ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

BACTERIAL MECHANOTAXIS: TRACKING GROUP BEHAVIORS

ME-480 MECHANOBIOLOGY

TAUGHT BY

SELMAN SAKAR, ALEXANDRE PERSAT

ASSOCIATE PROFESSOR
LABORATORY FOR APPLIED MECHANICAL DESIGN (LAMD)

Léo Assier De Pompignan, Baptiste Bühler, Kelly Touzeau, Pierre Maillard, Naïm Sabaghi

Fall 2023

Table of contents

1 Introduction

Throughout this project we will analyze microscopy data of motile cells in order to evaluate single and global motion. From these observations, we will hypothesize mechanosensing's function in group behavior.
The studied cells are *Pseudomonas aeruginosa*. We study 3 different types of cells :

- Wild type
- pilG mutants
- pilH mutants

For each type of cells, 2 files are explored : *dense* or *dilute*.
To achieve the goal of the project, we need to segment the files and then track each cell to analyze its motility. Please find details on our methodology in the next section.

2 Methodology

2.1 Segmentation

The segmentation consists of separating the bacteria from the background. To do so, we tried a couple of methods like a basic threshold, with or without image preprocessing, or machine learning algorithm. Find some details of each part below.

Threshold

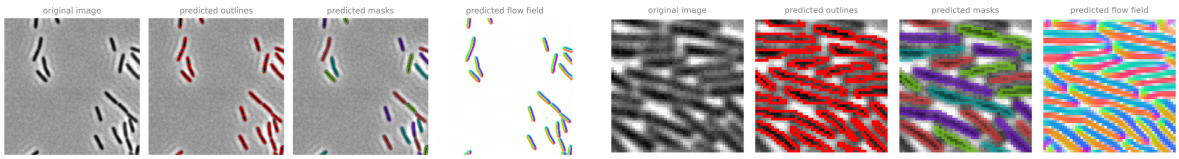
With a simple threshold, the result were not so bad for a small part of the images but when we took the whole size of the frames, there were some noisy parts. Furthermore, when the cells are very close, or when they collide, the threshold can not distinguish two different cells and thus, merge them. This plays a role during the tracking. To avoid this problem, we tried to add after the threshold a watershed segmentation as explained below.

Watershed

The watershed segmentation result consist of outlines. It can thus, potentially separate the merge cells. While the segmentation's global outcome was improve, the tracking performance remained unsatisfactory due to the method's inconsistency across frames.

Omnipose

As simple computing method were not convincing for us, we took a look at machine learning (ML) algorithms. **Omnipose** (Cutler (2022)) is the one that we retain for this project. **Omnipose**, enhance of **Cellpose**, is an open source software and python library. This deep neural network algorithm can segment cells of different size and shape and as a pre-trained model for the type of bacteria being studied called *bact_phase_omni*. You can find examples of the segmentation using **Omnipose** in our visualization's notebook (*ME480_P1_Visualization.ipynb*). This method allows the masks of each image to be output directly. These masks are then used directly for tracking. We chose to segment 200 slices per file with dimension 500x500. This choice is due to computation time.



(a) Dilute **(b)** Dense

Figure 1: Example of segmentation outputs with **Omnipose**.

2.2 Tracking

Laptrack

Laptrack (Fukai (2021), Fukai and Kawaguchi (2022)) is a is free and open source software and python library that provides a robust particle tracking algorithm using the Linear Assignment Problem. It is highly inspired on *TrackMate* which we first tried to use on Fiji. We tried two different approaches: The first was based on superimposing the cells to reconstruct the tracks. It was not very efficient for dense files, as many cells were superimposed between each image. The second method is based on calculating centroids of each cell before reconstructing tracks using a cost function. Tracking was then ameliorated as centroids were far enough apart to allow the algorithm to reconstruct the tracks. This method also gives us direct access to cell positions. **Laptrack** let us choose to two cost function parameters. We chose them qualitatively by observing the tracking on a part of the image using the Napari viewer. Finally, we chose a very small $splitting\_cost\_cutoff$ with a standard $track\_cost\_cutoff$ to allow the separation of tracks that would merge when two cells collide. Due to the exceedingly low splitting cost, we may lose certain tracks that split even in the absence of collisions as observed in Figure 3a.
The Figure 3 shows a visualization of tracking that you can also find in our visualization's notebook (*ME48_P1_Visualization.ipynb*).

