Computational and Structural Biotechnology Journal 18 (2020) 1259–1269

COMPUTATIONAL AND STRUCTURAL BIOTECHNOLOGY JOURNAL

journal homepage: www.elsevier.com/locate/csbj

Deep learning predicts microbial interactions from self-organized spatiotemporal patterns

Joon-Yong Lee [a], Natalie C. Sadler [a], Robert G. Egbert [a], Christopher R. Anderton [b], Kirsten S. Hofmockel [b,c], Janet K. Jansson [a], Hyun-Seob Song [a,d,e,*]

[a] Biological Sciences Division, Pacific Northwest National Laboratory, Richland, WA, USA
[b] Environmental Molecular Sciences Laboratory, Pacific Northwest National Laboratory, Richland, WA, USA
[c] Department of Ecology, Evolution and Organismal Biology, Iowa State University, Ames, IA, USA
[d] Department of Biological Systems Engineering, University of Nebraska-Lincoln, Lincoln, NE, USA
[e] Nebraska Food for Health Center, Department of Food Science and Technology, University of Nebraska-Lincoln, Lincoln, NE, USA

ARTICLE INFO

Article history:
Received 30 December 2019
Received in revised form 16 May 2020
Accepted 17 May 2020
Available online 29 May 2020

Keywords:
Machine learning
Agent-based modeling
Soil microbiomes
Microscopy imaging technology
Network inference

ABSTRACT

Microbial communities organize into spatial patterns that are largely governed by interspecies interactions. This phenomenon is an important metric for understanding community functional dynamics, yet the use of spatial patterns for predicting microbial interactions is currently lacking. Here we propose supervised deep learning as a new tool for network inference. An agent-based model was used to simulate the spatiotemporal evolution of two interacting organisms under diverse growth and interaction scenarios, the data of which was subsequently used to train deep neural networks. For small-size domains (100 μm × 100 μm) over which interaction coefficients are assumed to be invariant, we obtained fairly accurate predictions, as indicated by an average R^2 value of 0.84. In application to relatively larger domains (450 μm × 450 μm) where interaction coefficients are varying in space, deep learning models correctly predicted spatial distributions of interaction coefficients without any additional training. Lastly, we evaluated our model against real biological data obtained using *Pseudomonas fluorescens* and *Escherichia coli* co-cultures treated with polymeric chitin or N-acetylglucosamine, the hydrolysis product of chitin. While *P. fluorescens* can utilize both substrates for growth, *E. coli* lacked the ability to degrade chitin. Consistent with our expectations, our model predicted context-dependent interactions across two substrates, i.e., degrader-cheater relationship on chitin polymers and competition on monomers. The combined use of the agent-based model and machine learning algorithm successfully demonstrates how to infer microbial interactions from spatially distributed data, presenting itself as a useful tool for the analysis of more complex microbial community interactions.

© 2020 Published by Elsevier B.V. on behalf of Research Network of Computational and Structural Biotechnology. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

Communities of soil microorganisms play a key role in controlling the global biogeochemical cycle. Soil microbial activities in soil include cycling of organic carbon compounds, which exert a significant impact on atmospheric CO_2 concentrations [1,2]. A mechanistic understanding of microbial interactions underpinning cycling of C and other nutrients remains a grand challenge due to the overwhelming compositional and functional complexity of the soil microbiome. Soil microbial communities dynamically shift membership, interactions, and functions across both spatial and temporal scales [3]. Thus, gaining a predictive understanding of microbial interactions in soil is critical for future efforts to control community dynamics and ecosystem function. However, the ability to predict microbial interactions in such a highly complex multidimensional space is currently lacking.

Soil microorganisms self-assemble into specific spatial patterns that are governed both by the metabolic capacity of individual members (e.g. quorum sensing and antibiotic production), and by myriads of environmental variables [4–6]. Microbial spatial patterns formed at microscales are thus a result of the dynamic interplay among populations and can consequently serve as a key input for predicting interspecies interactions. Rapid advancement in experimental and instrumental technologies is enabling the generation of high-resolution and high-throughput images that enable

* Corresponding author at: 209 L.W. Chase Hall, P.O. Box 830726, 3605 Fair Street, Lincoln, NE 68583-0726, USA.
E-mail address: hsong5@unl.edu (H.-S. Song).

https://doi.org/10.1016/j.csbj.2020.05.023
2001-0370/© 2020 Published by Elsevier B.V. on behalf of Research Network of Computational and Structural Biotechnology.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

visualization of the spatial distribution of microorganisms at the microscopic scale [7–9]. For example, image data from fluorescence microscopy provides clues to interspecies interactions by quantifying cell-to-cell spatial distances and cellular aggregation patterns [5]. In a previous study, imaging techniques were used to determine that inter-cell distances were shorter in surface soil and lengthened with depth [10], indicating relatively closer microbial interactions near the soil surface. Despite these advancements, spatial data *per se* do not provide direct information on microbial interactions.

