Accurate Segmentation of Bacterial Cells using Synthetic Training Data

Georgeos Hardo[1], Maximilian Noka[1], and Somenath Bakshi[1,*]

[1]Department of Engineering, University of Cambridge, CB2 1PZ, UK
[*]Corresponding author: Somenath Bakshi, sb2330@cam.ac.uk

January 2022

Abstract

We present a novel method of bacterial image segmentation using machine learning based on Synthetic Micrographs of Bacteria (SyMBac). SyMBac allows for rapid, automatic creation of arbitrary amounts of training data that combines detailed models of cell growth, physical interactions, and microscope optics to create synthetic images which closely resemble real micrographs, with access to the ground truth positions of cells. Models trained on SyMBac data generate more accurate and precise cell masks than those trained on human annotated data. This enables precise analysis of cell size regulation along changing conditions in a growing bacterial cell culture, revealing novel insights about the physiological dynamics of individual bacterial cells during entry and exit from dormancy.

Main

High-throughput time-resolved imaging of bacterial cells using microfluidic technologies have revolutionised the fields of quantitative bacterial physiology, systems microbiology, and synthetic microbiology [1, 2, 3, 4]. Deep-learning based approaches are rapidly gaining popularity for automated processing of such high volume of data. The performance of any such deep-learning algorithm is fundamentally dependent on the quality and volume of training data. However, the task of generating training data with accurate ground truth is not only slow and difficult but a near-impossible one for microscopic images of micron-sized bacterial cells. The point-spread function (PSF) of the microscope corrupts the images of tiny bacterial cells, blurring them to the extent that neither expert users nor traditional computational segmentation programs can accurately annotate the correct pixels. Additionally, geometric impacts in the microscopic 2D projection of 3D objects in an image lead to inaccuracies in contrast based segmentation by both humans and computational programs. Inaccuracies in the ground-truth of training data cause deep learning models to misinterpret the relations between the object and its image, resulting in systematically artifactual masks from test data. This limits one's ability to infer a cell's true size and shape from the image and confounds the analysis.

To address these limitations, we have developed SyMBac (Synthetic Micrographs of Bacteria), a tool to generate realistic synthetic micrographs of bacteria growing in microfluidic devices. Here, we focus our attention to the microfluidic device called the mother machine (Fig. 1a) (but the pipeline can be easily adapted to different imaging devices and modalities). We combine understanding of cell growth and morphology, its position and interaction with the imaging platform, and the microscope's optics to render synthetic images capable of training highly accurate segmentation CNNs without human annotated data. A neural network trained on this synthetic data can precisely learn about the corruption modes introduced during image formation, and as a result output accurate cell masks. This not only greatly speeds up the process of image segmentation (because no human

annotation is necessary), but most importantly generates more accurate masks of cells, enabling precise analysis of size regulation and growth-dynamics. Using our method, given any change in experimental settings, it becomes trivial to generate high volumes of synthetic images and ground truths to rapidly analyse new data. This also solves the robustness problem of image processing using deep-learning because models can now be easily adapted and benchmarked, since synthetic data can be generated at any desired spatial and temporal resolution.

In Figs. 1a-d, we show the overall process of generating synthetic training data for mother machine experiments. A rigid body physics simulation is combined with an optical model of the microscope to rapidly produce many synthetic micrographs with ground truth. These images can be tuned according to biology, the imaging platform, and whether the experiment is phase contrast or fluorescence (Fig. 1e). SyMBac can generate images and ground truth pairs orders of magnitude faster than human annotators (Fig. 1f). Human annotators are not only slow, but also inaccurate in the training data generation (Fig. 1g) and the results are highly variable across users (SI 1). On the other hand, we also find that the unsupervised approaches using Otsu or local thresholding or membrane images suffer from bias and lack of robustness towards size variation (SI 1). These suggests that these traditional approaches are inaccurate and imprecise in annotating cells, determining cell size and shape, and are incapable of producing high quality training data for ML algorithms. SyMBac addresses this critical issue, since the synthetic images from SyMBac are accompanied with perfect ground truth information of cell shape and position. This therefore helps the neural network to learn the true relation between object and its corresponding image.

