

# **Nouvelle méthode heuristique pour la découverte automatique des classes thématique dans les images satellitaires multispectrales. Application au milieu urbain de la ville d'Alger.**

KHEDAM Radja§, BELHADJ-AISSA Aichouche§

§ Laboratoire de Traitement d'Images et Rayonnements (LTIR), Faculté d'Electronique et d'Informatique (FEI), Université des Sciences et de la Technologie Houari Boumediene (USTHB). BP. 32, El Alia, Bab Ezzouar, 16111, Alger, Algérie. Téléphone : +213(21)247950, poste 806, Télécopieur : +213(21)247187. [radjakhedam@lycos.com](mailto:radjakhedam@lycos.com), [h.belhadj@lycos.com](mailto:h.belhadj@lycos.com)

Mots-clés : classification non dirigée, contexte spatial, optimisation, colonie de fourmis, convergence.

## **Introduction**

Pour classer d'une manière automatique les pixels d'une image satellitaire en un nombre de classes thématiques, de nombreux algorithmes classiques, tels que les centres mobiles ou Isodata, requièrent qu'une partition initiale soit donnée en entrée. C'est l'inconvénient majeur de ces méthodes : la partition obtenue à partir de cette initialisation risque d'être localement optimale et le seul moyen d'y remédier est de relancer la méthode avec une partition initiale différente. En outre, le nombre de classes exigé par ces méthodes diminue leur intérêt pour un expert cherchant justement à connaître ce nombre. AntClass (Monmarché, 2000) est un nouvel algorithme de classification non dirigée. Il utilise les principes exploratoires stochastiques d'une colonie de fourmis, pour découvrir automatiquement les classes dans des données numériques sans connaître le nombre de classes *a priori*, sans partition initiale et sans paramétrages délicat. En effet, dans la nature, les fourmis réelles offrent un modèle stimulant pour le problème du partitionnement et de classification. La constitution de cimetières, le tri collectif du couvain quand il est dérangé, et le rassemblement des œufs en fonction de leur état de développement, sont les exemples les plus marquants (Deneubourg *et al.*, 1990) (voir figure 1). Les règles utilisées par les fourmis sont relativement simples : lorsqu'une fourmi rencontre un élément du couvain, la probabilité qu'elle s'en empare est d'autant plus grande que cet élément est isolé. Lorsqu'une fourmi transporte un élément du couvain, elle le dépose avec une probabilité d'autant plus grande que la densité d'éléments du même type dans le voisinage est grande (voir figure 2). Ces probabilités ont été clairement formalisées et explicitées (Lumer *et al.*, 1994, Monmarché, 2000). Encouragés par les résultats positifs et pertinents obtenus sur des données artificielles non spatiales, nous proposons dans ce travail de recherche, l'adaptation et l'application de l'algorithme AntClass pour la découverte automatique du nombre de classes thématiques dans une image satellitaire multi spectrale. Cette image, acquise en 2001 par le capteur ETM+ du satellite Landsat-7, couvre le milieu urbain de la ville d'Alger (Algérie). La projection de certains échantillons appartenant aux classes obtenues sur l'espace des caractéristiques spectrales, nous permet d'identifier la nature thématique de ces classes et de valider leur nombre.

## **Algorithme AntClass adapté**

Nous avons adapté AntClass aux pixels d'une image multi spectrale comme suit : la colonie de fourmis est disposée suivant une loi aléatoire uniforme sur l'image à classifier. Cette dernière est assimilée à la grille sur laquelle se déplaceront les fourmis. Le ramassage et le dépôt des pixels ne sont pas effectués d'une manière physique mais virtuellement suivant des