One promising approach to overcome this gap is computational network inference. Conventionally, this approach uses species abundance data as a primary input to predict interspecies interactions based on similarity metrics, through data-model fitting, or by setting up certain interaction rules [11,12]. These methods have proven effective in inferring how species are interacting at a population level (e.g., [13,14]) but have yet to be extended to incorporate spatial organizations of microorganisms. Here, we propose supervised deep learning as a new network inference tool. As a subfamily of machine learning, supervised learning trains an empirical model based on the known relationships between inputs and outputs and provides the most likely predictions of outputs for new inputs [15]. Artificial neural networks are frequently used for this purpose [16]. Neural networks are composed of hidden layers between input and output layers and get 'deeper' as the number of layers increases. Implementation of deep network models requires significant computational power, as well as a sufficiently large set of labeled data for training the network [15]. Due to these challenges, the use of a neural network for machine learning has previously been limited to structurally simple nets. However, various computational advances and breakthroughs over the last decade are now poised to enable overcoming the limitations associated with training, as exemplified by the use of deep learning that has come into the spotlight with unprecedented achievements in solving many historically challenging problems in diverse scientific fields [17–21]. In a recent study, the utility of deep learning models has also been extended to analyze bacterial colony structures for classification [22].

Training deep neural networks directly from experimental microscopy image data to infer microbial interactions is currently infeasible due to unknown input–output relationships (i.e., inappropriate for supervision) and insufficient data size (i.e., scanty for deeper networks). To avoid such limitations, agent-based models can be used as a tool for data generation under diverse growth and interaction contexts. In contrast with traditional population-level ecological models that neglect the variance of growth properties in individual cells, agent-based models account for phenotypic heterogeneity in a population by viewing individual cells as autonomous entities [23]. Consequently agent-based models enable predicting the formation of microbial assembly patterns in space as a community's higher-order property emerging from the interaction of individual cells with each other and with the environment. This high-fidelity simulation tool has been used for various applications, e.g., to examine how microbial growth on soil surfaces is localized in dry compared to wet conditions [24,25].

In this work, we present a new pipeline of network inference that synergistically combines agent-based models and deep learning techniques to predict microbial interactions from both *in silico* and actual image data (Fig. 1). Through case studies of two interacting organisms temporally co-evolving in a two-dimensional space, we demonstrate that our deep learning networks not only provide accurate predictions of microbial interactions, but also enable extracting local variations of interaction parameters in heterogeneous environments. In the subsequent application to the analysis of real microscopy images, deep learning models that were trained using *in silico* image data correctly predicted shifts in microbial interactions across different culture conditions. This approach has potential to significantly improve our understanding of how microorganisms colonize habitats and interact with each other in spatially heterogeneous environments such as soils.

2. Materials and methods

2.1. Agent-based models for generating spatiotemporal patterns of interaction species

2.1.1. Growth rules

Agent-based models were designed to simulate two interacting microbial species (denoted by 'G' and 'R' hereafter) that were co-growing in a two-dimensional space. Simulations were performed in two different sizes of computational domains (i.e., 100 × 100 and 450 × 450 grids). We set the size of each grid to represent the physical dimension of 1 μm × 1 μm and we assumed that it can be occupied by only one microbial cell. Thus, each grid can take three states: occupancy by 'G', occupancy by 'R', or vacancy.

As an initial condition, we randomly distributed 'G' and 'R' cells (50/50 ratio) to fill about 5% of the computational domain. The growth of individual cells was modeled based on the following equation [26,27]:

$$r_i = \left[r_{i0} + a_{ij}\varphi_j(1-\varphi_i)\right]\left[1-\chi\left(\varphi_i+\varphi_j\right)\right], (i,j) \in \{(G,R),(R,G)\} \quad (1)$$

where r_i denotes the actual growth rate, r_{i0} is the basal growth rate (i.e., the growth rate without the influence of neighbor species or in the absence of space limitation), φ_i and φ_j are fractional occupancies of species i and j in a defined domain of interaction, *DOI* (i.e., the area where species influence each other), $a_{ij}\varphi_j(1-\varphi_i)$ represents the effect of species j on the growth of species i, where a_{ij} is the interaction coefficient (i.e., the effect of species j on the growth of species i), and the term $\varphi_j(1-\varphi_i)$ accounts for the increase and decrease of the effect of the partner with the abundance of species j and species i. Therefore, '$1-\chi\left(\varphi_i+\varphi_j\right)$' reflects the decrease of r_i due to increased intra- and inter-specific competition when the total occupancy in the interaction domain increases.

2.1.2. Simulation of assembly patterns using a single set of parameters

We performed 5000 Monte-Carlo simulations using the agent-based model in the computational domain with 100 × 100 grids. We assumed that the parameter values governing the growth and interaction dynamics of two species are homogeneously distributed in this small space. Thus, in each simulation, the model was assigned with a single set of parameters that were randomly sampled over the following ranges: $0 \le r_{i0} \le 0.1$, $-1 \le a_{ij} \le 1$, $0 \le \chi \le 1$, and 3 × 3 grids ≤ *DOI* (the size of interaction domain) ≤ 11 × 11 grids. These parameter ranges reflect the conditions commonly considered in the literature [27]. We applied periodic boundary conditions on the four sides of the computational domain. The datasets obtained from this setting are used as the primary source to develop neural network models (Section 3.2).

