données de référence floues. L'intensité du niveau de gris correspond à la probabilité d'appartenance à la classe

$P_s$ ) et pour chacune des quatre classes, nous avons instancié un processus CoSC et avons observé l'évolution de la fonction objectif au fil du temps (en termes d'itérations), le temps d'exécution et la précision floue. De fait, 192 instances différentes de CoSC ont donc été testées. Il est à noter que le coût en termes de temps de calcul pour certaines instances est assez élevé. Ainsi, nous n'avons effectué qu'une observation par instance : nous avons préféré explorer un maximum de possibilités lors de cette étude afin de mieux comprendre l'effet des différents paramètres et ainsi repérer des configurations intéressantes. Nous étudions principalement trois aspects : l'optimisation du critère d'évaluation, le temps d'exécution et la performance par rapport aux données de référence.

- Les figures 4, 5, 6 et 7 montrent l'évolution de la fonction objectif optimisée par CoSC au fil des itérations pour un type de classifieur et une stratégie de modification donnée. Chaque colonne correspond à une fonction objectif particulière ( $Q_{clsf}$ ,  $Q_{mix}$

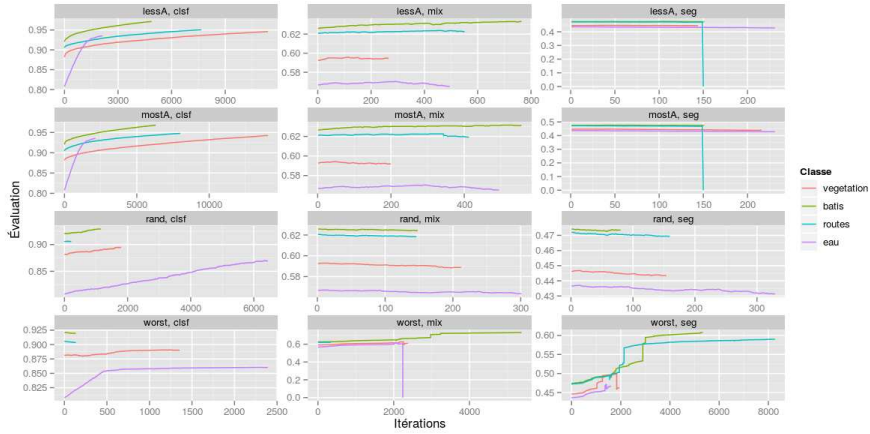


Figure 4. Comportement de CoSC pour  $P_c = VFDT$  et  $P_m = M_F$

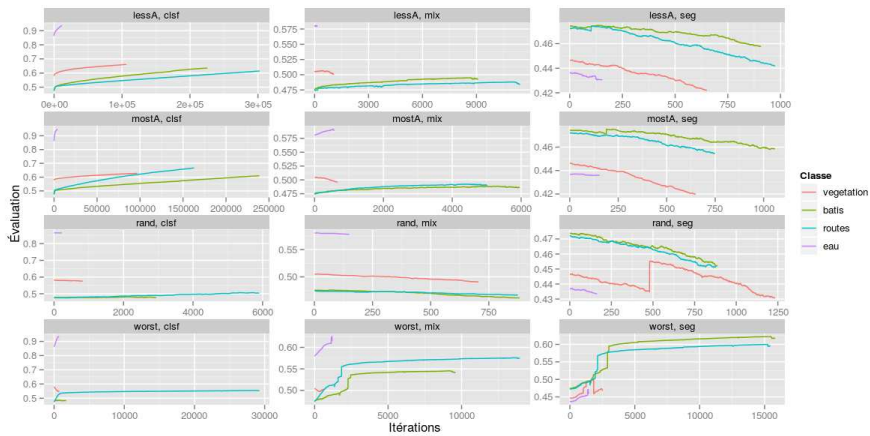


Figure 5. Comportement de CoSC pour  $P_c = MLP$  et  $P_m = M_F$

et  $Q_{seg}$ , notées clsf, mix et seg respectivement dans la figure); chaque ligne correspond à une stratégie de sélection ( $s_{amb}^-$ ,  $s_{amb}^+$ ,  $s_{alea}$  et  $s_{seg}$ , notées lessA, mostA, rand et worst respectivement dans la figure). Dans chaque graphe, les abscisses représentent le nombre d'itérations et les ordonnées représentent le score de la fonction d'évaluation. Il est à noter que l'échelle est différente sur chaque graphe. En effet, chaque fonction objectif mesure des propriétés différentes et pas comparables. Par conséquent, les scores moyens de chaque fonction objectif différent, et l'échelle est choisie pour faciliter l'analyse du comportement de CoSC à chaque cas.

