

Développement d'un utilitaire de sélection de particules observées au microscope électronique

Pick_EM

FAUX - HÉRICÉ - PAYSAN-LAFOSSE - SANSEN

Master 1 Bioinformatique
Projet de programmation sous la direction de Jean-Christophe TAVEAU



Mai 2012

1 Introduction

- Contexte
- Objectifs

2 Analyse

- Besoins Fonctionnels
- Besoins non Fonctionnels

3 Conception - Réalisation

- Interface Graphique (GUI)
- Algorithmes
- Modularité

4 Conclusion

Introduction

CBMN

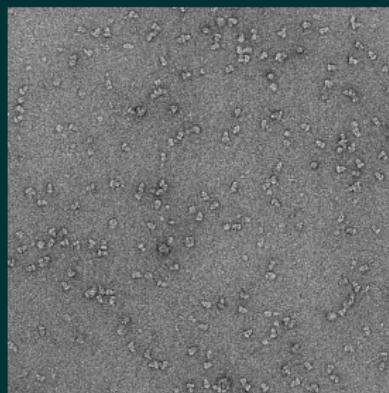
Laboratoire de Chimie et Biologie des Membranes et Nanoobjets

ACMPC

Équipe Architectures des Complexes Membranaires et Processus Cellulaires

Contexte

- Micrographies de structures protéiques issue de MET
- Utilisation du logiciel ImageJ
- Sélection manuelle fastidieuse
 - Chronophage
 - Accapare un membre de l'équipe
 - Répétitive



Exemple de micrographie

Objectifs

Création d'une interface

- Facile d'utilisation
- Claire et succincte
- Récupération des paramètres utilisateurs

Implémentation d'algorithmes

- Automatisation du traitement et de la sélection
- Récupération des coordonnées
- Images résultantes

Besoins Fonctionnels

Interface

- Choix entre plusieurs algorithmes
- Diffère entre chaque algorithme

Algorithmes

- Piquage automatique
- Résultats : tableau de coordonnées (x, y, z) et images résultantes

Besoins non Fonctionnels

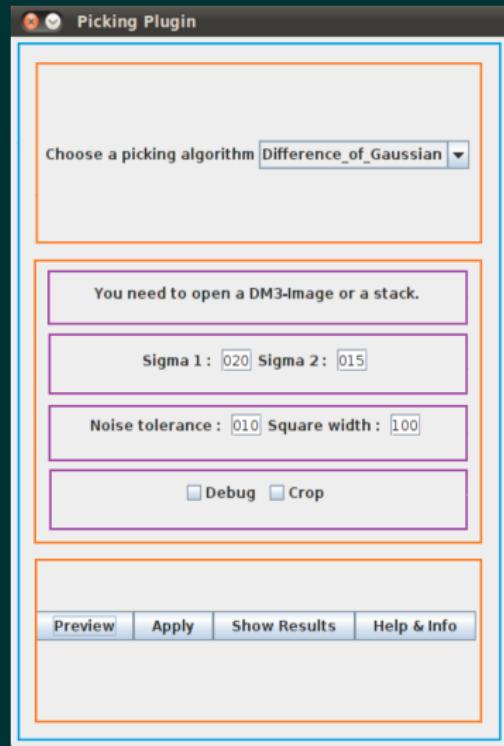
Interface

- Implémentation en Java (AWT ou **Swing**)
- Modularité

Algorithmes

- Implémentation en Java (tests avec l'outil Macro d'ImageJ)
- Traitement rapide (grands jeux de données)
- Minimiser les étapes intermédiaires

Interface Graphique (GUI)

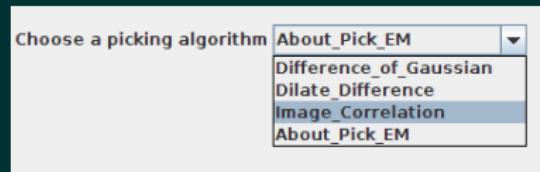


Organisation générale de l'interface

GUI - Récupération des paramètres utilisateurs

Choix de l'algorithme

- JComboBox



Paramètres propres aux algorithmes

- JTextField
- JCheckBox

The screenshot shows a user interface with several input fields:

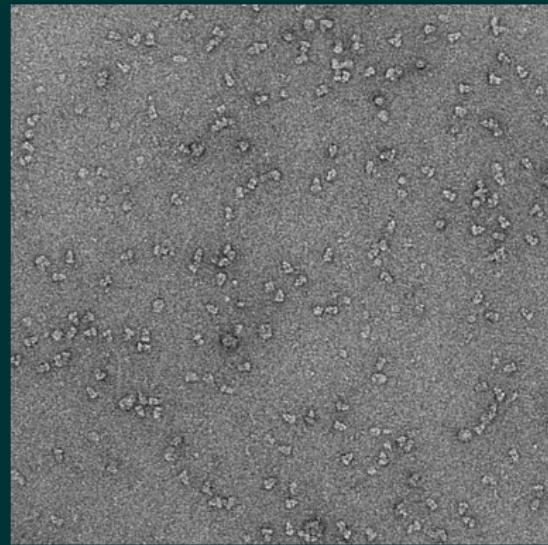
- "You need to open a DM3-Image or a stack."
- "Sigma 1: 020 Sigma 2: 015"
- "Noise tolerance : 010 Square width : 100"
- " Debug Crop"

