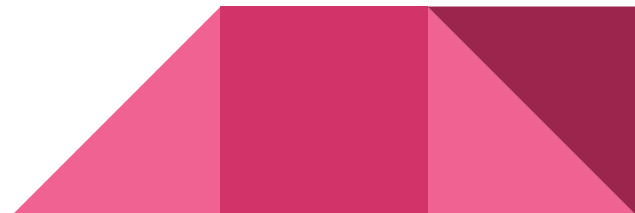


Les Méthodes de segmentation non-supervisée pour les images hyperspectrale

Réalisée par :
Zexing YAO

Sommaire

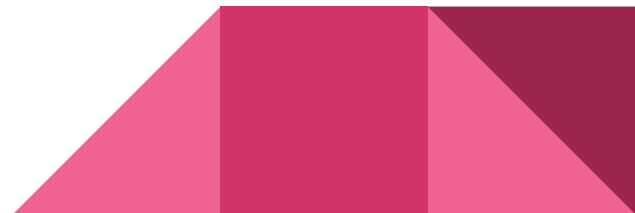
- Introduction
- Méthode de Décomposition: ACP et Ondelette (db1)
- Méthode de Clustering: Expectation-Maximization
- Analyse les resultats



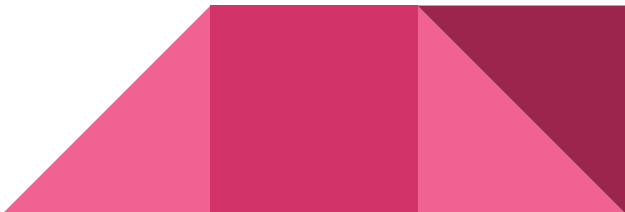
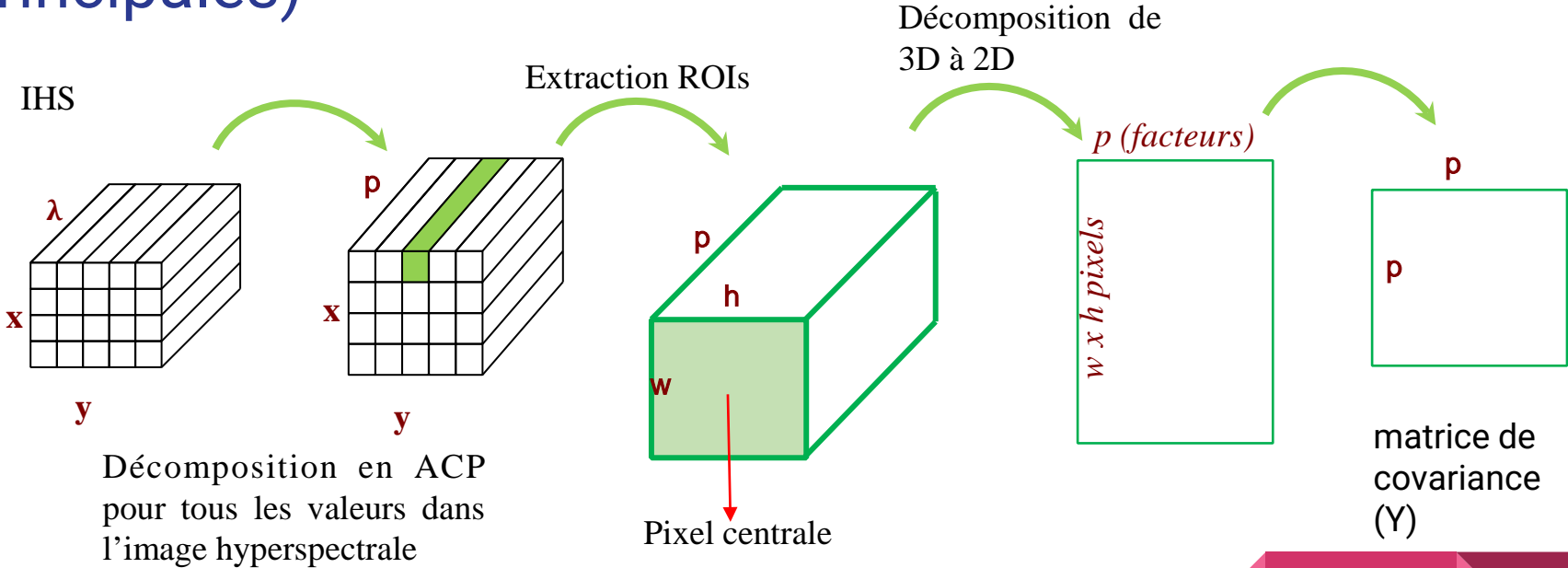
Introduction

L'imagerie hyperspectrale permet de mesurer un grand nombre de longueurs d'ondes, notamment dans le visible et l'infrarouge.