2.1.3. Simulation of assembly patterns using a composite set of parameters

We extended the agent-based model simulation to the computational domain with 450 × 450 grids. For this relatively larger domain, we released the previous assumption of homogeneous parameter distribution. As illustrated in Supplementary Fig. S1, we divided the computational domain into five subdomains, each of which is governed by a distinct set of parameters. The way we assign parameter values in each subdomain is the same as that

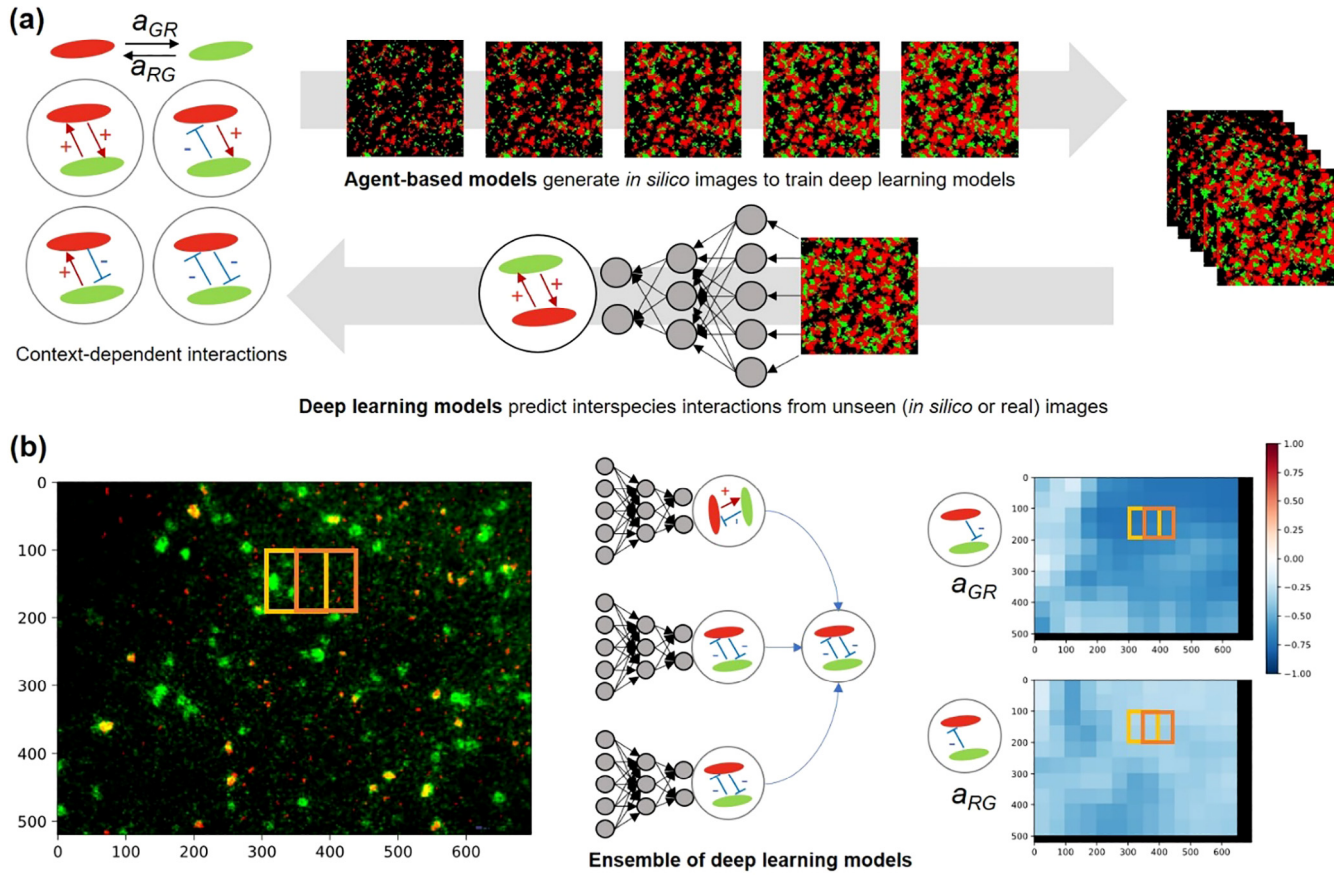

Fig. 1. Deep learning pipeline to infer microbial interactions from microbial assembly patterns: (a) training deep learning networks using *in silico* images generated from agent-based models to predict interactions from new test (or unseen) datasets. a_{GR} and a_{RB} represent the effect of R (red) on the growth of G (green) and the effect of G on the growth of R, respectively; (b) prediction of spatial variation of interaction coefficients in real images using a sliding window method (left panel), final prediction determined by taking averages from an ensemble of best-performing deep learning models (middle panel), and interaction heatmaps generated by integrating neighboring sliding window estimations (right panel). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

described in the previous section. These datasets were used for further testing neural network models in heterogeneous environments (Section 3.3).

2.1.4. Dynamic stochastic simulation of microbial growth

Cellular growth in space is often described as a stochastic process. To reflect this nondeterministic aspect of growth, we ran dynamic simulations of the agent-based model using the Gillespie's stochastic simulation algorithm (SSA) [28,29]. In each time instance, the SSA algorithm independently determines the following two random variables: 1) the time until the next cell growth occurs (a continuous random variable), and 2) the index of the cell to grow (a discrete random variable). For the details of implementing the SSA, we refer the reader to a review article by Higham [30].