Par exemple le sous-graphe en haut à gauche de la figure 4 correspond aux paramètres  $P_c = VFDT$ ,  $P_m = M_F$ ,  $P_e = Q_{clsf}$  et  $P_s = s_{amb}^-$ . Dans ce cas, on observe

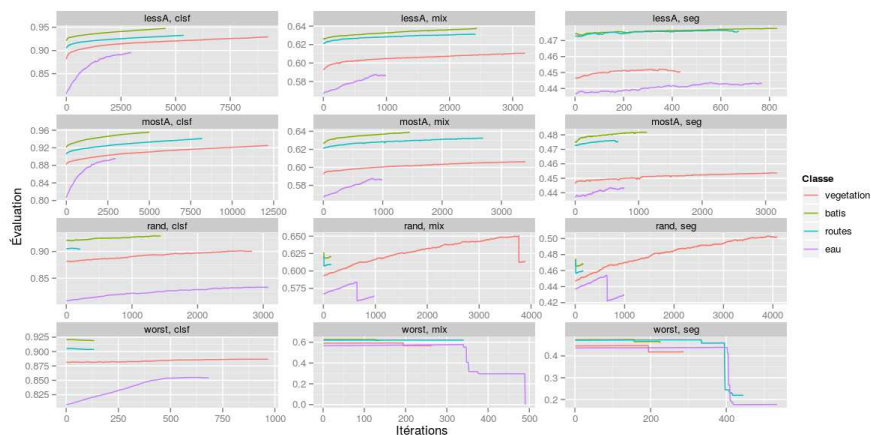


Figure 6. Comportement de CoSC pour  $P_c = VFDT$  et  $P_m = M_S$

que les courbes des quatre classes évolue de manière plus ou moins logarithmique, les courbes sont croissantes de manière presque linéaire pendant un certain nombre d'itérations au début, puis elles croissent plus lentement jusqu'à l'arrêt du processus. On retrouve ce même comportement de manière plus ou moins marquée dans la quasi totalité des cas où  $P_e = Q_{clsf}$ . On peut remarquer également que le temps de convergence est lié au modèle de classement ainsi qu'à la classe extraite. En effet, lors que  $P_c = MLP$  le processus d'extraction de la classe eau converge plus rapidement que les autres, alors que avec  $P_c = VFDT$  c'est souvent le processus d'extraction de la classe routes ou de la classe bâtis qui convergent plus rapidement. Par ailleurs, le comportement de CoSC avec la fonction basée sur la segmentation ( $P_e = Q_{seg}$ ) semble erratique mais on y trouve tout de même quelques régularités. Par exemple, lorsque  $P_m = M_F$  seule la stratégie de sélection au pire segmenté ( $P_s = s_{seg}$ ) permet une amélioration remarquable de la fonction objectif. En revanche, quand  $P_m = M_S$  on observe l'effet inverse. Les processus avec sélection du pire segmenté semble stagner pendant plusieurs itérations puis la fonction objectif chute brusquement avant l'arrêt du processus, alors que les processus mettant en place d'autres stratégies de sélection permettent d'améliorer légèrement la fonction  $Q_{seg}$ . Une autre particularité à noter est que les valeurs et les courbes des processus utilisant la fonction objectif  $Q_{mix}$  reflètent bien qu'il s'agit d'une combinaison linéaire de  $Q_{clsf}$  et  $Q_{seg}$ . Elles ne présentent pas de comportements différents à ceux observés avec les deux autres fonctions objectif.

