

TP VisA, Algorithmes de type FCM appliqués à l'imagerie couleur

Tristan Camus et Arnaud Cojez

1 Introduction

Poursuivant nos travaux sur la logique floue, nous allons dans ce TP segmenter des images. Alors que l'algorithme du C-Means est déjà fortement utilisé en logique classique, nous allons utiliser différentes variantes de ce dernier, notamment l'algorithme FCM utilisant de la logique floue. Nous tenterons ainsi de segmenter, à l'aide des algorithmes Fuzzy C-Means, Hard C-Means, Possibilistic C-Means et de l'algorithme de Davé, l'image suivante en 6 classes distinctes :

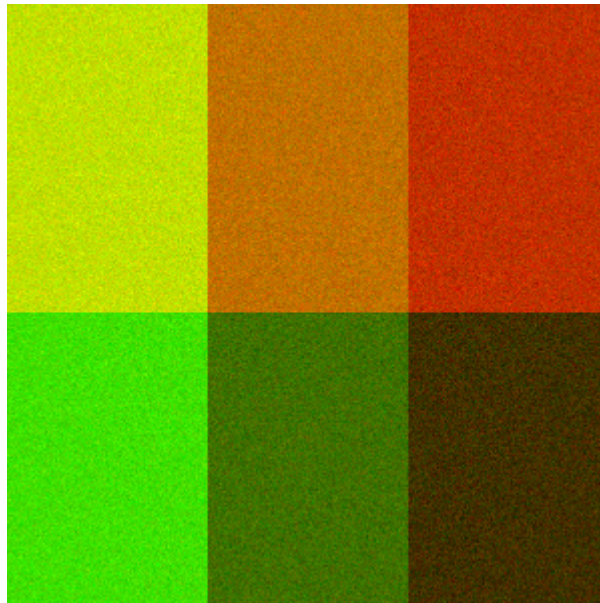


Figure 1: Image à segmentée

2 Algorithme FCM

Fuzzy C-Means est la variante floue la plus simpliste de C-Means. Les résultats de cet algorithme dépendent de la position des centroides calculés aléatoirement au début de l'exécution. Ainsi, les résultats peuvent varier d'une tentative de classification à une autre. Nous montrons ici des exemples de bons résultats à l'aide de FCM.

Ainsi, pour les paramètres suivants :

Nombre de classes	6
Valeur de m	2
Nombre d'itération	100
Valeur de seuil de stabilité	0.1
Randomisation	1

Table 1: Paramètres utilisés pour l'algorithme de FCM

Nous obtenons l'image suivante :

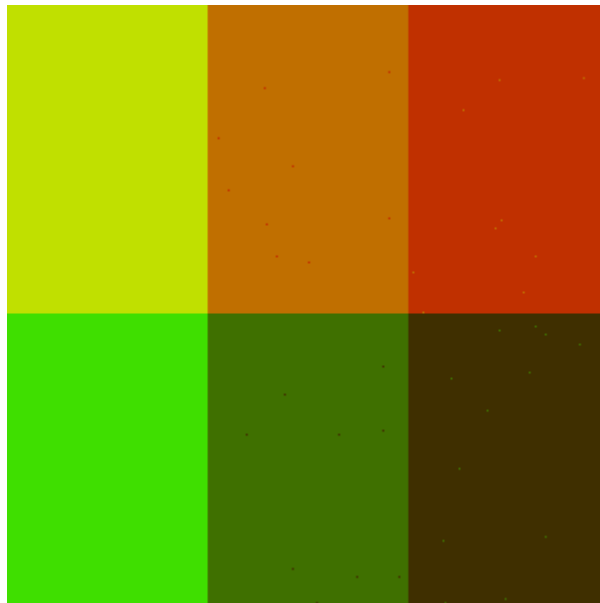


Figure 2: Image segmentée par FCM

Ainsi que la courbe de performance suivante :

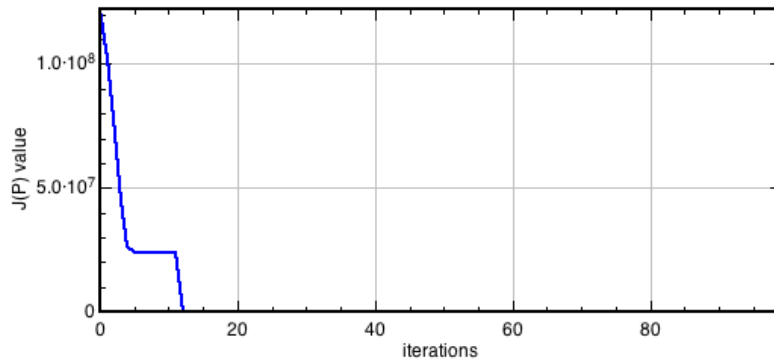


Figure 3: Courbe de performance pour FCM

Le résultat obtenu est assez satisfaisant. On peut distinguer les 6 classes. On note cependant que des points parasites de certaines classes se trouvent dans d'autres.