2.2. Development of deep artificial neural net models

2.2.1. Neural network architectures

We employed plain convolutional neural networks (CNNs) [31,32] and residual networks (ResNets) [33], the effectiveness of which has been demonstrated in various fields, especially in image processing. Similar to Visual Geometry Group (VGG) networks [31], our plain CNN networks consist of a stack of convolution layers, followed by fully connected layers. Specifically, we considered 3 to 6 convolution layers and 2 to 3 fully connected layers so that we tested for eight plain CNN structures with different depths (Supplementary Table S1(a) and Supplementary Fig. S2(a)). We also tested the effect of the width of the convolution layers, which started from 8 or 16 in the first layer and then increased by a factor of 2 after each pooling layer, until it reached 32 or 64. For the pooling layer, we applied the 2D average pooling and max pooling in each network configuration.

For deeper network architectures, we tested the ResNets, where the identity shortcut connections (or residual connections) are simply inserted in the plain networks (see Supplementary Table S1(b) and Supplementary Fig. S2(b)). We adapted the ResNet-18 and ResNet-34 structures [33] to fit our data. We tested the basic building blocks of two 3×3 convolution layers and the bottleneck blocks of three convolution layers of 1×1, 3×3, and 1×1 as well. In addition, we compared the original residual networks with the pre-activation residual networks introduced in [34], which improved accuracy and reduced overfitting by using identity shortcut connections and identity after-addition activation (Supplementary Fig. S2(c)). For convolutional layers, the rectified linear unit (ReLU) activation function [35] and batch normalization (BN) [36] were adopted in each convolution to improve the trainability and stability of deep neural networks. BN is prevalent for deeper models since it allows higher learning rates and less careful initialization by employing a way to normalize activations in intermediate layers. Details of the algorithms can be found in these papers [31,33,34].

During training, the input to the convolution layers was set with a fixed-size (e.g., 100×100) ternary image with a single channel that is generated from the agent-based model. We used 1, −1, 0 to denote to denote occupancy by 'G', occupancy by 'R', and vacancy, respectively.

To infer a set of interaction coefficient values, we put a linear layer with the same number of output nodes at the end layer. In our case study of two species, we set the number of nodes for the final linear layer to be two to predict asymmetric binary interaction coefficients.

2.2.2. Hyperparameters

We performed hyperparameter optimization to search for the best neural net models. We considered the following parameters as a set of hyperparameters to optimize: configurations of convolution layers and fully-connected layers (see Supplementary Table S1), mini-batch sizes, learning rates, optimization algorithms, dropout probabilities, and L2 regularization coefficients (see Supplementary Table S2 for details). Instead of performing a time-consuming exhaustive search in such a large parameter space, we pre-determined 300 sets of hyperparameters through random selection and performed random grid search.

Out of 35,000 *in silico* images (=5,000 simulation conditions × 7 time points) obtained from agent-based model simulations, we set aside the 20% of the data for testing and used 80% of the data for developing deep neural network models, which were in turn split into five groups for *k*-fold cross validation [37]. In each training, the best hyperparameter set was determined based on the minimum average validation loss for the validation dataset.

2.2.3. Model training and early stopping

We trained deep neural networks following the standard supervised training procedure [31]. Using the commonly used mini-batch algorithms: mini-batch gradient descent (GD) with momentum and Adam [38,39], we determined parameters in deep learning networks such that the following mean squared error (MSE) is minimized:

$$MSE = \frac{\sum_{k=1}^{n}\sum_{i,j}\left(a_{ij}^{k} - \widehat{a}^{k}{}_{ij}\right)^{2}}{2n}, (i,j) \in \{(G,R),(R,G)\} \tag{2}$$

where the a_{ij} and $\widehat{a}_{ij}$ denote the known (or target) and predicted interaction coefficients, respectively, and n is the number of batch images.

To avoid overfitting, we used the validation loss improvements for early stopping [40]. We set the patience to be 20 during 200 epochs. That is we stopped the training if the performance against the validation set did not show any improvement after twenty consecutive epochs in a maximum of 200 epochs. For initialization of the neural network weights, we used the same strategies as the one in [33], which helps to avoid learning stagnation, a common problem encountered in training deep networks.

2.2.4. Performance metric for model inference

To measure the model performance, we took the following couple of commonly used metrics: root mean squared error (RMSE) and R^2 (coefficient of determination), i.e.,

$$RMSE = \sqrt{\frac{\sum_{k=1}^{n}\sum_{i,j}\left(a_{ij}^{k} - \widehat{a}^{k}{}_{ij}\right)^{2}}{2n}}, (i,j) \in \{(G,R),(R,G)\} \tag{3}$$

$$R^2 = \frac{1}{2}\sum_{i,j}\left(1 - \frac{\sum_{k=1}^{n}\left(a_{ij}^{k} - \widehat{a}^{k}{}_{ij}\right)^{2}}{\sum_{k=1}^{n}\left(a_{ij}^{k} - \overline{a}^{k}{}_{ij}\right)^{2}}\right), (i,j) \in \{(G,R),(R,G)\} \tag{4}$$

where $\overline{a}_{ij}$ denotes the average of the true a_{ij}'s.