Once realistic synthetic images are generated, a segmentation network (e.g U-net [5]) is trained. This network learns the relation between the synthetic phase contrast image and ground truth pair to infer image to object relationships (Fig. 1h). We evaluated the usefulness of synthetic data for this purpose by applying it to the DeLTA implementation of U-net [6], a popular and robust pipeline micrograph analysis. Since the model trained with SyMBac has access to perfect ground truths for synthetic data, it produces less flawed cell masks and visually more "natural" cell shapes (Fig. 1l and SI 9). To perform a rigorous comparison, a synthetic training dataset based on reference images from the original DeLTA paper was generated. In the the DeLTA paper, a semi-automatic tool was used to generate training data, which involves subjectivity in the choice of thresholding parameters and morphological operators. This results in training data with oddly shaped masks, which are consequently learned and then reproduced by the network during training and prediction (Fig. 1k). The network trained with SyMBac produces more realistic mask predictions compared with models trained with human curated training data (dataset from Lugagne et al. [6]). Furthermore, we observe that the DeLTA model trained with SyMBac show higher temporal coherence when compared to models trained on human annotated data (Fig. SI 16 and 1k) which improves segmentation precision and makes models robust to large variations in cell size (SI 8 and SI 9).

For high-throughput data, image processing algorithms must demonstrate low cell identification errors. Since SyMBac uses synthetic data for training, validation data is not available in the traditional sense. Instead, identification errors are used to evaluate the model as it learns (Fig. SI 11). Automatic error rate calculation is possible without validation data due to the predictable nature of bacterial growth in the mother machine (Fig. SI 13). In general, 100 epochs were enough to bring the identification error below 0.5%. We emphasise that models trained with SyMBac data require no pre-processing of training data, and no post-processing of output masks. To perform a comparison, we generated synthetic data matching the dataset provided by DeLTA, and were easily able to achieve comparable segmentation accuracy (table in Fig. 1l). This is remarkable, since it shows that SyMBac is able to generate realistic training data from just the reference images, which means one can use it to analyse pre-existing datasets. However, the true advantage of SyMBac is in its ability to generalise to experimental variations (as shown in Fig. 1e). When we tested against a data set collected with a 60x air objective, the pretrained DeLTA model could not produce any masks and therefore we could not quantify the error. Conversely, SyMBac can generate similar images and ground truth pairs, yielding highly accurate segmentation with an error rate of 0.36% (table (top) in Fig. 1j). DeLTA trained with SyMBac data also gave

18x better accuracy compared to pretrained DeLTA models for a separate 100x dataset with different imaging conditions (detailed in Bakshi et al. [7]). This analysis highlights two powerful attributes of SyMBac. Firstly, it is very easy to generate large volumes of high quality training data for new experiments where optics, illumination, platform, or the biology might have changed. Secondly, SyMBac enables high-accuracy segmentation from low resolution, low signal-to-noise data collected with air objectives. This will have important implications in using the mother machine for high-throughput screening (for persisters [7], or identifying optimal designs of synthetic circuits [4]), where low resolution air objectives provide the necessary travel distance, speed, and large field of view that is required for such experiments.

Segmentation algorithms (both ML and non-ML) face an even harder challenge when the data type itself changes during the experiment, for example cells changing size along a growth-curve experiment as they transition from exponential to stationary phase and back (Fig 2a) [7]. Since cells in the stationary phase are very small, the annotation error grows and compromises the performance of the ML algorithm (Fig. 1g). On the other hand, non ML models suffer from changes in object size, as algorithmic parameters need to be re-tuned. SyMBac can address this issue by generating high-quality training datasets for cells at any size (Fig. 1e), producing accurate cell masks throughout the experiment (Fig 2b). While models trained with small cell training data perform well in stationary phase, this comes at the cost of reduced accuracy in exponential phase, the converse is also true (bottom table in Fig. 1j). Instead, a complete dataset, which combines synthetic images of both types, leads to an overall better performance in identification accuracy and highly precise masks for estimating cell length and width (Fig 2c)