3 Algorithme HCM

Nous avons ensuite utilisé l'algorithme Hard C-Means afin de segmenter l'image. Les résultats de cet algorithme dépendent de la position des centroides calculés aléatoirement au début de l'exécution. Ainsi, les résultats peuvent varier d'une tentative de classification à une autre. Nous montrons ici des exemples de bons résultats à l'aide de HCM.

Ainsi, pour les paramètres suivants :

Nombre de classes	6
Valeur de m	1
Nombre d'itération	100
Valeur de seuil de stabilité	0.01
Randomisation	1

Table 2: Paramètres utilisés pour l'algorithme de HCM

Nous obtenons l'image suivante :

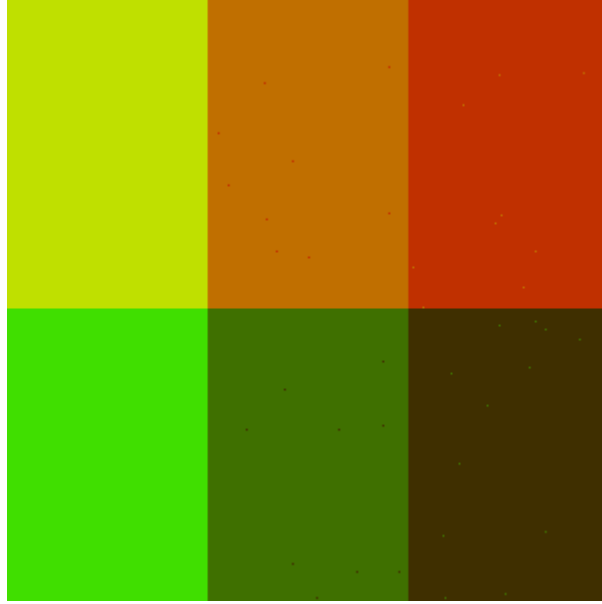


Figure 4: Image segmentée par HCM

Ainsi que la courbe de performance suivante :

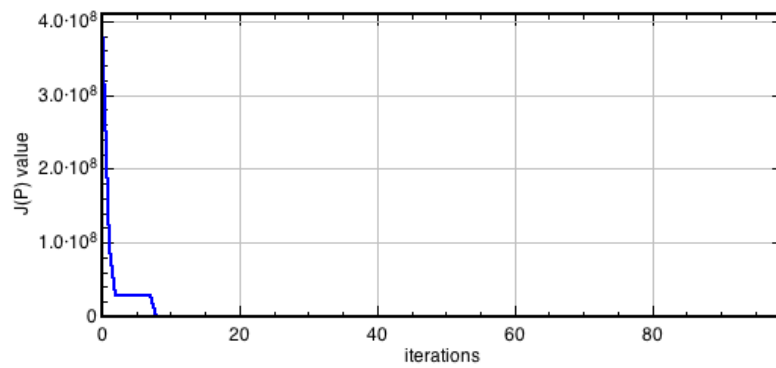


Figure 5: Courbe de performance pour HCM

Le résultat obtenu est assez satisfaisant. On peut distinguer les 6 classes. Cependant, cette image présente plus ou moins le même problème qu'avec FCM et, elle est un exemple de bon résultat. Bien souvent, les deux classes inférieures droites ne forment qu'une seule classe verte foncée. L'algorithme est aussi plus long à s'exécuter.

4 Algorithme PCM

Nous avons ensuite utilisé l'algorithme Possibilistic C-Means afin de segmenter l'image. Les résultats de cet algorithme dépendent de la position des centroides calculés aléatoirement au début de l'exécution. Ainsi, les résultats peuvent varier d'une tentative de classification à une autre. Nous montrons ici des exemples de bons résultats à l'aide de PCM.

Ainsi, pour les paramètres suivants :

Nombre de classes	6
Valeur de m	2
Nombre d'itération	40
Valeur de seuil de stabilité	0.001
Randomisation	1

Table 3: Paramètres utilisés pour l'algorithme de PCM

Nous obtenons l'image suivante :

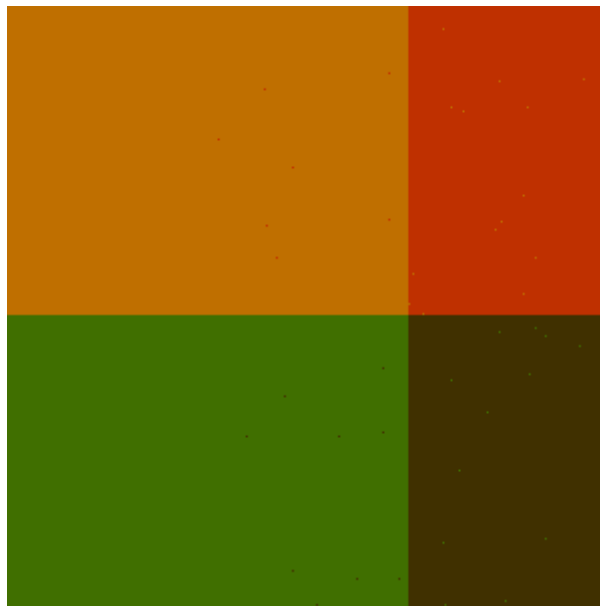


Figure 6: Image segmentée par PCM

Ainsi que la courbe de performance suivante :