2.2.5. Sliding window estimator to predict interactions in larger and irregular sized images

In order to extend the pre-trained models to image data with larger size than the input data (used for training), we adopt the sliding window method, which has been widely used for object detection in images [41,42]. A basic idea is to scan the larger size image using the smaller-size square window that slides with a fixed step size. The size of window is the same as that of the images used for training. While any integer value can be considered for the step size, we recommend setting it as a factor of the input size (i.e., window size) for convenient interpretation. The numbers of sliding windows across rows and columns (N_{row} and N_{col}) are as follows:

$$N_{row} = (h - n_i)\%n_s + 1 \tag{5}$$

$$N_{col} = (w - n_i)\%n_s + 1 \tag{6}$$

where n_s, w, h, and n_i denote the step size, image width and height, and input size (=window size), respectively. The input size n_i is divisible by the step size n_s. The % represents the integer divide operator that divides two numbers and returns the integer part of the result. As the step size gets smaller, the computational cost to analyze a test image should get higher due to the increase of N_{row} and N_{col}. For simplicity, we discarded the residual part in the right and bottom edges in test images.

As the sub-image in each window was set to fit the model input, prediction of interaction values for each window is straightforward. With the step size typically smaller than the window size, grid blocks will be generated by overlapping multiple windows. Thus, the values of interaction coefficients in each of the grid blocks can be estimated as the averaged interactions of associated neighboring windows, which overlays the grid block. In the same manner, we can also estimate the variability of the neighbor interactions, which allows to quantify the local variance of interactions. That is, if the predicted interactions of the neighboring windows of the overlapping block has large variance, it can be said that this local area is likely to be governed by a composite of different context-dependent relationships.

To validate this approach, we defined the target interaction coefficients $a_{ij,S}^{*}$ of each sliding window S that contains K subdomains governed by different interaction parameters as follows:

$$a_{ij,S}^{*} = \sum_{k=1}^{K} w_{k,S}a_{ij,S}^{k}, (i,j) \in \{(G,R),(R,G)\} \tag{7}$$

where k indicates a subarea in a window that has a different interaction and $w_{k,S}$ represents the area ratio of the subarea k to the total area of the sliding window S.

2.2.6. Pre-processing of real images

To infer the interaction coefficients from the real images, we transformed the RGB images into ternary images (2D array of 1 s, −1 s, and 0 s) to be available for the trained models. For identifying red and green colors, we used the following color ranges in an HSV color space: (H:36, S:25, V:25) to (H:70, S:255, V:255) for green colors and (H:-20, S:25, V:25) to (H:20, S:255, V:255) for red colors. To get prediction for the sliding windows, split images are fed to the ensemble of trained deep learning models.

2.2.7. Deep learning implementation

In silico image data generated using the agent-based model were stored in the mat file format. For the implementation of deep learning methods, we used PyTorch 1.0.1 (Python 3.6), an open source deep learning software library for the Python programming language (https://pytorch.org/). We employed the Python packages such as pandas to convert the mat files to pickle format, scikit-learn to efficiently manage training/test data, and OpenCV to

import and process the real images to be fit to the deep learning models.

We used the GPU-equipped machine to train deep learning models, which were configured as follows: Intel(R) Xeon(R) CPU E5645 @ 2.40 GHz CPU, 96 GB RAM, and Nvidia GeForce GTX 1080 Ti with 12 GB VRAM. Using this hardware configuration, training a ResNet-18 model with pre-activation blocks took approximately 6.4 h on average. The use of GPU was critical - model inference for an input took about 1.62 ms on average in the GPU mode, but took about 1.05 s in the CPU mode.

2.3. Experimental design and microscopy

2.3.1. Construction of fluorescent bacterial strains

Red and green fluorescent bacterial populations for co-culture studies were prepared as follows. We modified a non-chitinolytic *E. coli* K-12 BW25113 △chiA, a chitinase gene chiA deletion strain from the Keio Knockout Collection (Dharmacon) [43], to include a genomic insertion of constitutively expressed mScarlet-I red fluorescent protein (RFP). To do so, we used a pMRE-Tn5-155, a miniTn5 plasmid with mScarlet-I RFP cargo, gifted from Mitja Remus-Emsermann (Addgene plasmid # 118545; http://n2t.net/addgene:118545; RRID:Addgene_118545) [44].