- La figure 8 montre le temps d'exécution de chaque processus pour les différentes configurations testées. Chaque colonne correspond à une stratégie de sélection, chaque ligne correspond à une fonction objectif et chaque sous-graphe correspond à une classe. La couleur de chaque cellule varie de blanc (temps d'exécution faible) à violet (temps d'exécution élevé). On remarque que les processus faisant intervenir l'évaluation de la segmentation (soit avec  $P_e = Q_{mix}$  ou  $P_e = Q_{seg}$  soit avec  $P_s = s_{seg}$ ) ont généralement un temps d'exécution plus important. D'autres para-

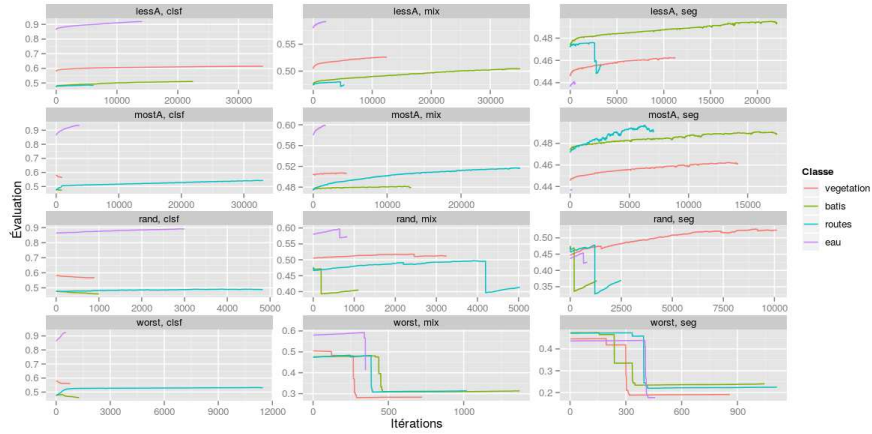


Figure 7. Comportement de CoSC pour  $P_c = MLP$  et  $P_m = M_S$

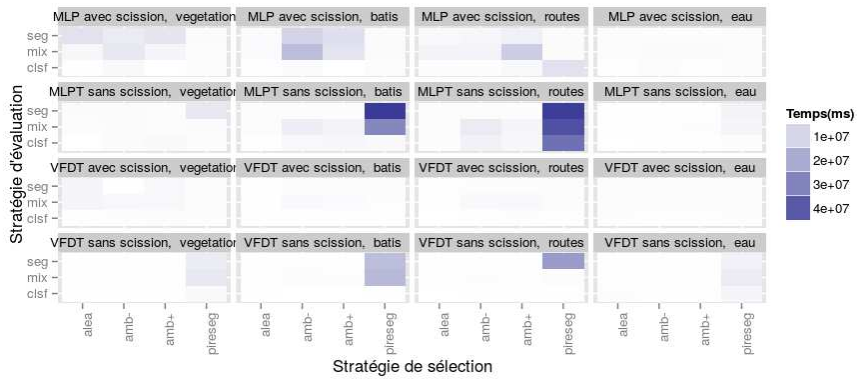


Figure 8. Temps d'exécution (millisecondes) de CoSC pour les différentes configurations testées

mètres tels que le couple (modèle de classement, classe) ou la stratégie de modification semblent avoir également un effet (moins évident) sur le temps d'exécution.