This precise estimate of cell size enables the computation of single cell growth rates during exponential phase, along the entry and exit from stationary phase along a growth curve (Fig 2d), and at different points within stationary phase (Fig. 2e). Upon entry to stationary phase, the cell growth rate dramatically reduces by 20-fold, to a population mean of $0.071\,\mu\text{m}\,\text{min}^{-1}$ and then asymptotically decreases as cells progress deeper into the stationary phase, as seen in Fig. 2e). During entry, the length and width of individual cells drop during division and settles at various stages as the division halts. This leads highly variable cell size in the stationary phase (Fig 2f, h, i), as demonstrated by the large population variance compared to single cell fluctuations in length and width. During resuscitation, cells increase their width and their length and reach homeostasis during exponential phase, as demonstrated by the comparable variability across the population and individual cells (2h, i). We also find that, after waking up, cells divide once they have reached a critical size (2j). When we look at the length and width regulation during this phase, we find that cells strictly regulate their width during wake up and each cell divides upon reaching a critical width, which is explainable, as the volume of a cell has a stronger dependence on width than length ($V = \pi r^2(1 - \frac{2}{3}r)$). The large variability in stationary phase coupled with this strict requirements for initiating exponential growth leads to strong correlation between the shape and size of individual cells and their resuscitation behaviour (sizer), with important implications for persistence towards antibiotics and population fitness in general.

In summary, SyMBac provides a major technical advancement for the implementation of machine learning approaches for analysing microscopy data of microbial systems. Through rapid and easy generation of high quality synthetic data with large coverage of experimental setups and cellular morphologies, SyMBac addresses a critical obstacle in the application of ML tools for high-throughput low-resolution single cell image analysis, eliminating the problems of user-subjectivity in training data creation and curation and easy adaptation to changing microbial species and experimental settings. The highly accurate masks of SyMBac have revealed novel insights into cellular growth and physiology under changing conditions and paves the way for systems-level analysis of the underlying factor using such high throughput data.

