**Cell Averaging Model**

Single cell crops were generated using eHooke1 while selecting only cells in phase 22.In each crop, pixels not corresponding to each respective cell were set to 0. A binary image of each crop was generated by selecting the pixels only above a threshold calculated using the isodata algorithm3,4. The outline of each cell was calculated by performing binary erosion on the binary image and subtracting the original binary image by the eroded one. Cells were aligned by first performing a Principal Components Analysis (PCA) on the outline coordinates to calculate the orientation of the major axis of each cell. Then, the angle of the major axis was calculated, and each single cell crop was rotated to make every cell have its major axis at a 90º angle. The average height and width of all cells was calculated, and each cell was resized to those values. The intensities of each cell image were normalized by setting the intensity range between 0 and 1, with 0 corresponding to the minimum intensity of each single cell crop and 1 corresponding to the maximum. To generate the average cell model, the normalized intensities were averaged by summing each pixel and dividing the total by the number of cells. Heatmaps were generated by calculating the background threshold using the isodata algorithm3,4 and setting the values below the threshold to white. Remaining pixels were colored by using the (adicionar nome do colormap escolhido entre aspas) colormap provided by the Python library matplotlib5, assigning the minimum and maximum of non-background intensity values of the cell models to the minimum and maximum values of the colormap.

1. Saraiva, B. M., Krippahl, L., Filipe, S. R., Henriques, R. & Pinho, M. G. eHooke: a tool for automated image analysis of spherical bacteria based on cell cycle progression. *Biological Imaging* 1–25 (2021) doi:10.1017/s2633903x21000027.

2. Monteiro, J. M. *et al.* Cell shape dynamics during the staphylococcal cell cycle. *Nature Communications* **6**, 1–12 (2015).

3. Dias Velasco, F. R. Thresholding using the ISODATA Clustering Algorithm. *IEEE Transactions on Systems, Man, and Cybernetics* **10**, 771–774 (1980).

4. van der Walt, S. *et al.* Scikit-image: Image processing in python. *PeerJ* **2**, e453 (2014).

5. Hunter, J. D. Matplotlib: A 2D graphics environment. *Computing in Science and Engineering* **9**, 90–95 (2007).