- La figure 9 présente la précision floue calculée par rapport aux données de référence. Il n'y a pas de patron évident qui se dégage de ces figures. On constate tout de même que tous les résultats se situent entre 0,730 (obtenu pour l'extraction des routes avec  $P_c = VFDT$ ,  $P_m = M_S$ ,  $P_e = Q_{seg}$  et  $P_s = s_{seg}$ ) et 0,926 (obtenu pour l'extraction de l'eau avec  $P_c = MLP$ ,  $P_m = M_S$ ,  $P_e = Q_{seg}$  et  $P_s = s_{seg}$ ). On remarque que ces deux configurations ne diffèrent que par le modèle de classement et la classe extraite. Cela laisse à penser que ce couple de paramètres influe de manière importante sur les résultats en termes de classification. En effet, si le modèle fait une classification



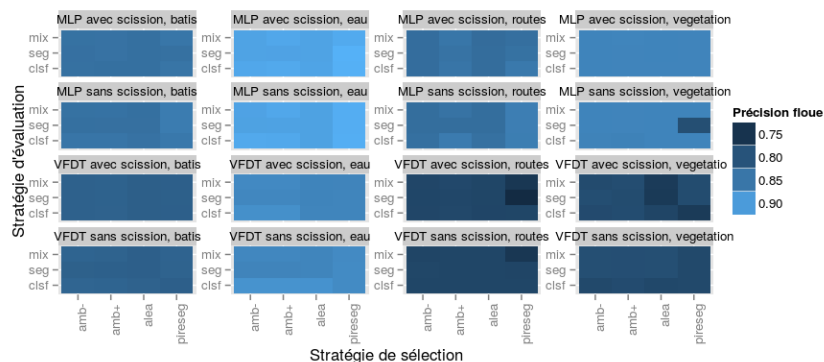


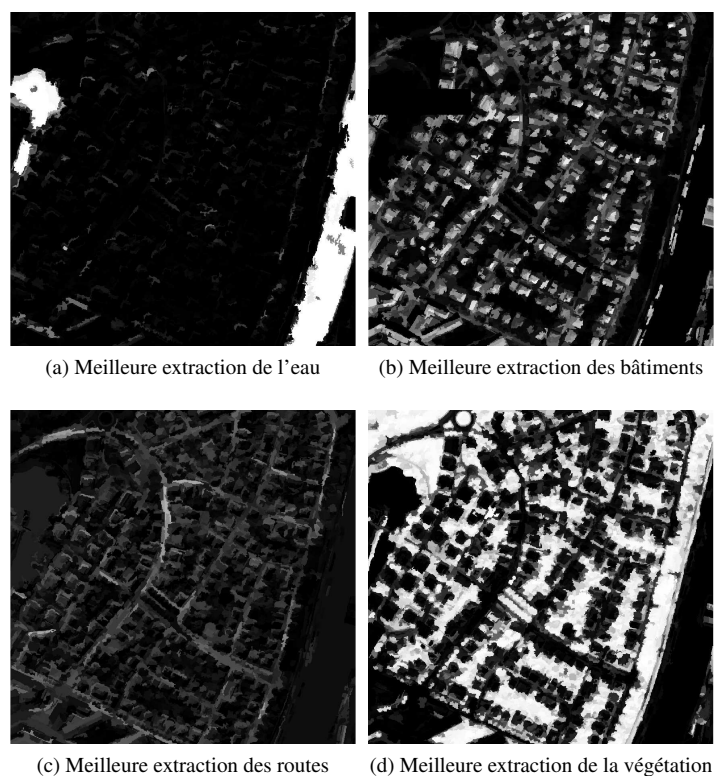
Figure 9. Précision floue pour chaque configuration testée

presque parfaite dès la première étape, alors CoSC converge très rapidement, mais sa classification sera très performante tout de même. Au contraire, si le modèle établit des prédictions aléatoires ou très peu pertinentes, CoSC va itérer un grand nombre de fois sans jamais réussir à améliorer la classification. À titre d'exemple, nous montrons dans la figure 10 les meilleurs résultats obtenus d'après la précision floue pour les quatre classes étudiées. Nous observons que les prédictions pour l'eau et la végétation sont effectivement plus certaines que pour les bâtiments et les routes. Cela peut expliquer les différents résultats en termes de précision floue pour chaque classe.

