Tracking Code Documentation

detection4.m

Run this file to save a list of detection coordinates per frame. (Detection is done using the LoG Filter and Matlab Morphology)

Make sure to specify the correct base directory, and have a file inside the base directory with your .tiff video frame screenshots

if particles are very close together, consider commenting out lines 61-70 and 86-100
if a boundary condition is NOT imposed, comment out lines 119-150

Input parameters to adjust:

parameter	description	optimal range
size	Controls the width of the LoG filter, that we convolute with the image in the particle detection step. The wider the LoG filter, the more particles are detected.	3-4
sigmah	Controls the breadth of the peak of the LoG filter, that we convolute with the image in the particle detection step. The broader the peak, the less affected the the filter is to noise, resulting in less particles being detected.	0.43-0.45

Output:
X: cell array of x coordinates of detections/ frame
Y: cell array of corresponding y coordinates of detections/frame

tracking4.m

Run this file after detection4.m, to get a set of trajectories for the particles detected. Object tracking is done using the nearest neighbor method with collision fix. Best used when particles are relatively stationary and have a narrow velocity distribution.

Input parameters to adjust:

parameter	description	optimal range
MUNKRES_REJECT	When using the Hungarian one-to-one matching in the particle matching step, we want to disregard matches that are too far apart. If matches are father apart than Thresh_1, we reject them in our algorithm. High values of Thresh_1 are optimal in cases where there are large true particle jumps between frames. Thresh_1 essentially corresponds to the farthest true jump a particle makes.	5-35
COLLISION_REJECT	In the particle matching step, if a particle has not been matched up for Thresh_2 number of frames, we assume that the particle is no longer in the image and we remove it from our tracks. If the data is noisier and particles are not detected reliably, we can have a higher value of Thresh_2.	4-8
REMOVE_FRAME	If there are collisions in the particle trajectories, we allow for two particles to be matched to the same detection in the next frame if the matches are less than Thresh_3 distance away from each other. Thresh_3 should only be set higher if we anticipate collisions, or have noisy data. Thresh_3 essentially corresponds to the farthest jump a particle makes into a collision. Furthermore, Thresh_3<Thresh_2.	0-15

Output:

final_tracks: Trajectory of each particle detected, X and Y coordinates sorted by particle in a cell array. Useful for direct input into classificationTrackMateDZ.m
Q_loc_estimateX: X Coordinates of particle Trajectories
Q_loc_estimateY: Y Coordinates of particle Trajectories

bayes4.m

Run this file after detection4.m, to get a set of trajectories for the particles detected. Object tracking is done using a probability bayesian method with collision fix. Best used when particles have a broad velocity distribution.

Input parameters to adjust:

parameter	description	optimal range
MUNKRES_REJECT	When using the Hungarian one-to-one matching in the particle matching step, we want to disregard matches that are too far apart. If matches are father apart than Thresh_1, we reject them in our algorithm. High values of Thresh_1 are optimal in cases where there are large true particle jumps between frames. Thresh_1 essentially corresponds to the farthest true jump a particle makes.	5-35
COLLISION_REJECT	In the particle matching step, if a particle has not been matched up for Thresh_2 number of frames, we assume that the particle is no longer in the image and we remove it from our tracks. If the data is noisier and particles are not detected reliably, we can have a higher value of Thresh_2.	4-8
REMOVE_FRAME	If there are collisions in the particle trajectories, we allow for two particles to be matched to the same detection in the next frame if the matches are less than Thresh_3 distance away from each other. Thresh_3 should only be set higher if we anticipate collisions, or have noisy data. Thresh_3 essentially corresponds to the farthest jump a particle makes into a collision. Furthermore, Thresh_3<Thresh_2.	0-15

Output:

final_tracks: Trajectory of each particle detected, X and Y coordinates sorted by particle in a cell array. Useful for direct input into classificationTrackMateDZ.m
Q_loc_estimateX: X Coordinates of particle Trajectories
Q_loc_estimateY: Y Coordinates of particle Trajectories