Conjugation with the Tn5 mScarlet-I RFP plasmid was performed as follows: *E. coli* △chiA recipient was grown on Luria-Bertani (LB) (Lennox; Sigma Aldrich) agar supplemented with 50 mg L^{-1} kanamycin media for 24 h. Donor strain *E. coli* S17-1 was streaked and cultured overnight on LB solidified with 1.5% agar (Difco, BD) supplemented with 100 mg L^{-1} Ampicillin (Sigma Aldrich) to maintain the pMRE-Tn5 plasmid. Freshly grown bacteria were harvested and resuspended in 1× phosphate buffered saline (PBS) to reach an $OD_{600\ nm}$ of 1.0 by spectrophotometer (Genesys20, Thermo Scientific). Donor and recipient strains were mixed in a 1:1 ratio and concentrated by centrifugation (4000*g*, Room temperature, 5 min) to reach an estimated $OD_{600\ nm}$ of 20. The bacterial mix was drop spotted on Luria-Bertani (LB) agar and incubated for 18 h at 37 °C. After incubation, the cells were harvested and resuspended in 1 mL 1× PBS. The *E. coli* S17 strain is an auxotroph for Proline and Arginine [44], so MOPS minimal media [45] agar supplemented with 0.2% glucose and 50 mg L^{-1} kanamycin (Sigma Aldrich) was used to select for mScarlet-I RFP transformed *E. coli* △chiA. This selection was carried out four additional times and colony fluorescence was assessed by fluorescent plate scans conducted in a FluorChemQ (ProteinSimple) fluorescent imaging cabinet. Highly fluorescent colonies were selected and cultured in LB with 50 mg L^{-1} kanamycin and prepared as 10% glycerol stocks for long term storage at −80 °C.

We engineered reporter strain bRE007 by transposon insertion of a fluorescent protein reporter into strain *Pseudomonas fluorescens* SBW25. The reporter cassette consists of a superfolder GFP gene (*sfGFP*) translationally fused to a chlorampenicol actetyltransferase gene (*cmR*) expressed from a highly efficient constitutive promoter (apFAB46). Briefly, to generate the strain we transformed a nucleoprotein complex of hyperactive Tn5 transposase (EZ-Tn5, Lucigen) and a PCR-amplified sfGFP-cmR cassette that includes the inverted repeat sequences recognized by Tn5. We prepared electrocompetent cells by growing SBW25 to OD 0.4 and washing multiple times in pre-chilled 10% glycerol. The cells were electroporated (1800 V, 200 O, 25 μF) with 100 ng of the amplified cassette in a 1 mm gap cuvette, recovered in LB broth at 30 °C for 2 h, and plated on LB agar supplemented with 34 μg mL^{-1} chloramphenicol. Highly fluorescent colonies identified by blue-light transillumination were plated onto fresh LB agar plates supplemented with 34 μg mL^{-1} chloramphenicol to obtain isogenic colonies. To identify the sfGFP-cmR insertion site, the isolated genome (DNeasy, Qiagen) was subjected to Nextera fragmentation (Illumina) followed by paired end Illumina sequencing (SNPSaurus, Oregon, USA). De novo assembly of sequence reads that did not match the SBW25 genome via Geneious was used to verify the presence of the sfGFP-cmR cassette and the insertion location. The reporter cassette inserted within gene PFLU_RS22785 (locus 5127878) is in a reverse orientation.

2.3.2. Coculture experiments

Bacterial strains were cultivated for 24 h in 5 mL of LB broth without antibiotics for wild type strains and either 20 mg L^{-1} chloramphenicol or 50 mg L^{-1} kanamycin for *P. fluorescens* SBW25 mNeonGFP or *E. coli* △chiA mScarlet RFP, respectively. To prepare culture chambers, silicon 4 well micro-inserts (Ibidi) with well dimensions of 2.0 × 1.5 × 4.2 mm (w × l × h) were secured to 25 × 75 mm, #1 (~0.13–0.17 mm) float glass coverslips (Chemglass Life Sciences) and wrapped in foil before autoclaving for 30 min, 121 °C. Next, chitin beads were fluorescently stained and prepared for microcosm incubations as follows: 100 μL of chitin resin (New England Biosciences) in 20% ethanol suspension was transferred to a 1.7 mL Eppendorf tube and centrifuged at room temperature (RT) for 2 min at 4000*g* (5810 R, Eppendorf). The supernatant was removed, and the chitin resin was resuspended in 20% Calcofluor White (Sigma Aldrich) in 1× PBS. The resin was stained in the dark at RT for 30 min, buffer exchanged twice, and diluted with enough filter sterilized MOPS minimal media to achieve ~10 beads (each bead ~50–120 μm) per μL as assessed by 10× light microscopy (MicroStar, American Optical). For the microcosms receiving homogenous carbon substrates, either the chitin monomer N-acetyl glucosamine (Sigma Aldrich), or pentaacetyl-chitopentaose (Megazyme) which is a chitin oligomer composed of only 5 N-acetyl glucosamine monomers, were prepared as 0.2% (w/v) in MOPS minimal media and filter sterilized with Millex 0.22 um PES syringe filters (Millipore). To set up the C substrate explicit microcosms, 1 μL of Calcofluor White stained chitin beads in MOPS minimal media was dispensed into each well followed by as additional 7 μL MOPS media. For the homogenous substrate cultures, 8 μL of 0.2% pentaacetyl-chitopentaose or 0.2% n-acetyl glucosamine in MOPS minimal media was dispensed into wells. The microcosm chambers were then wrapped in Parafilm (Bemis Company, Inc) until inoculation. To prepare the inoculum, 10 or 2 μL *P. fluorescens* and *E. coli* cultured to of 0.40 $OD_{600\ nm}$ (3.2×10^8-cell mL^{-1}) were centrifuged (6000*g* at RT) to pellet cells. Spent media was removed and cells were resuspended in 3.2 mL of fresh MOPS minimal media. Cell suspensions were then mixed in a 1:1 ratio and microcosm were inoculated with 1 μL of cell suspension