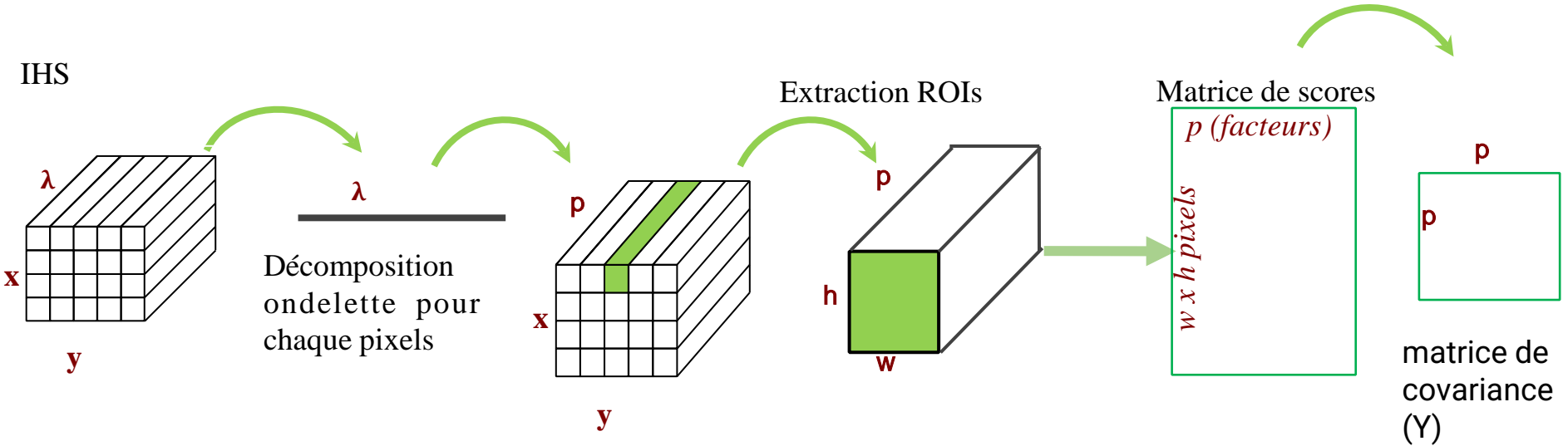
Nous souhaitons explorer une nouvelle méthode pour évaluer et segmenter des images hyperspectrale de végétation avec l'exemple de feuilles de tomate à l'aide de l'espace des matrices de covariance.



Décomposition: ACP(analyse en composantes principales)



Décomposition en 1D: ondelette(db1)



La facteur de composante p n'est pas une valeur aléatoire. Ce correspondant la type d'ondelette et la fois de décomposition.

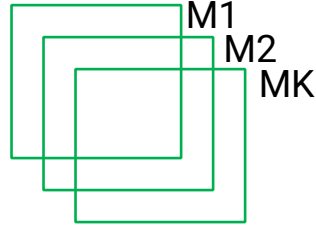
Clustering: Expectation-Maximization

Choisir le nombre de classes K(paramètre à régler)

La famille de matrice de covariance Y (p,p,N)

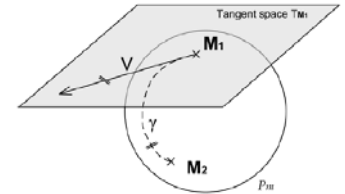


Créer la matrice aléatoire représente les centre masse de chaque class (p,p,K)



Calculer la probabilité entre la matrice covariance et la centre masse avec la distance géodésique

$$p(Y|M, \sigma) = \frac{1}{\zeta(\sigma)} \exp\left[-\frac{d^2(Y,M)}{2\sigma^2}\right]$$



$$\sigma = \operatorname{argmax}_{\sigma_i} \left(\frac{1}{\zeta(\sigma)} * \exp\left(-\frac{d^2(Y|M)}{2\sigma^2}\right) \right)$$