Listing 1: Creation of LapTrack objet

```
lt = LapTrack(track_cost_cutoff=10**2, splitting_cost_cutoff=5**2)
```

Tracks cleaning

To try and improve the analysis, we have tried to remove any spurious tracks resulting either from a separation (if the segmentation has merged 2 cells together) or from a false cell (if the segmentation has considered the space between 2 cells to be a cell). To achieve this, after getting ride of absurd tracks (two location on the same frame or more than 10 pixels jump between two frames) we simply chose a minimum number of frames on which the tracks must appear to be consider "real" tracks.
On the Figure 2, we can see that this minimum of frame can be set to 75 to keep sufficient information on tracking.

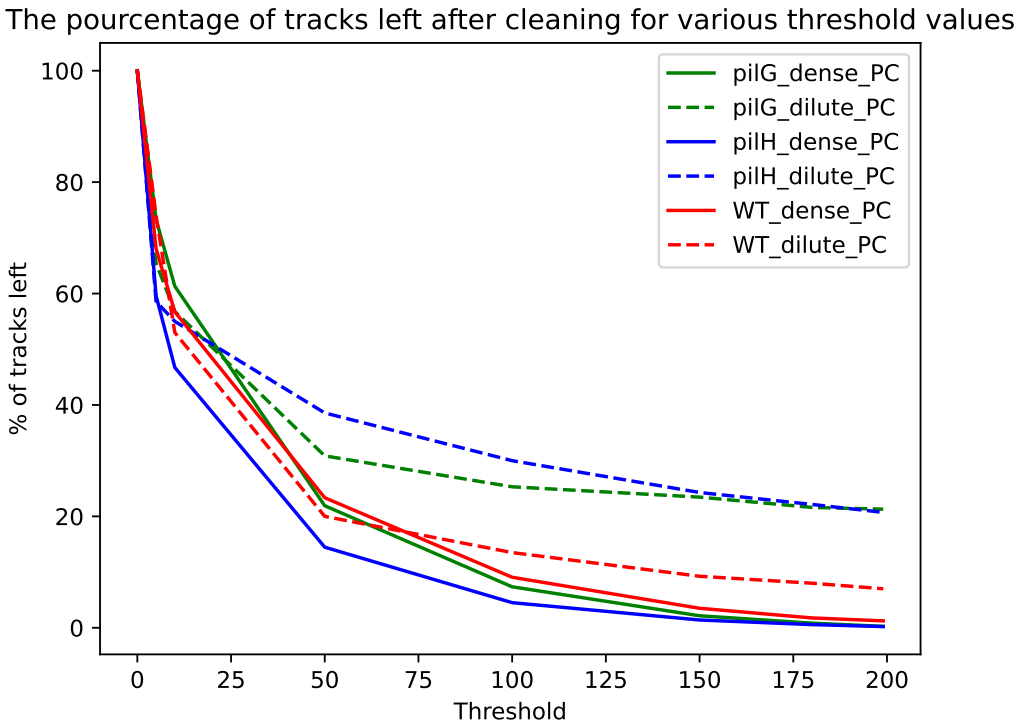
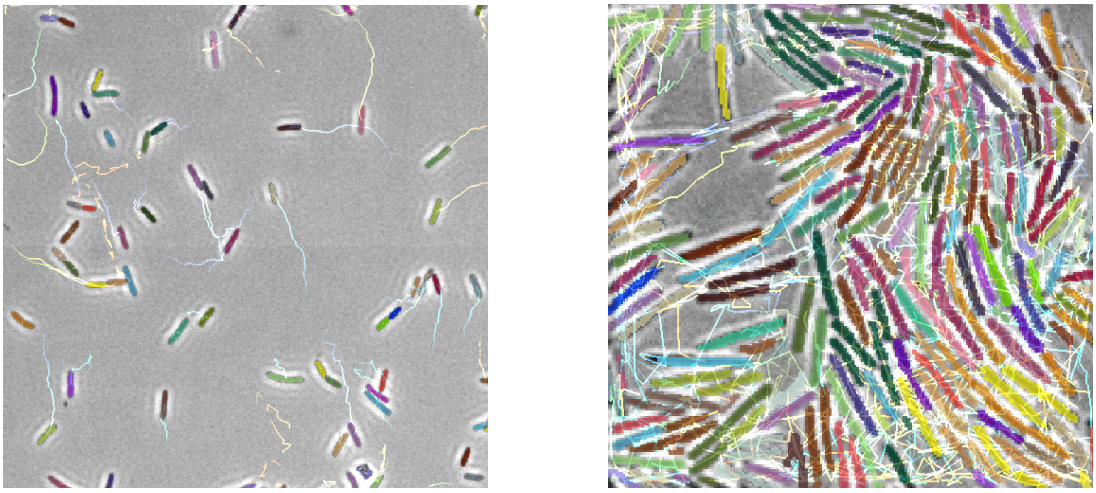