mesures de probabilités calculées dans le voisinage spatial (contexte spatial) du pixel sur lequel la fourmi est déposée. Quant à la mesure de similarité entre deux pixels, nous avons utilisé la distance radio métrique euclidienne. Nous avons considéré les paramètres  $T_{create}$  et  $T_{remove}$  (compris entre 0 et 1) qui permettent d'obtenir des classes plus homogènes, ce qui limite les erreurs de classification. Ces deux paramètres sont déterminés d'une manière empirique et la capacité de transport d'une fourmi est égale à 1. Cette première étape est un processus itératif qui s'achève lorsque le taux d'exploration des pixels est de 100% ou égal à un certain seuil. A la fin de cette étape, une image étiquetée est obtenue. Cependant, cette image risque de présenter deux problèmes : 1) un nombre élevé de classes, 2) le déplacement des fourmis étant aléatoire, il y a risque de présence de pixels non classifiés (pixels libres). La réutilisation de AntClass peut s'avérer judicieuse pour la résolution du premier problème, c'est-à-dire, réduire le nombre de tas (ou classes). Mais, vu que le déplacement des fourmis reste strictement aléatoire, le problème des pixels libres pourrait réapparaître. De ce fait, nous proposons une deuxième phase d'exploration durant laquelle un autre algorithme à base de fourmis est appliqué, tel que : 1) la colonie se compose d'une seule fourmi, 2) le déplacement de la fourmi est déterministe, 3) la fourmi dispose d'une mémoire interne propre pour cibler directement les pixels libres, 4) la capacité de transport de la fourmi est infinie, la fourmi devient capable de manipuler des tas d'objets, 5) les règles heuristiques de base sont adaptées pour permettre à la fourmi de manipuler des tas entiers, 6) les paramètres  $T_{create}$  et  $T_{remove}$  sont aussi utilisés.

## Résultats et commentaires

Nous avons fixé  $T_{create} = 0.008$  et  $T_{remove} = 0.96$  et nous avons analysé l'influence du nombre de fourmis sur le nombre des classes trouvées et le nombre de pixels libres (voir figure 3). Avec 250 fourmis, la plus grande partie des pixels est classifiée dans l'une des 123 classes découvertes (figure 4). Les 0.8 % pixels libres dont la majorité est située sur les bords, sont étiquetés au second tour (figure 5) durant lequel la partition finale est aussi obtenue par l'agrégation des classes similaires (figure 6) et l'arrivée ainsi, à 07 classes bien séparables. Cependant, et comme nous le constatons sur la figure 7, avec  $T_{create} = 0.016$  et  $T_{remove} = 0.56$ , le résultat obtenu présente 05 classes dont une est prédominante, ce qui, contrairement aux 07 classes, ne concorde pas avec la réalité de terrain de cette zone. Les figures 8 et 9 nous permettent d'identifier ces 07 classes et qui sont : urbain dense (UD), urbain moins dense avec deux types (UMD1, UMD2), sol nu (SN1), végétation moins dense mélangée avec du sol (VMD+SN2), végétation dense (VD), et une classe hétérogène (CH) qui présente une confusion (CH1+CH2) au niveau du canal 4 (voir figure 10).

## Conclusion

AntClass est un algorithme de classification automatique intéressant qui peut être implanté sur une architecture parallèle. Cependant, il serait judicieux de calculer les paramètres  $T_{create}$  et  $T_{remove}$  d'une manière adaptative, ou alors les considérer comme des variables incrémentales avec le passage de chaque fourmi.

## Bibliographie

DENEUBOURG, J.L., GOSS, S., FRANKS, N., SENDOVA-FRANKS, A., DETRAIN, C. and CHRETIEN, L., 1990, The dynamic of collective sorting robot-like ants and ant-like robots. In (Meyer and Wilson), 356-365.

- LUMER, E. and FAIETA, B., 1994, Diversity and Adaptation in Populations of Clustering Ants. Proceedings of the third International Conference on Simulation of Adaptive Behaviour: From Animals to Animates. MIT-Press, 501-508.
- MONMARCHE, N., 2000, Algorithmes de fourmis artificielles : applications à la classification et à l'optimisation. Thèse de Doctorat de l'université de Tours, France, 231.