Calculer la maximum de vraisemblance de l'écart type avec la probabilité

Retrouver la centre masse pour chaque class de matrice



Analyser les paramètres

- L'échantillonnage
- La taille de fenêtre(h,w)
- Le nombre de classes K



L'échantillonnage = 10%, 25%, 50%, 100%

image avec 10.0% échantillonnage, $p = 5$, stride = 2, taille = 20 et $K = 4$

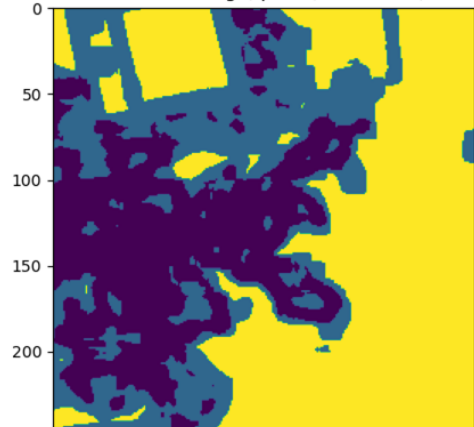


image avec 25.0% échantillonnage, $p = 5$, stride = 2, taille = 20 et $K = 4$

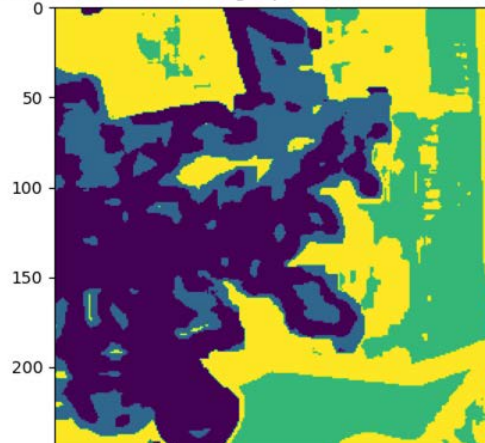


image avec 50.0% échantillonnage, $p = 5$, stride = 2, taille = 20 et $K = 4$

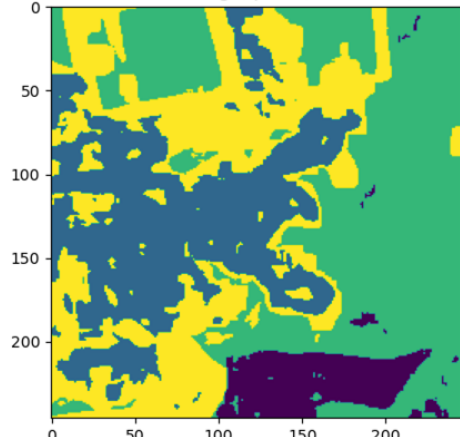
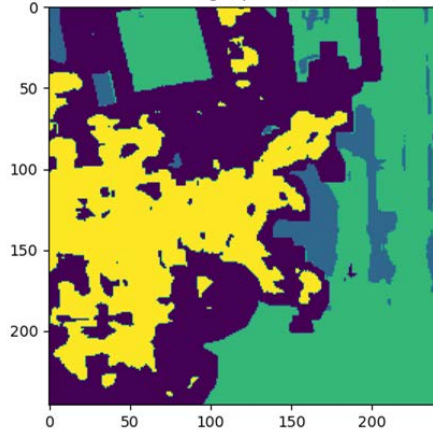


image avec 100% échantillonnage, $p = 5$, stride = 2, taille = 20 et $K = 4$



La taille de fenêtre = 5, 10, 20

image avec 10.0% échantillonnage, $p = 5$, $\text{stride} = 2$, $\text{taille} = 5$ et $K = 4$

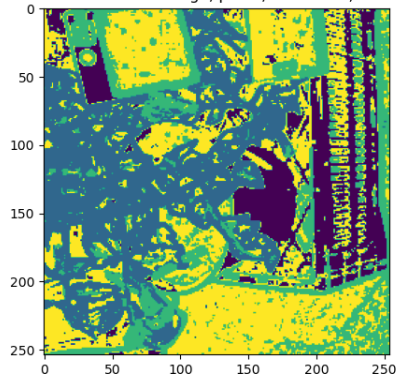


image avec 10.0% échantillonnage, $p = 5$, $\text{stride} = 2$, $\text{taille} = 10$ et $K = 4$