Figure 2: Percentage of tracks left after cleaning for various threshold values

3 Analysis steps and results

3.1 Segmentation and Tracking quality



(a) Dilute **(b)** Dense

Figure 3: Example of segmentation and then tracking with **Napari** before tracks cleaning.

Qualitatively, we can see on the Figure 3a that the tracking is quite clear and good. We can also notice that some tracks are discontinuous. These tracks will be deleted with the cleaning algorithm. On the Figure 3b the tracking is difficult to observe. This is due to the density of the cells. Furthermore we can see that the behavior of the cells is more messy on the dense files than on the dilute ones. This is shown ins our computation's notebook (*ME480_P1_Computation.ipynb*).
On Figures 4 and 5 we can see that after a threshold value of 75 the behaviour is quite constant for the majority of the files. Thus, we conclude that we were measuring the right value for speed and distance. As you can see the files *pilG_dense_PC* and *pilH_dense_PC* do not show a constant behavior after a threshold of 75. This is possibly due to the density which causes problems during the tracking. The tracking is not really stable for these files, you can see this comparing the animations *anim_illustration_nojump* that show the tracking of a dilute file *anim_illustration_jump*, tracking from a dense file. (The two animation are the tracks after a threshold of 195 for good illustration purpose) We can clearly see that some tracks in the dense file "jumps" between two existing tracks. By bypassing the cleaning processe, these tracks could be the cause of the high speed and distance observed in the *pilG_dense_PC* and *pilH_dense_PC*.

3.2 Cells type behaviors

To compare the ability of different cell types to move, we calculated the average speed and total displacement for all files. To compare group behavior, nematic order was also calculated. For reasons of calculation time, the analysis was performed on a 500x500 pixels square centered on the original image over 200 frames.

Cells type	Speed [μm/s]	Distance travelled [μm]
Dense files		
Wild type	0.206 ± 0.104	30.82 ± 14.28
pilG mutants	0.352 ± 0.140	50.99 ± 17.90
pilH mutants	0.306 ± 0.123	44.22 ± 16.01
Dilute files		
Wild type	0.091 ± 0.065	47.13 ± 31.87
pilG mutants	0.029 ± 0.016	16.34 ± 7.56
pilH mutants	0.043 ± 0.024	24.31 ± 13.89

Table 1: Mean speed and mean distance travelled for each cells type over 200 frames. A threshold of 75 frames have been previously applied to clean the tracks.