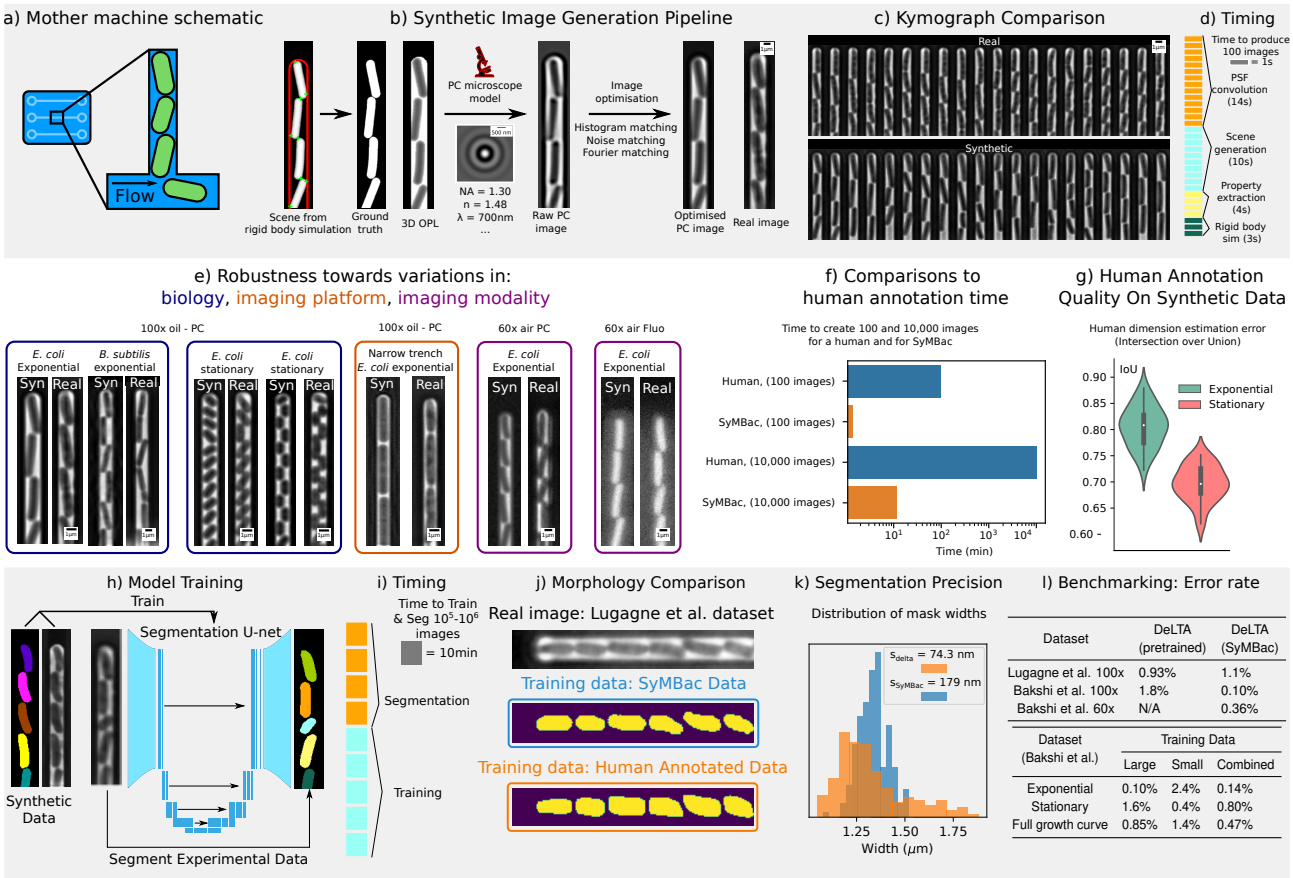

Figure 1:
a) Schematic of the microluidic device, the mother machine; which enables high-throughput time-lapse imaging of bacteria under precisely controlled conditions [8, 7]
b) Synthetic image generation pipeline: rigid body physics simulations are combined with agent based modelling to simulate bacterial growth in the microfluidic device. From these simulations, geometry of the device and each cell is used to draw ground truth masks and optical path length (OPL) images.Thee OPL is convolved with the microscope's PSF, which is generated using known parameters of the objective lens and phase ring/annulus, to generate the synthetic image. The intensity distribution and noise spectrum of the convolved image is then optimised to maximise its similarity to real images.
c) A kymogrpah of synthetic data generated by SyMBac, compared with a real kymograph from a mother machine experiment.
d) Typical timescales for individual steps in the generation of training data.
e) Synthetic data can be adapted to different biological conditions, imaging platforms, and imaging modalities. With real data, many experiments would need to be conducted to generate training data with the same kind of coverage.
f) SyMBac takes advantage of multiprocessing, and where possible offloads array operations onto the GPU, which allows it to be approximately 10,000x faster than a human at generating training data (10,000 images in less than 10 minutes)
g) Synthetic data is accompanied by reproducible, perfect masks. This means large sets of training data can be generated without error and the need for curation. Humans annotating images consistently undersegmented cells, especially in stationary phase when cells are smaller.
h) An outline of the U-net model being trained using synthetic data and then segmenting real data to produce accurate masks.
i) A typical time to train the network (on 2000 images) and segment approximately one million images (Nvidia GeForce 1080Ti).
j), k) The masks from SyMBac trained models are truer to the geometry of the cells, displaying no aberration when compared to model outputs trained on human annotated data. The masks also maintain a narrow distribution of widths, while the masks from DeLTA display wide variation with the peak shifted to lower values and show 2.5x higher variation in cell width.
l) Performance benchmarking of DelTA trained with SyMBac. Top: Models trained with SyMBAc perform well for both oil and air objectives and outperforms models trained with user-generated training data. Bottom: SyMBac enables generation of complete training data to tackle datasets with varying cell sizes, such as along the growth-curve. To optimise the performance throughput the entire dataset, models trained with a combination of large and small cells resulted in best performance across the entire growth curve (error rate 0.47%). Example kymographs are shown in SI 9. {fig:fig1}