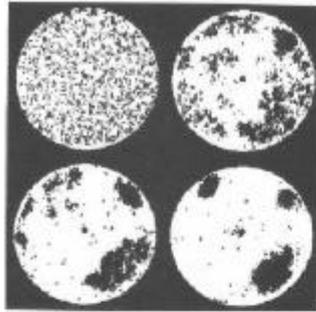


Figure 1. Expérience du tri du couvain chez les *Myrmica ruginodis*. 4 images prises (de gauche à droite) à l'état initial, 2 heures, 6 heures et 26 heures après le début de l'expérience.

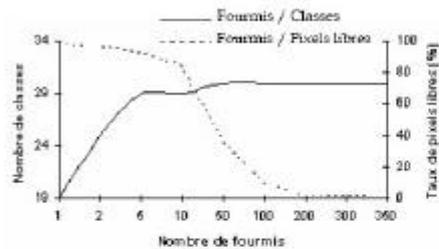


Figure 3. Influence du nombre de fourmis sur le nombre de classes et le nombre de pixels libres

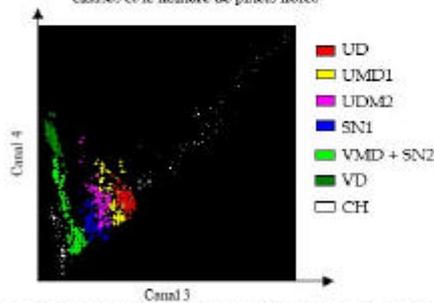


Figure 8. Représentation des 07 classes dans l'espace des caractéristiques

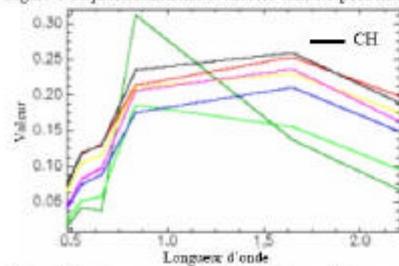


Figure 9. Signatures spectrales des 07 classes obtenues par AntClass

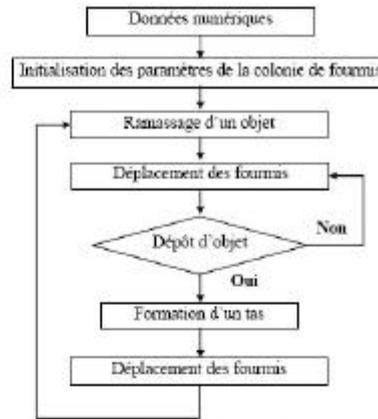


Figure 2. Schéma de principe de l'algorithme AntClass.

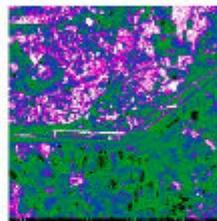


Figure 4. AntClass avec 250 fourmis (0.8% de pixels libres)

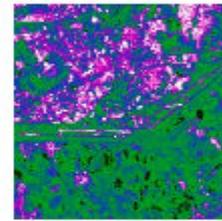


Figure 5. Classification des pixels libres

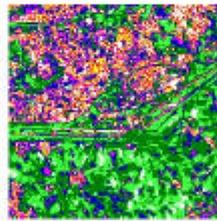


Figure 6. Classification finale pour  $T_{\text{Threshold}} = 0.008$  et  $T_{\text{Minor}} = 0.96$

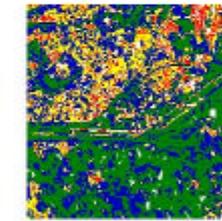


Figure 7. Classification finale pour  $T_{\text{Threshold}} = 0.016$  et  $T_{\text{Minor}} = 0.56$

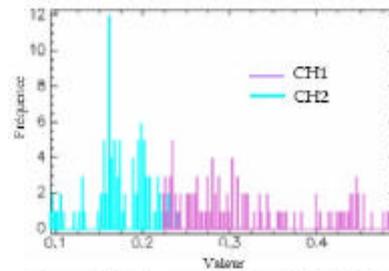


Figure 10. Histogramme de la classe CH dans le canal 4