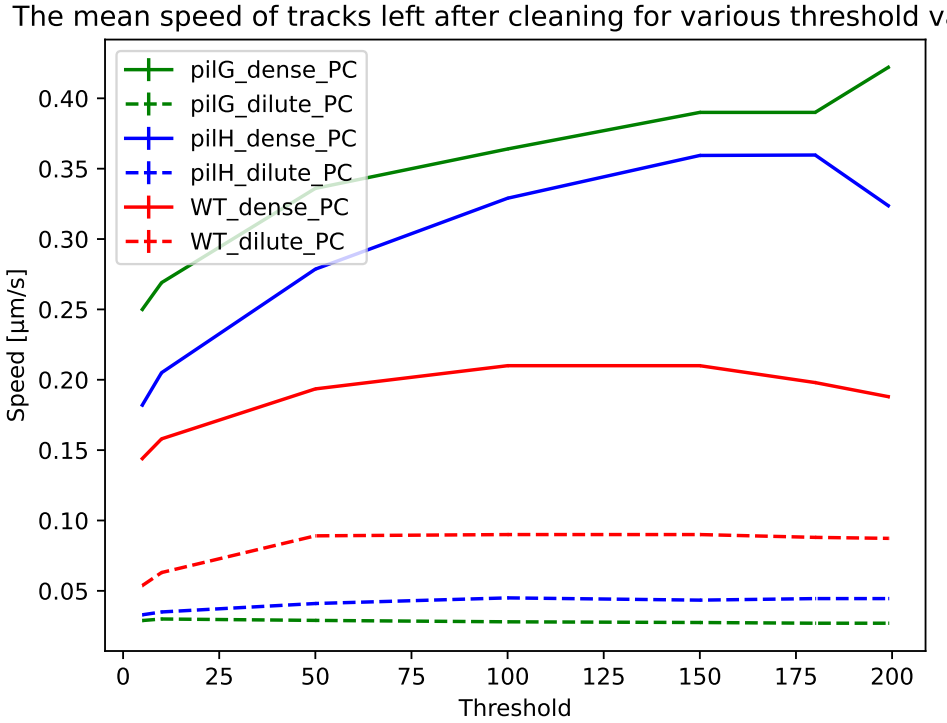


Figure 4: Mean speed of all cells type with respect to the threshold applied when cleaning the tracks. The standard deviations are not showed for clarity purpose but can be found on *ME480_P1_Computation.ipynb*