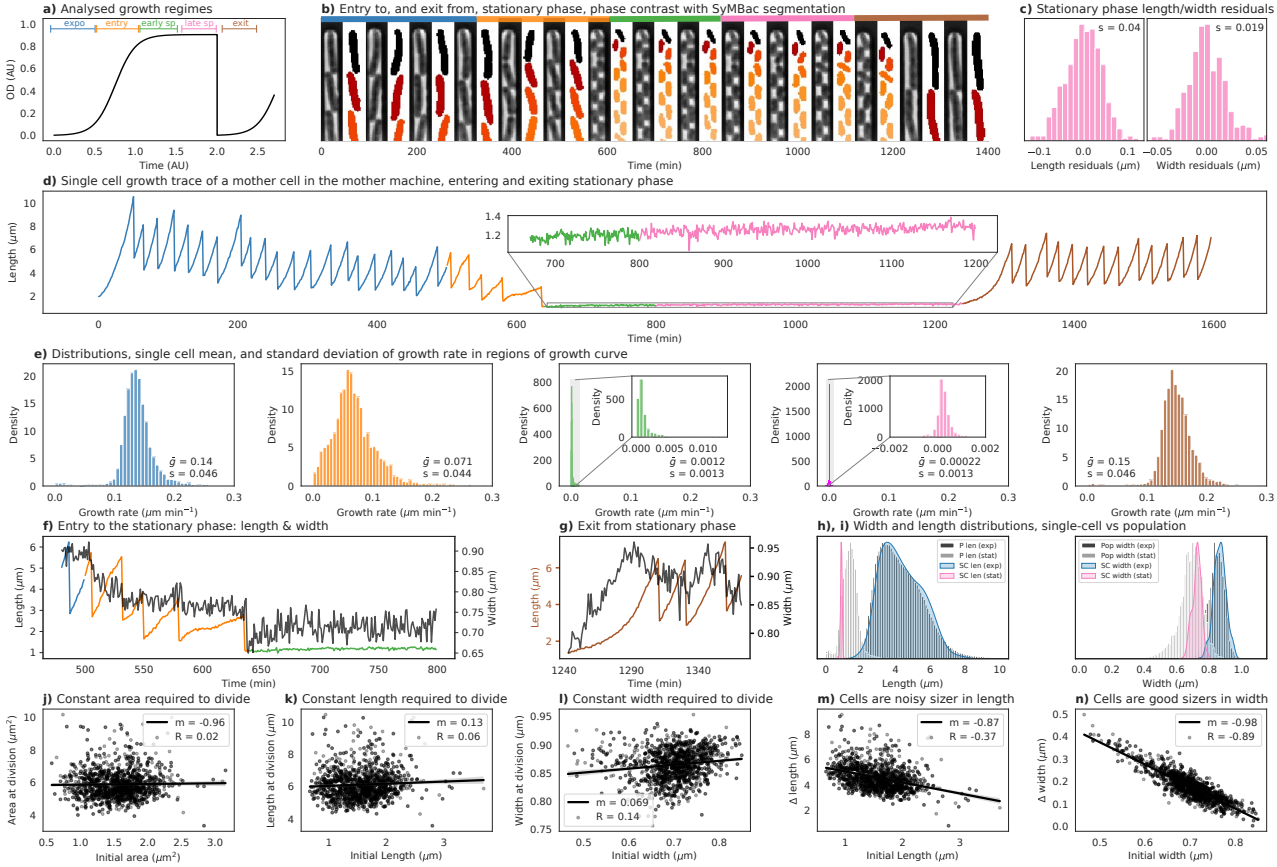

Figure 2: a) DelTA trained with SyMBAC was used to segment single cell data throughout all growth curve regimes (colour-coded and used throughout the figure). **b)** Sample kymograph with masks of a single mother machine trench going through the growth curve. **c)** The SyMBac trained model produces masks with precisions of 40 nm for length and 19 nm for width.

d) Example time series of cell size of a single cell going through an entire growth curve. Inset shows cell length changes during stationary phase.

