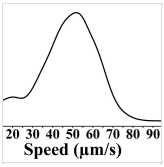

Mar 02, 2021

Chlamydomonas reinhardtii cell motility quantification

Joao Vitor Molino[1]

[1]University of Zürich

1 *Works for me* This protocol is published without a DOI.

Joao Vitor Molino
University of Zürich

SUBMIT TO PLOS ONE

ABSTRACT

This protocols describe the steps required for the motility quantification in *Chlamydomonas reinhardtii*.

PROTOCOL CITATION

Joao Vitor Molino 2021. Chlamydomonas reinhardtii cell motility quantification. **protocols.io** https://protocols.io/view/chlamydomonas-reinhardtii-cell-motility-quantifica-bsw5nfg6

CREATED

Mar 02, 2021

LAST MODIFIED

Mar 02, 2021

PROTOCOL INTEGER ID

47805

GUIDELINES

All steps described in this protocol are intended to be conducted in a research laboratory. Follow aseptic procedures.

DISCLAIMER:

Cell preparation

1 1. Culture the cells for 5 days following the growing *Chlamydomonas reinhardtii* protocol.
2. Prepare a sample of the cells by washing them in the final desired media by centrifugation.
3000 x g, 25°C, 00:03:00 , removal of old media and addition of new media.

Glass slide preparation

1. In glass slide, attach the Frame-seal Slide chambers, 15x15, 65 µl.

2 2. Add the sample
3. Seal the chamber with a cover slip

Imaging

3 1. Set the glass slide in the microscope for image acquisition
2. Record sequential images in a fixed interval (e.g. 100 ms per frame) using the desired laser/filter settings.

Cell tracking analysis - imageJ

4 1. Open the image file in imageJ
2. Adjust brightness and contrast (Image -> Adjust -> Brightness/Contrast -> auto)
3. Open the Plugins -> Tracking -> TrackMate
4. Check the image settings (eg. Size and pixels), next
5. Choose the detector system, next
6. Input the estimate size of cells and define a threshold, next
7. After detection, choose a range of spots quality, next
8. Select a view option, next
9. Set filter in spots, next
10. Define the maximum linking distance between frames, next
11. Set color for tracks, next
12. Open analysis and save the results in a csv file

*The analysis files contains information about the tracks, as average speed, max speed, position, etc.
** A video demonstrating the TrackMate workflow can be found here.

Data analysis

5 Analyse the data generated. For example, after organizing the data of each test in a excel tab, the R code below demonstrate a possible workflow and analysis to generate a density plot.

```
library(dplyr)
library(ggplot2)
library(readxl)

#set the working directory
wd <- setwd("D:/Path")

#Get all file names with .xlsx in the working folder.
FileNames <- Sys.glob("*.xlsx")


#load file with results

cc1690_1 <- read_excel("UZH20201206 Motility.xlsx", sheet = "1690_1")

cc1690_2 <- read_excel("UZH20201206 Motility.xlsx", sheet = "1690_2")

cc1690_3 <- read_excel("UZH20201206 Motility.xlsx", sheet = "1690_3")

cc1690 <- rbind(cc1690_1, cc1690_2, cc1690_3)


MAW8_1 <- read_excel("UZH20201206 Motility.xlsx", sheet = "MAW8_1")

MAW8_2 <- read_excel("UZH20201206 Motility.xlsx", sheet = "MAW8_2")

MAW8_3 <- read_excel("UZH20201206 Motility.xlsx", sheet = "MAW8_3")

MAW8 <- rbind(MAW8_1, MAW8_2, MAW8_3)


GP1_1 <- read_excel("UZH20201206 Motility.xlsx", sheet = "GP1_1")
```

```
GP1_2 <- read_excel("UZH20201206 Motility.xlsx", sheet = "GP1_2")

GP1_3 <- read_excel("UZH20201206 Motility.xlsx", sheet = "GP1_3")

GP1 <- rbind(GP1_1, GP1_2, GP1_3)


#Take the last 10 seconds of the experiment

cc1690 <- filter(cc1690, start_t > 50)

MAW8 <- filter(MAW8, start_t > 50)

GP1 <- filter(GP1, start_t > 50)


#Add column with Strain information

cc1690_n <- c("cc1690")
cc1690 <- cbind(cc1690, strain = cc1690_n)

MAW8_n <- c("MAW8")
MAW8 <- cbind(MAW8, strain = MAW8_n)

GP1_n <- c("GP1")
GP1 <- cbind(GP1, strain = GP1_n)



#Combine data
df <- rbind(cc1690, MAW8, GP1)

# Convert data to speed um/s (File recorder at 100ms/frame|speed in frames/s)
df$speed_r <- df$speed*10


#generates the plot
myplot <- ggplot(df, aes(speed_r, fill=strain))+
  geom_density(alpha=0.4) +
  scale_fill_manual(limits = c("cc1690",
                               "MAW8",
                               "GP1"),
                    values=c("#1b9e77",
                             "#d95f02",
                             "#7570b3"))+

  theme(text=element_text(size=12,
                          #       family="Comic Sans MS"))
                          #       family="CM Roman"))
                          #       family="mono"))
                          #       family="sans"))
                          family="serif",
                          face="bold"),
        plot.background = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_rect(fill='white', colour='white'),
        axis.line = element_line(colour = "black", size = 0.6), #thickness axis
        axis.ticks = element_line(size=0.8), #thickness ticks
```

```
        axis.text.x = element_text(face="bold", colour='black'),
        axis.text.y = element_text(face="bold", colour='black'),
        legend.key=element_blank())+
  labs(x = (expression(paste("Speed (",  #label x axis
                             mu, "/", "s",
                             ")",
                             sep=""))),
       y = "Density")  #  Lavel y axis


#to save graph
ggsave(
  paste("motility",".png"),
  myplot,
  width = 6.41,
  height = 3.73,
  dpi = 1200
)
```