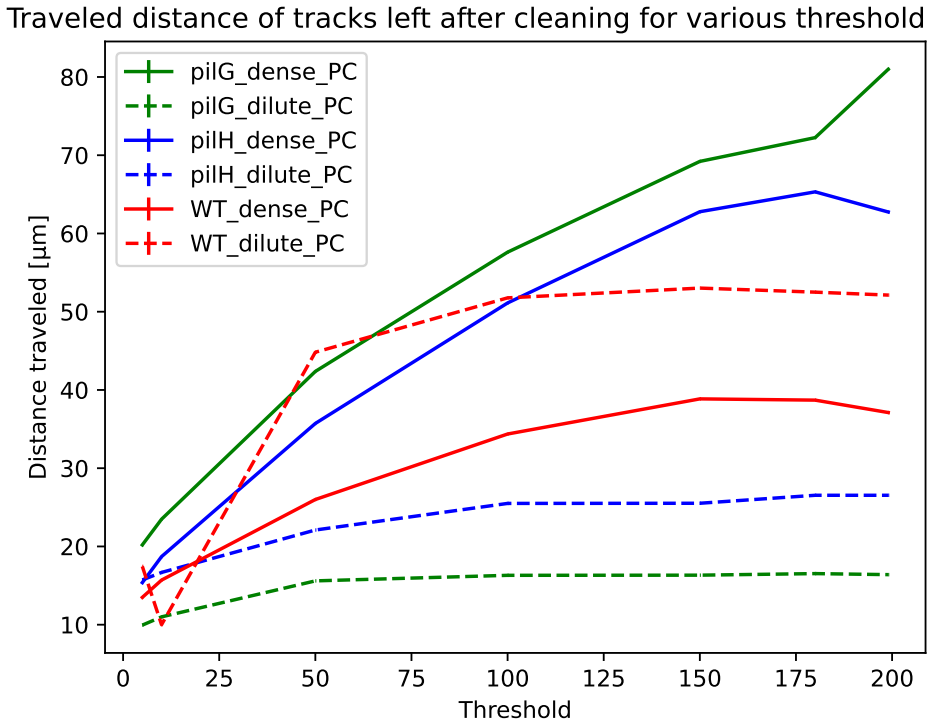


Figure 5: Mean traveled distance of all cells type with respect to the threshold applied when cleaning the tracks. The standard deviations are not showed for clarity purpose but can be found on *ME480_P1_Computation.ipynb*

3.3 Nematic Order

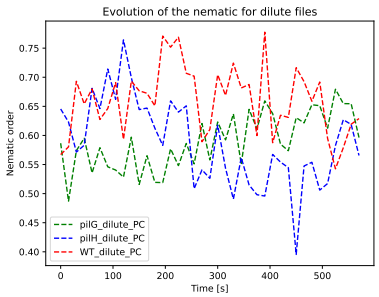
As explained in the paper on large-scale orientational order in bacterial colonies during inward growth (Basaran (2021)), the nematic order is a relative measure of the orientation of the cell. It compares the orientation of the cell with respect to the x-axis and its polar angle with respect to the colony center.
When a cell is oriented towards the center of its colony its nematic order will be equal to 1. On the other hand, if the cell is oriented tangentially to the center of the colony, it will have a nematic order equal to -1.
We computed the global nematic of the provided files below:

Cells type	Nematic order
Dense files	
Wild type	0.652 ± 0.085
pilG mutants	0.455 ± 0.122
pilH mutants	0.619 ± 0.098
Dilute files	
Wild type	0.660 ± 0.058
pilG mutants	0.590 ± 0.047
pilH mutants	0.587 ± 0.072

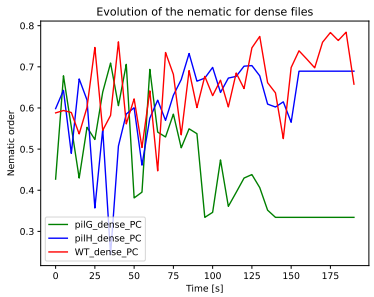
Table 2: Nematic order for each cells type over 200 frames. A threshold of 75 frames have been previously applied to clean the tracks.

From these results, we can assume that the majority of the cells are pointed towards the center of the colony. However, the nematic order between the two types of cells are too close to be used as a determining distinction factor between pilG and pilH cells.

The nematic order in time stays has some fluctuation around its mean value for all the types of cells. The fluctuations seems to be faster in the dense files for all types of cells.
One think to note is that although it fluctuates, the nematic order stays bounded between 0.4 and 0.8 indicating that most of the cells are pointing towards the center of the colony at all time. Very few of them show a behaviour of moving tangentially to the center.



(a) Dilute files



(b) Dense files

Figure 6: Nematic order with respect to time for all cell types. This is for tracks left after a cleaning with a threshold of 75 frames.