## 5. Discussion

Comme nous avons vu dans la section précédente, le comportement de CoSC varie principalement en fonction de la fonction objectif utilisée, mais aussi en fonction du modèle de classification (et de la classe extraite). De cette analyse nous pouvons tirer plusieurs conclusions :

- Le choix de la stratégie de sélection doit être fait en accord avec la fonction objectif employée. En effet, il est nécessaire de définir des stratégies permettant de sélectionner les segments les plus à même de maximiser le critère d'évaluation choisi, par exemple le plus ou le moins ambigu pour le critère basé sur la classification, ou le moins bien segmenté pour le critère basé sur la segmentation.
- À qualité égale, le type du modèle de classification n'a pas l'air d'influencer le comportement de CoSC, mais il a un impact sur le temps d'exécution. En effet les modèles plus complexes demandent plus de calculs qui se traduisent par de temps d'exécution plus élevés.
- Indépendamment du type du modèle, la qualité de celui-ci (en terme de classification) peut avoir une influence considérable sur le comportement de la méthode mais aussi sur le temps d'exécution. En effet, un modèle faisant des



*Figure 10. Meilleures classifications monoclasses obtenues avec des stratégies différentes*

prédictions avec une très haute confiance va entraîner une convergence plus rapide du processus. En revanche, un modèle faisant des prédictions très incertaines résultera en un processus beaucoup plus lent à converger et à des faibles améliorations du critère d'évaluation.

- La stratégie de modification avec scission ne semble pas adaptée pour améliorer le critère de segmentation, soit parce qu'elle sur-segmente trop rapidement l'image, soit parce qu'elle entraîne des modifications de segment inappropriées.

Nous observons que de manière générale l'extraction de la classe Eau obtient une meilleure précision floue que les autres classes. Généralement, le modèle MLP obtient de meilleures performances que le modèle VFDT pour toutes les classes, mais nous ne remarquons pas de couple de stratégies gagnant systématiquement. Pour le modèle VFDT ce sont souvent les stratégies de sélection du plus ambigu ou du moins ambigu qui semblent les plus performantes. Pour le modèle MLP c'est souvent la stratégie de sélection du moins bien segmenté la plus performante. La stratégie de sélection

aléatoire obtient, dans la plupart des cas, des performances plus faibles. Le critère combiné s'avère souvent une bonne option mais comme nous l'avons vu dans la section 4 il est très coûteux et demande des temps d'exécution plus élevés que les autres critères d'évaluation. En somme, une configuration qui nous semble des plus stables avec des temps de calcul raisonnables est  $P_e = Q_{clsf}$ ,  $P_s = s_{amb}^+$  et  $P_m = M_F$  avec le modèle de classement le plus adaptée à la classe à extraire.

## 6. Conclusion

Les paradigmes de segmentation et de classification jouent un rôle crucial en analyse d'images de télédétection. Le cadre collaboratif CoSC fournit un moyen générique d'extraire une classe thématique d'intérêt grâce à l'interaction de techniques de segmentation et de classification en tenant compte de la qualité des résultats de ces deux paradigmes. Nos expériences ont montré que CoSC peut être adapté facilement à l'extraction de classes d'intérêt variées. Deux perspectives de recherche en découlent naturellement. La première consiste à étendre le cadre CoSC pour effectuer une analyse multiclasse d'images de télédétection. En effet, il semble tout à fait possible de s'inspirer des techniques d'apprentissage par ensemble afin d'agrèger plusieurs instances de CoSC différentes mais complémentaires pour obtenir une carte de labels complète de l'image. La seconde vient de la généralité de CoSC, il serait envisageable de transcrire d'autres méthodes collaboratives existantes dans le formalisme de CoSC afin de les comparer avec nos implémentations particulières et à plus long terme élaborer un formalisme plus théorique sur les méthodes collaboratives segmentation-classification.