2.3.3. Microscopy imaging

For confocal laser scanning microscopy, co-culture chips were placed onto a Leica DMI6000 microscope equipped with a CSU 10 Confocal scanning unit (Yokogawa Corporation of America, Sugar Land, TX). Approximately 10 random fields for each culture were selected for imaging as follows: Calcofluor White stained chitin beads (355 nm Ex; 460/50 m Em), GFP (488 nm Ex; 525/50 m Em) and RFP (561 nm Ex; 595/30 m Em) with a Leica Plan APO 20/0.7 objective using a Coolsnap HQ^2 (Photometrics, Tucson, AZ) controlled by MetaMorph version 7.7.8.0 (Molecular Devices, Sunnyvale, CA) software. The images were further processed in MetaMorph.

3. Results and discussion

3.1. Spatiotemporal simulation of interaction-specific microbial assembly

Using the agent-based model described in the previous section, we simulated the dynamic growth of two interacting microorgan-

isms in a two-dimensional space of 100 μm × 100 μm size, over which all parameters were assumed to be invariant. The datasets generated under this configuration were used as a main source for training deep learning networks (Section 3.2). This size range was selected because at the scale of 100 μm, microbial interactions are known to play a key role as a primary driver of population structure and dynamics [5].

Previous studies have shown that cooperative organisms develop stronger intermixed spatial patterns while competitive relationships lead to spatial segregation [25,27]. Consistent with these findings, we also obtained distinct spatial patterns for three representative interaction types (competition, cooperation, and exploitation) (Supplementary Fig. S3), the intermixing levels of which were in the following order: cooperation > exploitation > competition (Supplementary Fig. S4). These results suggest that specific spatial patterns can be considered as ecological signatures of microbial interactions, although we recognize they may also be affected to a degree by other parameters (e.g., basal growth rates) that are not directly associated with interactions. Not surprisingly, spatial patterns become more interaction-specific as the population density increased with time, as indicated by the fact that the differences in the intermixing levels among different interaction types were initially smaller but became progressively larger (Supplementary Fig. S4).

3.2. Microbial assembly governed by spatially homogeneous parameters

We evaluated the performance of deep learning models in inferring the interactions of two co-evolving organisms under a previously described simple configuration. For this purpose, we iteratively performed 5000 dynamic simulations using the agent-based model. In each simulation, we randomly determined the parameter set and collected multiple *in silico* images of two organisms with time such that the total population density (in collected images) ranged from 10 to 70%.

The simulation results in the previous section indicated that spatial images with higher population densities will be more informative than those with lower densities and that deep learning networks could be best developed using the images with the maximum highest available density. This gave rise to the following fundamental questions in implementing deep learning for actual applications: 1) "What is a threshold of population density, above which a reasonably good performance of deep learning networks is ensured?"; 2) "Will the deep learning networks trained using high density data be generalizable to infer interactions from lower density data?", and 3) "Alternatively, what is the best training strategy for handling data with varying levels of population densities?". These questions are systematically examined in the following sections.

3.2.1. Accurate prediction of interaction coefficients by deep learning networks trained by snapshot images

We illustrate the procedures of developing and evaluating deep learning networks when they are trained and tested with one-time snapshot image data with a similar population size (i.e., 70%). Developing deep learning networks requires the predetermination of hyperparameters (i.e., parameters to determine model architectures such as the topology and size of neural networks and other options associated with the selection of optimization algorithms and regularization methods). Depending on the choice of network architectures and other hyperparameters, we considered 28 cases (i.e., eight CNN and six ResNet models optimized using two alternative optimization algorithms: mini-batch GD vs. Adam). Following a typical practice, we iteratively split 5000 datasets into 4000 (80%) for training and 1000 (20%) for testing. Among all plain CNN models, CNN#6 showed the best performance regardless of the choice of optimizers (Supplementary Tables S3 and S4). Interestingly, the CNN#6 features are relatively less deep but have a wider network structure (i.e., with 16, 32, and 64 channels) compared to other models. This is understandable considering that plain CNNs with increased depth tend to show higher training errors due to the difficulty in finding optimal parameters [33]. ResNet models had smaller validation errors than CNN models on average (Supplementary Table S5). Overall, the performance of deep learning networks was shown to be fairly good: the average R^2 values of top 5 plain CNNs and top 3 ResNets for test datasets were 0.83 and 0.84, respectively (Table 1).

Table 1
Test performance of the top 5 in plain CNN configurations and the top 3 in ResNet configurations.