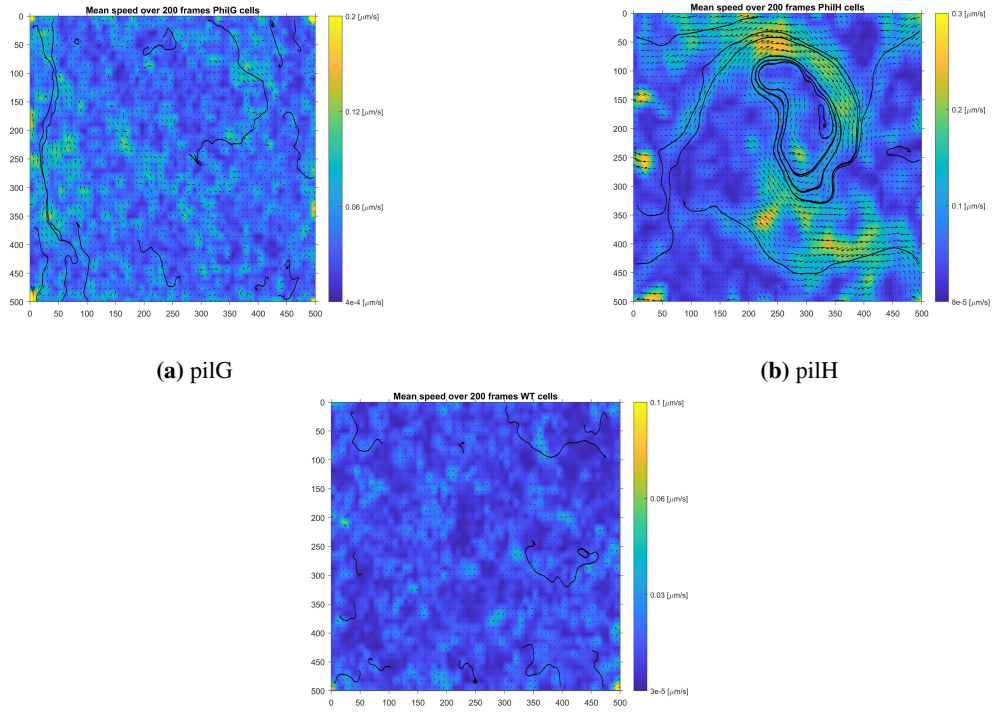
We had previously assumed that the WT would have behaved in a more randomised manner than the other two, implying a more vast variety of nematic order in time, and potentially negative nematic orders. Also, we expected the nematic order behavior results to be significantly different between all of the types of cells. Whereas here, although a clear definition of the behavior is made difficult by the extensive fluttering of the value, we see on Figure 6, that for all types the boundary values of the nematic order are similar.
This might be due to different aspects of our code:

1. Our method of tracking, and especially of cleaning our data by deleting non conform tracks, might have induced error in our nematic order.

2. Our method for computing the nematic order might being also be the cause of these errors. We are currently determining the orientation of the cell by computing the angle of its path and assuming that it is a similar value. We also assumed the center of the colony to be at the center of the image which might not precise enough and would randomise the nematic order result.
   We could have potentially obtained smoother results by computing mean nematic orders for a certain number of frames and iterate the process. Another way to improve our results, would be to compute the nematic order for only a cluster of cells of high density for each frame rather than the nematic order of the whole frame. From this we might expect some typical behaviour from our cells, while still lessening the noise produced by lone cells throughout the image.

3.4 PIV analysis

To conclude on the cells behaviour we tried a PIV analysis using the Matlab application *PIVlab* (Thielicke and Sonntag (2021)), by comparing the average speed field over 200 frames in the dense files. Results are shown on the Figure 7. We can observe that pilH show a "fluid like" behavior. What we want to express here is that, since the pilH cells are *collision-blind*, they are more likely to move in groups than individually. This is shown in the files by the streamlines converging in the formation of a small vortex. To characterize the behavior of the two additional cell types, pilG and WT, in qualitative terms, it can be stated that both exhibit no "fluid-like behavior" compared to pilH. Instead, they demonstrate more random individual displacement.



(a) pilG

(b) pilH

(c) WT

Figure 7: PIV analysis of each type of bacteria. The vector field represents the mean speed, combined with a colormap illustrating the norm of the speed field. The black tracks correspond to the streamlines. (a) pilG show weak "fluid like behavior". (b) pilH show "fluid like behavior". (c) WT show no "fluid like behavior".