### Remerciements

*Ces travaux de recherche ont été financés par l'Agence Nationale de la Recherche dans le cadre du projet COCLICO (ANR-12-MONU-0001).*

## Bibliographie

- Achanta R., Shaji A., Smith K., Lucchi A., Fua P., Süsstrunk S. (2012). SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, p. 2274–2282.
- Belarte B., Wemmert C., Forestier G., Grizonnet M., Weber C. (2013). Learning fuzzy rules to characterize objects of interest from remote sensing images. *2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS*, p. 2986-2989.
- Binaghi E., Brivio P. A., Ghezzi P., Rampini A. (1999). A fuzzy set-based accuracy assessment of soft classification. *Pattern Recognition Letters*, vol. 20, n° 9, p. 935 - 948.
- Blaschke T. (2010). Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 65, p. 2–16.
- Chow C. (1970). On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, vol. 16, n° 1, p. 41-46.

- Coppin P., Jonckheere I., Nackaerts K., Muys B., Lambin E. (2004). Digital change detection methods in ecosystem monitoring: a review. *International Journal of Remote Sensing*, vol. 25, p. 1565–1596.
- Farmer M., Jain A. (2005). A wrapper-based approach to image segmentation and classification. *IEEE Transactions on Image Processing*, vol. 14, p. 2060–2072.
- Gao Y., Mas J. F., Kerle N., Navarrete Pacheco J. A. (2011). Optimal region growing segmentation and its effect on classification accuracy. *International Journal of Remote Sensing*, vol. 32, n° 13, p. 3747–3763.
- Haykin S. (1998). *Neural networks: A comprehensive foundation* (2nd éd.). Upper Saddle River, NJ, USA, Prentice Hall PTR.
- Hulten G., Spencer L., Domingos P. (2001). Mining time-changing data streams. *International Conference on Knowledge Discovery and Data Mining - SIGKDD*, p. 97–106.
- Lacoste C., Descombes X., Zerubia J. (2010). Unsupervised line network extraction in remote sensing using a polyline process. *Pattern Recognition*, vol. 43, n° 4, p. 1631 - 1641.
- Liasis G., Stavrou S. (2016). Building extraction in satellite images using active contours and colour features. *International Journal of Remote Sensing*, vol. 37, n° 5, p. 1127–1153.
- Musci M., Feitosa R., Costa G. (2013, April). An object-based image analysis approach based on independent segmentations. *Joint Urban Remote Sensing Event (JURSE)*, p. 275-278.
- Pham H. M., Yamaguchi Y., Bui T. Q. (2011). A case study on the relation between city planning and urban growth using remote sensing and spatial metrics. *Landscape and Urban Planning*, vol. 100, p. 223–230.
- Sellaouti A., Hamouda A., Deruyver A., Wemmert C. (2012). Hierarchical classification-based region growing (HCBRG): a collaborative approach for object segmentation and classification. *International Conference Image Analysis and Recognition*, p. 51–60.
- Troya Galvis A., Gancarski P., Berti-Équille L. (2016). Un cadre collaboratif pour la segmentation et la classification d'images de télédétection. *Extraction et Gestion des Connaissances (EGC 2016)*.
- Troya-Galvis A., Gancarski P., Passat N., Berti-Équille L. (2015, mai). Unsupervised quantification of under and over segmentation for object based remote sensing image analysis. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, n° 5, p. 1936–1945.
- Unsalan C., Sirmacek B. (2012, Nov). Road network detection using probabilistic and graph theoretical methods. *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, n° 11, p. 4441-4453.
- Westen C. V. (2013). Remote Sensing and GIS for Natural Hazards Assessment and Disaster Risk Management. *Treatise on geomorphology*, p. 259–298. Academic Press.
- Zhang H., Fritts J. E., Goldman S. A. (2003). An entropy-based objective evaluation method for image segmentation. *Electronic imaging 2004*, p. 38–49.