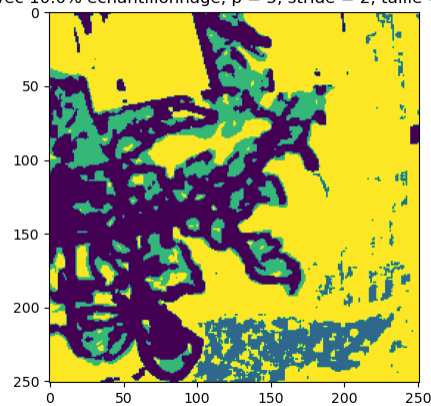
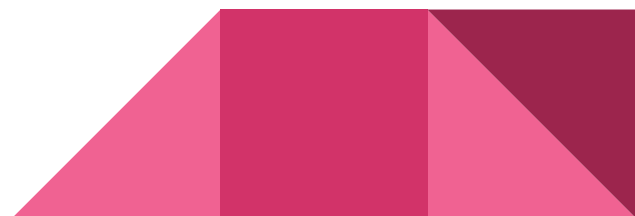
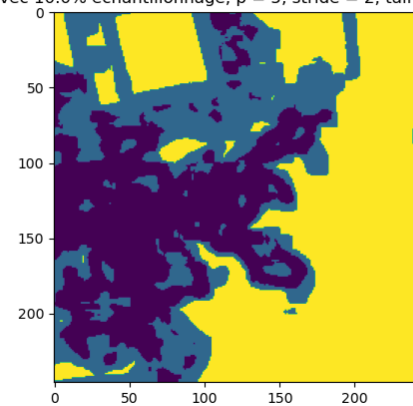


image avec 10.0% échantillonnage, $p = 5$, $\text{stride} = 2$, $\text{taille} = 20$ et $K = 4$



Le nombre de class K

image avec 10.0% échantillonnage, $p = 5$, $\text{stride} = 2$, $\text{taille} = 10$ et $K = 3$

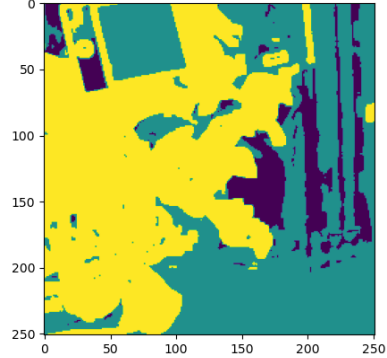


image avec 10.0% échantillonnage, $p = 5$, $\text{stride} = 2$, $\text{taille} = 10$ et $K = 4$

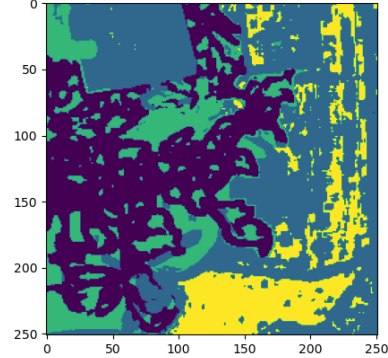


image avec 10.0% échantillonnage, $p = 5$, $\text{stride} = 2$, $\text{taille} = 10$ et $K = 5$

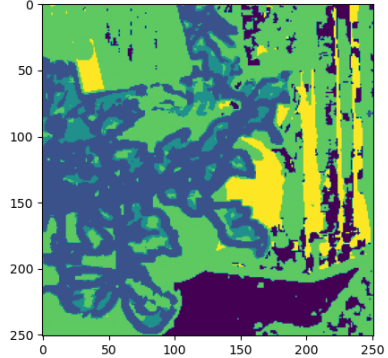
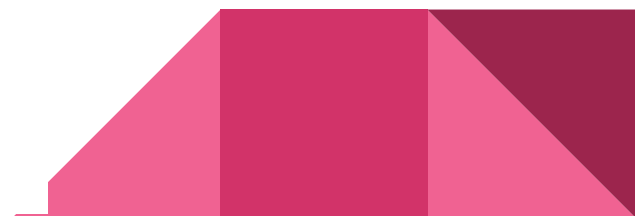
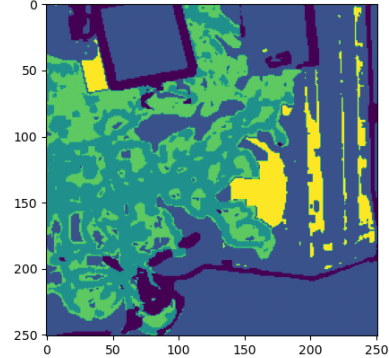



image avec 10.0% échantillonnage, $p = 5$, $\text{stride} = 2$, $\text{taille} = 10$ et $K = 6$



Conclusion

Perspective:

- ❖ Chercher le meilleur nombre de classes K automatique
 - ❖ Accélérer les programmations
 - ❖ Comparer avec les autres méthodes de segmentation.(Kmeans,KNN)
 - ❖ Utiliser cette méthode en apprentissage profond
- 

Merci de votre attention !