e) During the exponential phase, Cells are exponentially increasing their size with a mean growth rate of $0.14\,\mu\text{m}\,\text{min}^{-1}$, which is equivalent to a population doubling time of 19.8 min, consistent with the bulk growth measurements of cells in this rich defined media [7]. The distribution of growth rates shifts to the left as cells enter stationary phase (orange and green phase) and eventually stops 6 hours into the stationary phase (pink). Upon exit from stationary phase (induced by a step change nutrient resumption in growth medium), cells exit dormancy and immediately start growing at the exponential growth rate $0.15\,\mu\text{m}\,\text{min}^{-1}$.

f) During the entry to stationary phase, cells not only shrink in length due to reductive divisions, but the width also gradually shrinks during a generation, 19 nm per generation, and $-6.9\,\mu\text{m}$ during division (1-sample t-test: $T = 12.9$, $p < 10^{-30}$).

g) The same trace as shown in **f)** but at the end of the stationary phase, showing the cell's width and length increase in the wake-up phase.

h,i) Comparison of the length and width distribution of the population and for an example cell over time. During exponential phase, individual cells sample the entire distribution of width and length, but they are locked into specific values during entry to stationary phase. This causes large variability across the population.

j), k), l) There is a constant area, length, and width required for the cell to divide irrespective of its initial width.

m) shows that the cell is a noisy length-sizer.

n) shows that the cell is an almost perfect width-sizer, dividing only when individual cells reach a critical width. {fig:fig2}

Methods

Image Generation

Our method employs several steps which make up a pipeline:

- Simulation of growth and division of cells and their interactions with each other and the microfluidic device.
- Extracting geometric information and rendering images of the bacteria and device in the scene.
- Microscope model for generating phase-contrast and fluorescence versions of these images.
- Image optimisation to further improve similarity to experimental images.
- Training ML segmentation algorithms with synthetic images.
- Performance testing.
- Model evaluation.

We give brief descriptions for each of these steps below, with additional information presented in the supplementary material.

Simulation of growth and division of cells and their interactions with each other and the microfluidic device

We built an agent based model (ABM) of cell growth and division, taking into account the three size regulation modes of bacteria (adder, sizer, timer), while also adding variability to the key parameters governing cell division. Cells are agents with properties such as length, width, position, age, and can be customised to include new properties such as time-dependent fluorescence intensity. These cell agents exist as dynamic objects in a rigid body physics world called a "scene", modelled using Python bindings of the popular physics engine Chipmunk (pymunk) [9]. We add static objects to these scenes which are shaped into a mother machine trench geometry, with configurable length and width. A global event loop keeps track of cell growth, divisions, inter-cellular collisions and cell-trench collisions at each time-step. A simple API was created to allow simulations to be run by entering the desired distributions of cell lengths and widths by defining the mean, variance and size-regulation mode, and then selecting the desired trench geometry (SI 2).

Extracting geometric information and rendering images of the bacteria and device in the scene

After simulations are complete, geometric information about the cell the trench position are extracted. This includes each cell's vertices in every time-point. The cell's 3D structure is a spherocylinder, which affects how light interacts with it, yet the simulation engine runs only in 2D. Therefore to render 3D cells, the cell's vertices are revolved along their major axis to generate each cell's spherocylindrical hull. All of these 3D cells are then morphed to simulate cell bending and then re-rendered and in a 3D scene, computationally represented by a 3D NumPy array. A 2D projection of this 3D array is then taken, such that an image is generated where the intensity in each pixel is proportional to the optical path length in that region (Fig. SI 6). The details of this calculation is provided in SI 3 This simulated image corresponds to what would be seen if a microscope had no PSF, and thus induced no diffraction artefacts in an image. From the simulated images at this stage, we also save the ground truth images of cell the positions.

Microscope model for generating phase-contrast and fluorescence images

The microscope model generates images from the scenes by convolving them with PSFs of relevant optics. First, the raw image is rendered at a high resolution (typically 3x the real pixel size) and then convolved with a variety of fluorescence or phase contrast PSFs (also rendered at higher resolution) depending on the application. After PSF convolution, the image is sub-sampled back to the original resolution. Convolution at higher resolution is more accurate, because sub-resolution features of the PSF are maintained, and propagated to the image. Otherwise, the PSF convolution results would be corrupted by low resolution artefacts. The user can select an ideal fluorescence PSF or phase contrast PSF which are modelled as an Airy disk and obscured Airy disk respectively (SI 4). The PSFs needs to be parameterised by inputting the microscope objective's numerical aperture, refractive index of the medium, emission wavelength, camera pixel size, and phase ring and annulus dimensions (details provided in SI 4). Upon defining the PSF of choice, it is convolved over the raw image to produce a synthetic micrograph.