Les Algorithmes implémentés

3 algorithmes implémentés :

- Différence de Gaussiennes
- Extraction de contours (Différence de dilatation)
- Corrélation d'images

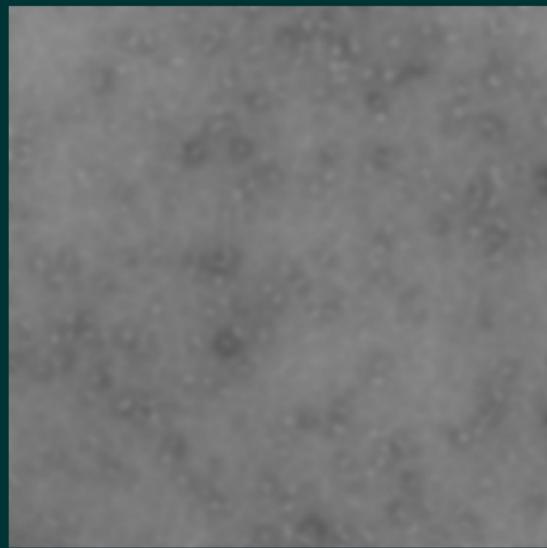
Algorithmes - Démonstration avec la Différence de Gaussiennes



Micrographie de protéines transmembranaires

- Application de filtres gaussiens
- Soustraction des images filtrées
- Récupération des maxima
- Extrapolation des points sélectionnés sur l'image de bases

Algorithmes - Les filtres

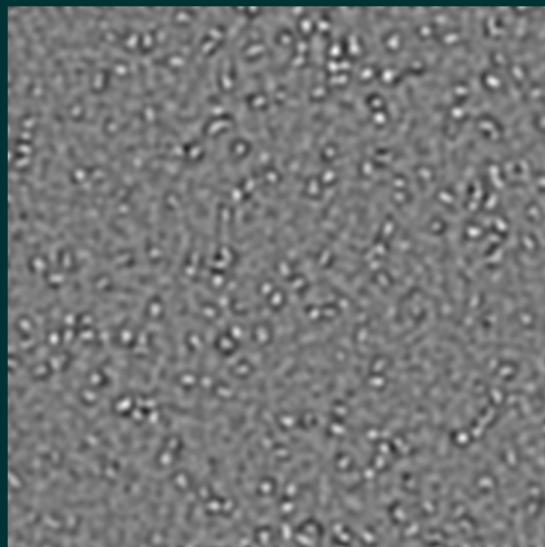


Filtre Gaussien (Radius 15)

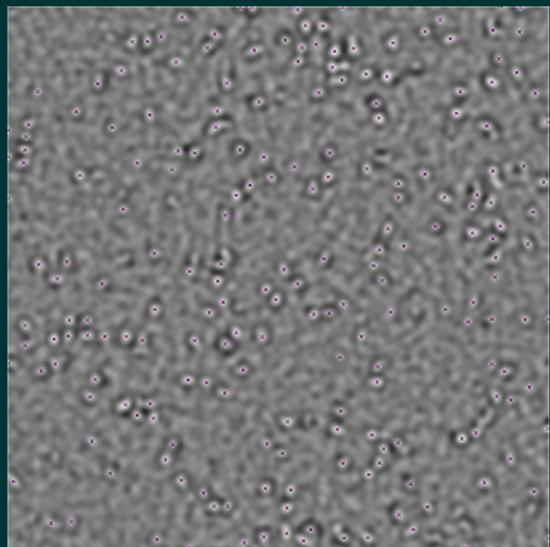


Filtre Gaussien (Radius 20)