4 Discussion

4.1 Interpretation of results

As the paper on mechanotaxis (Kühn (2021)) said : "WT cells often reversed their twitching direction after a side collision. ΔpilGcpdA reversed almost always after impact, whereas ΔpilH almost never did", we want to check if the results are coherent with this sentence. We can separate the dilute and dense files to interpret the results.
The limited interactions among dilute bacterial populations hinder comprehensive insights into their group behavior. However, discernible distinctions in activity, particularly in terms of speed and spatial displacement, are

evident among bacterial variants. This observation underscores the involvement of the Chp response regulators PilG and PilH, which govern the polarization of the extension motor PilB, in influencing bacterial motility. PilG stimulates polarization favoring forward migration, while PilH inhibits polarization, inducing reversal. Subcellular segregation of PilG and PilH efficiently orchestrates their antagonistic functions, ultimately enabling rapid reversals upon perturbations (Kühn (2021)).

In the context of densely populated bacterial cells, the ramifications of impaired sensory perception, colloquially referred to as *blinded* cells, can be observed without explicit inference of their blindness. While a comprehensive assessment, including analysis of directional changes post-collision frequency, could establish such impairment, it is regrettably beyond the scope of our current investigation. Noteworthy is the observed reduced directional alterations post-collision and the tendency of these cells to exhibit group-following behavior, indicating that PilH-mediated motility enables sustained movement despite collisions, resulting in increased speed and extended trajectories. This phenomenon is corroborated by PIV analysis, which underscores the ”fluid-like” behavior of PilH-expressing cells. Contrary to expectations, PilG cells exhibit movement patterns in dense regions resembling those of wild-type (WT) cells, given the limitations of our measurements in distinguishing between WT and PilG behaviors. However, anomalous findings of heightened mobility and speed in dense regions for PilG cells suggest potential tracking inconsistencies, warranting cautious interpretation of these outcomes, as mentioned above.

4.2 Improvement and further analysis

As discussed before, the tracking methods could be improved on dense files. Either by using other parameters, totally change the tracking algorithm. We could also add some other cleaning methods such as checking the smoothness of the tracks. This would allow us to get rid of tracks that jump between several cells.To improve the analysis it may be smart to add more parameters like the number of direction changes or a specific nematic order for dense files. Another idea is to represent the traveled distance on 2D to see if there are more motion on certain zones than others. It would also be interesting to observe the behaviors of the cells after collision. Either a change in speed of the relation between the the direction of the two cells after colliding.
We essentially chose to improve the segmentation and the tracking instead of analyzing results. The use of some tools to visualize and asses the quality of the segmentation and tracking as well as a good understanding of the methods used were our priority. Segmentation and tracking took us a lot of time making the analysis part weaker for this project. We hope that what we learned on this first project could serve us for the second project where we will maybe spend more time analyzing the results.

Bibliography

M. Basaran. Large-scale orientational order in bacterial colonies during inward growth. 2021. URL https://doi.org/10.7554/eLife.72187.

K. J. Cutler. Omnipose: a high-precision morphology-independent solution for bacterial cell segmentation. 2022. URL https://doi.org/10.1038/s41592-022-01639-4.

Y. T. Fukai. laptrack, 2021. URL https://doi.org/10.5281/zenodo.5519537.

Y. T. Fukai and K. Kawaguchi. LapTrack: Linear assignment particle tracking with tunable metrics. *Bioinformatics*, page btac799, Dec. 2022. ISSN 1367-4803. doi: 10.1093/bioinformatics/btac799.

M. J. Kühn. Mechanotaxis directs pseudomonas aeruginosa twitching motility. 2021. URL https://doi.org/10.1073/pnas.2101759118.

W. Thielicke and R. Sonntag. Particle image velocimetry for matlab: Accuracy and enhanced algorithms in pivlab. *Journal of Open Research Software*, 2021. ISSN 9-12. doi: https://doi.org/10.5334/jors.334.