Image Optimisation

Next, the user can use the optional image optimisation to maximise the similarity between the synthetic image and an example real microscope image. Multiple image manipulations are possible, including intensity histogram matching, noise matching, rotational Fourier spectrum matching, camera noise simulation, stage drift simulation, and even manual intensity corrections. This is done using an interactive IPython Notebook interface, where the user is presented with a side-by-side comparison of a synthetic image and a real image from their experiment. Two plots showing the relative errors in intensity and variance between cells, the mother machine device, and the space between cells are given such that the user can manipulate their image to minimise this error (SI 5 and Fig. SI 10). (Note: While examples of black box optimisers for image similarity matching are included, we avoid their use for this step due to the very noisy error-landscape between synthetic and real images. Moreover, we were unable to define a perfect objective function which guarantees perfect image similarity). Synthetic images are then generated according to the optimal parameters. These parameters can be varied according to a uniform or Gaussian distribution during the image generation process to simulate minor fluctuations in the experimental setup. This also acts as a form of data augmentation, but occurs during the image generation process, which preserves the mapping between object and image. For this reason we do not use data augmentation during the machine learning process.

Testing

Interfacing code for training machine learning models using DeLTA [6] and StarDist [10, 11] are provided. For most analysis, U-net models were trained with DeLTA, as this provided the most accurate masks. StarDist, while not pixel-perfect, can be useful for especially low magnification data (SI 6) because of its robust ability to separate densely packed star-convex shapes.

We tested SyMBac against 4 datasets, which have their details given in Table 1. 4 sets of data were generated, each with approximately 2000 synthetic images. 3 of the datasets are images of *E. coli* growing in a mother machine, taken with a 100x oil objective, and one dataset was imaged with a 40x air objective, using a 1.5x post-magnification lens, giving an effective magnification of 60x. However synthetic images for this dataset were still generated according to the optics of the 40x objective, and then scaled to the appropriate pixel size.

Model Evaluation

For evaluation purposes, the DeLTA implementation of U-net was used to train models using synthetic data. Because the data is purely synthetic, validation data in the traditional sense does not exist. For this reason we saved the model every epoch and evaluated performance through the analysis of growth data. Bacterial growth

Table 1: Datasets analysed using SyMBac

Dataset	Optics	Biology	Notable features
Lugagne et al. (exponential)	100x oil	*E. coli* in balanced growth	Poor contrast data
Bakshi et al. (exponential)	100x oil (NA = 1.30)	*E. coli* in balanced growth	High contrast data
Bakshi et al. (stationary)	100x oil (NA = 1.30)	*E. coli* in stationary phase	Small stacked cells
Bakshi et al. 60x (exponential)	40x air + 1.5x post mag (NA = 0.95)	E. coli in balanced growth	Low resolution + non-uniform illumination

{tab:tab1}

in the mother machine traces out a predictable saw-tooth wave (Fig. SI 12i). These saw-tooth waves and their derivatives were analysed for spurious peaks, which are the signature of over and under-segmentation errors (Fig. SI 13). Detailed discussion is included in SI 7.

The output from a U-net is a probabilistic threshold image. This image needs to be converted into a binary mask by thresholding pixel values between 0 and 1. Thresholds close to 0 will generate larger, connected masks, and thresholds close to 1 will generate smaller less connected masks. In order to identify the optimal threshold value which generates the most accurate masks, an independent set of synthetic data is segmented using the neural network, and the probability thresholds are adjusted to maximise the Jaccard index between the ground truth and the model output (Fig. 2c). Depending on the type of dataset this value can range between 0.6 and 0.99. If the thresholding value is low, then masks will be connected. To alleviate this we use seeded watershed segmentation to cut masks appropriately. Seeds are generated by first thresholding masks at very high probabilities ($P > 0.999$), and performing the watershed on the optimal probability binary mask. It must be noted that we do not perform any post-processing on the training data or the segmentation results, as this would affect the shape of the mask. After generation of the masks, information on cell curvature, radius (width) and length are calculated using the Colicoords Python library [12].