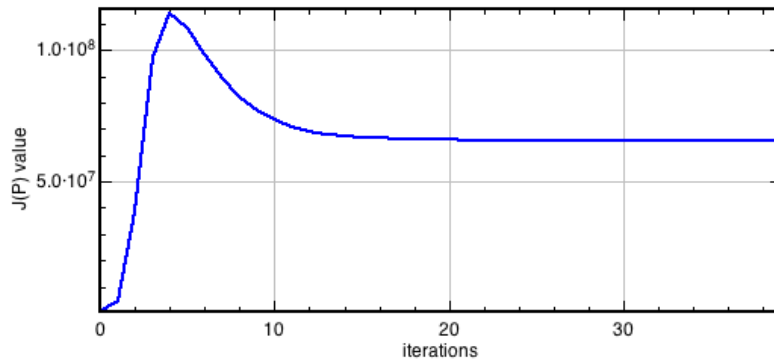


Figure 7: Courbe de performance pour PCM

Le résultat est plutôt mauvais. On ne peut distinguer que 4 classes. les classes de gauches n'en forment que deux, une orange et une verte. C'est encore un fois un exemple. C'est parfois d'autres classes qui ont ce problème.

5 Algorithme de Davé

Nous avons ensuite utilisé l'algorithme de Davé afin de segmenter l'image. Les résultats de cet algorithme dépendent de la position des centroides calculés aléatoirement au début de l'exécution. Ainsi, les résultats peuvent varier d'une tentative de classification à une autre. Nous montrons ici des exemples de bons résultats à l'aide de l'algorithme de Davé.

Ainsi, pour les paramètres suivants :

Nombre de classes	6
Valeur de m	2
Nombre d'itération	100
Valeur de seuil de stabilité	0.000001
Valeur du ratio d'aberration	25
Randomisation	1

Table 4: Paramètres utilisés pour l'algorithme de Davé

Nous obtenons l'image suivante :

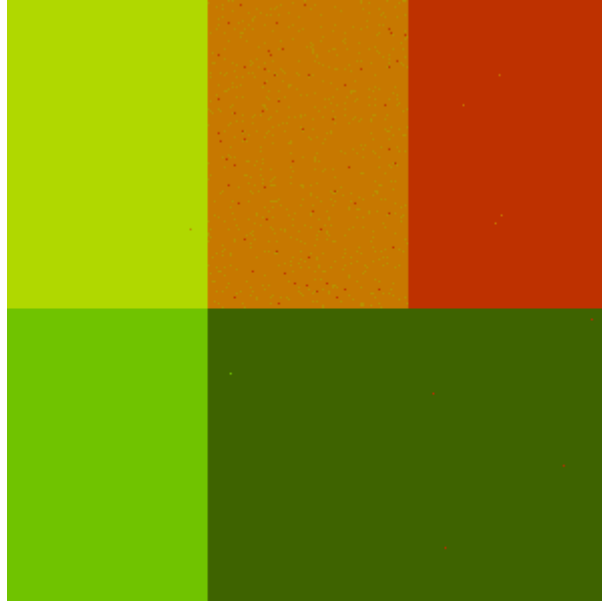


Figure 8: Image segmentée par l'algorithme de Davé

Ainsi que la courbe de performance suivante :

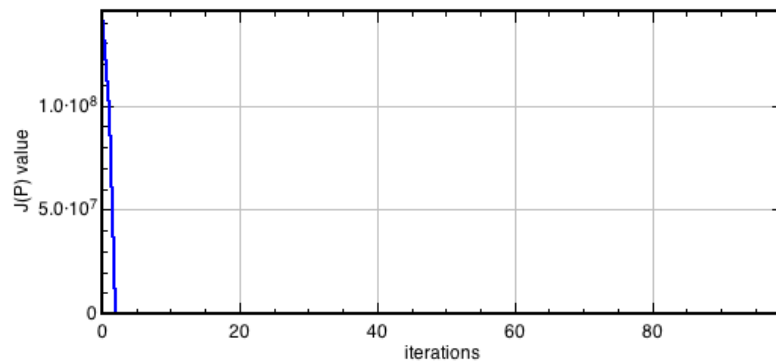


Figure 9: Courbe de performance pour l'algorithme de Davé

Le résultat obtenu est encore une fois assez mauvais. On peut distinguer les 5 classes mais la classe en haut au milieu est fortement parasitée. Encore un fois les résultats changent beaucoup en fonction de la position initiale des centroides. Peut être qu'en trouvant une meilleure valeur pour le ratio d'aberrations, on pourrait obtenir de meilleurs résultats.

6 Conclusion

Nous pouvons conclure que les différentes méthodes vues lors de ces TPs, ne sont pas véritablement des algorithmes de segmentations. En effet, elles calculent des degrés d'appartenance qui ne permettent pas de segmenter directement les images.

Pour segmenter les images dont les degrés d'appartenance aux classes ont été calculés, il faut définir des heuristiques qui vont séparer les pixels.