Algorithmes - Résultats intermédiaires

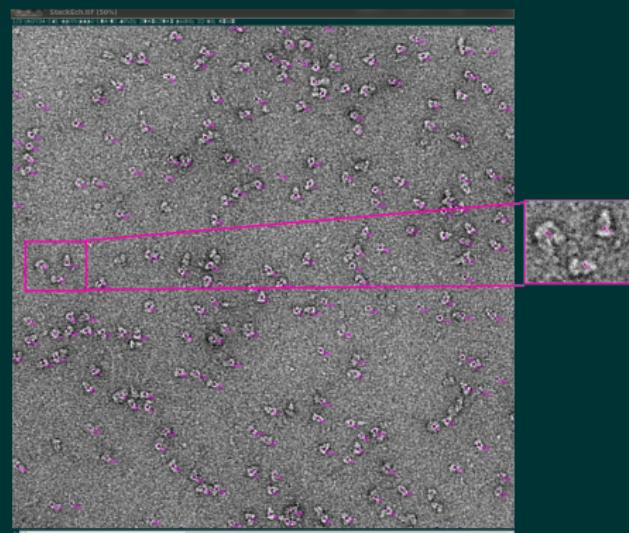


Résultat de la soustraction

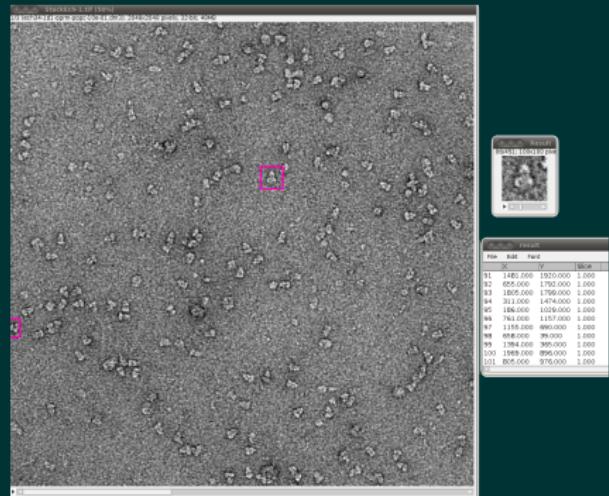


Résultat du maxima

Algorithmes - Résultats



Résultat du piquage



Images individuelles et tableau de coordonnées

Modularité - Diagramme de classes

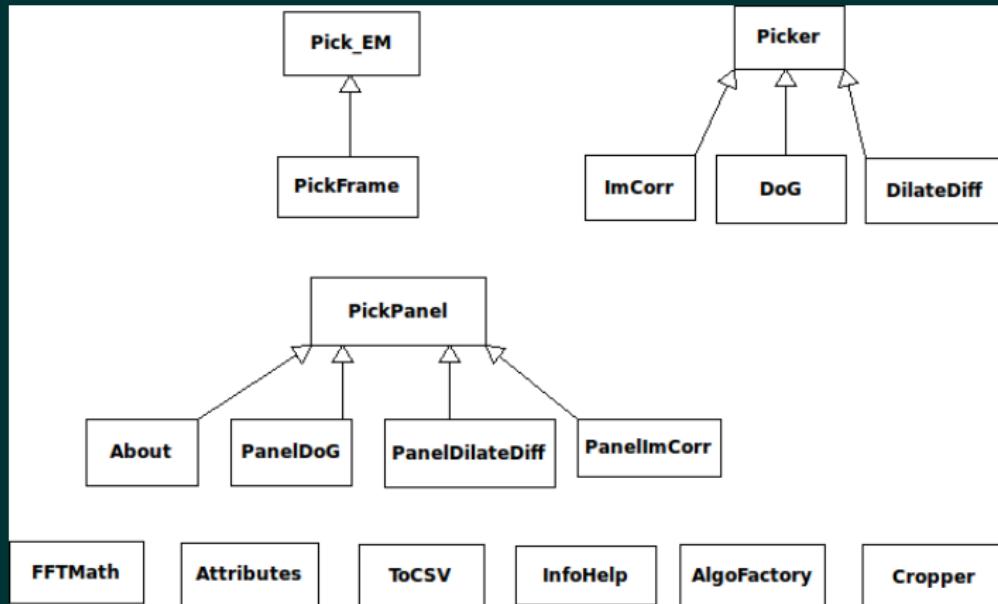


Diagramme des classes

Modularité - Patrons de conception

Singleton

- Sert à contrôler le nombre d'instances d'une classe présent à un moment donné,
- Restreindre l'instanciation d'une classe à un seul objet

Factory

Classe qui n'a pour rôle que de construire des objets, elle est seule responsable de la création / distribution de l'objet

Difficultés rencontrées

I'API ImageJ

Distinction :

- ImagePlus
- ImageProcessor
- ImageStack

GUI

- Gestion des Panels
- Organisation et tailles de fenêtres

Améliorations

Améliorer :

- Améliorer le mode debug
- Continuer le travail pour l'utilisation en macro

Ajouter :

- Ajouter automatiquement un algorithme
- Afficher les sélections pour chaque image d'un stack

Conclusion

Le projet

- Interface facile d'utilisation
- Piquage efficace
- Récupération des images individuelles et des coordonnées fonctionnelle

L'équipe

- Première expérience de travail en équipe sur un gros projet concluante
- Aperçu de notre futur métier

Remerciements

Merci !!! ;-)

Merci beaucoup pour votre attention !