References

[1] Taheri-Araghi, S., Bradde, S., Sauls, J. T., Hill, N. S., Levin, P. A., Paulsson, J. *et al.* Cell-Size Control and Homeostasis in Bacteria. *Current Biology* **25**, 385–391 (2015).

[2] Robert, L., Ollion, J., Robert, J., Song, X., Matic, I. & Elez, M. Mutation dynamics and fitness effects followed in single cells. *Science* **359**, 1283–1286 (2018).

[3] Lord, N. D., Norman, T. M., Yuan, R., Bakshi, S., Losick, R. & Paulsson, J. Stochastic antagonism between two proteins governs a bacterial cell fate switch. *Science* **366**, 116–120 (2019).

[4] Luro, S., Potvin-Trottier, L., Okumus, B. & Paulsson, J. Isolating live cells after high-throughput, long-term, time-lapse microscopy. *Nature Methods 2019 17:1* **17**, 93–100 (2019).

[5] Falk, T., Mai, D., Bensch, R., Çiçek, Ö., Abdulkadir, A., Marrakchi, Y. *et al.* U-Net: deep learning for cell counting, detection, and morphometry. *Nature Methods* **16**, 67–70 (2019).

[6] Lugagne, J. B., Lin, H. & Dunlop, M. J. Delta: Automated cell segmentation, tracking, and lineage reconstruction using deep learning. *PLoS Computational Biology* **16**, e1007673 (2020).

[7] Bakshi, S., Leoncini, E., Baker, C., Cañas-Duarte, S. J., Okumus, B. & Paulsson, J. Tracking bacterial lineages in complex and dynamic environments with applications for growth control and persistence. *Nature Microbiology 2021 6:6* **6**, 783–791 (2021).

[8] Wang, P., Robert, L., Pelletier, J., Dang, W. L., Taddei, F., Wright, A. *et al.* Robust Growth of Escherichia coli. *Current Biology* **20**, 1099–1103 (2010).

[9] Blomqvist, V. Pymunk: A easy-to-use pythonic rigid body 2d physics library (version 6.0.0) (2007). URL `https://www.pymunk.org`.

[10] Weigert, M., Schmidt, U., Haase, R., Sugawara, K. & Myers, G. Star-convex Polyhedra for 3D Object Detection and Segmentation in Microscopy. In *The IEEE Winter Conference on Applications of Computer Vision (WACV)* (2020).

[11] Schmidt, U., Weigert, M., Broaddus, C. & Myers, G. Cell Detection with Star-Convex Polygons. In *Medical Image Computing and Computer Assisted Intervention - MICCAI 2018 - 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II*, 265–273 (2018).

[12] Smit, J. H., Li, Y., Warszawik, E. M., Herrmann, A. & Cordes, T. Colicoords: A Python package for the analysis of bacterial fluorescence microscopy data. *PLoS ONE* **14**, e0217524 (2019).

Code availability

Code is available at `https://github.com/georgeoshardo/SyMBac`

Acknowledgements

We thank Hannah Earley for very detailed feedback on the manuscript. We thank the Dunlop Lab for producing the easy to use DeLTA, and for their permission to include the source code in SyMBac's codebase.

This research in the S.B.'s laboratory was supported by the Wellcome Trust Award [grant number RG89305]; a University Startup Award for Lectureship in Synthetic Biology [grant number NKXY ISSF3/46]; Royal Society Research Grant Award (Award number G109931); and the Ph.D. Studentship from BBSRC [grant number NMZR/077 (to Georgeos Hardo)].

Author Contributions

G.H: Conceptualisation, Methodology, Software, Experiments, Visualisation, Analysis, Writing. M.N: Conceptualisation, Methodology, Software, Pilot and Feasibility Study. S.B: Conceptualisation, Experiments, Analysis, Writing.