	Training Dataset Avg. MSE	Validation Dataset Avg. MSE	Test Dataset	
			Avg. MSE	Avg. R2
Plain CNN (Top 5)	3.066×10^{-4}	0.057	0.059	0.828
ResNet (Top 3)	2.269×10^{-4}	0.051	0.053	0.844

3.2.2. Enhanced performance by incorporating temporally evolving spatial images

In the previous section, we trained and tested deep learning networks against datasets with the same size of populations, but in reality, the population sizes may vary among images. This means that the population size in new images from which one attempts to extract interactions may not match with the ones that were used for training deep learning networks. Therefore, we examined to what extent deep learning networks trained with a specific level of population size (i.e., 70%) can be generalizable to different population sizes, ranging from 10 to 70%. As shown in Fig. 2(a), the performance of deep learning networks (as measured in terms of average RMSE and R^2 scores) was the highest when applied to the images with population densities of 70% and became progressively worse as the density level decreased. Notably, R^2 values became negative for the images with ≤30% densities, implying that deep learning networks performed worse than the null model. This result indicates that deep learning networks trained with a high-density data may be extended to the analysis of lower density data to a certain level but cannot be generalizable beyond that.

To expand the range of population densities that can be analyzed for inference, we tested two approaches. First, we independently trained deep learning models for each of the density levels between 10 and 70% and applied the resulting density-specific models only to the datasets with the corresponding same size of population. In comparison to the previous case above, the use of density-specific models led to improved inference (Supplementary Fig. S5). While the performance was still poor for population densities <20%, this may be ascribed to an intrinsic limitation in handling sparse data because such low population data will not provide sufficient spatial contexts to infer interaction coefficients. Despite improved performance, the use of density-specific models gave rise to an increased computational burden and inconvenience. In the example here, we had to develop seven individual deep learning networks (i.e., for population densities of 10~70%), the number of which increases if the population densities in new images do not match with any of the previously considered levels. One may handle this case by devising an additional heuristic – for instance, if the density level in a new image was 45%, one could average out the inference results obtained from 40% and 50%-density models. It should be realized, however, that this requires

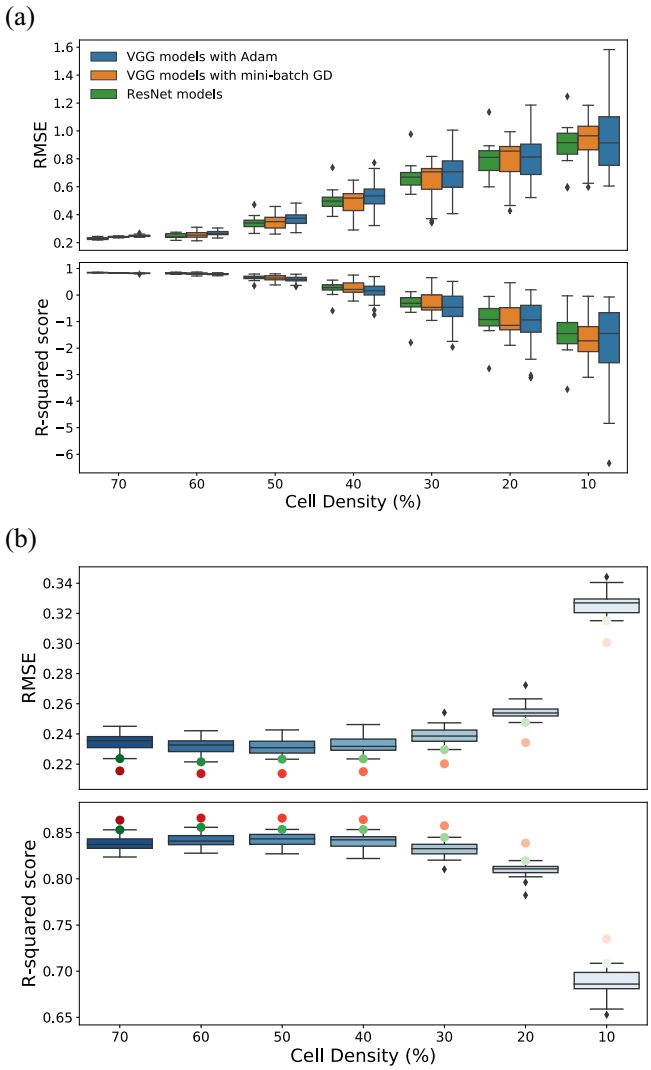

Fig. 2. Effect of the cell density on the prediction performance of deep learning models: (a) test performance of the best-performing models (chosen through rigorous hyperparameter optimization) for different cell density levels when trained using the images with the 70% cell density; (b) performance improvement of the ResNet models when trained using the images with the entire range of cell densities. In both (a) and (b), box plots describe the performance results of the best models and the red dots represent the performance results of the ensemble model in each cell density level. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

additional efforts to optimize these heuristics, which is often not straightforward.

In the second approach, we developed deep learning networks (using the top 3 ResNet models) by expanding the training dataset to include the entire range of population sizes from 10 to 70%. Interestingly, this new model (that was trained with a single set of datasets collecting all images across all population densities even though the size is seven times larger than the previous cases) achieved a comparable performance to the case with density-specific models (Fig. 2(b)). The performance was particularly high for the 40 ~ 60% range of density levels, while being poor again when the density was below 20%. A potential reason for this improved performance is that temporal evolution of spatial patterns allows deep learning networks to extract additional information on microbial interactions, which could not be deducible from stagnant snapshot images. We therefore further improved the performance by employing a randomization-based ensemble approach [46]. That is, we aggregated the outputs from the top three ResNet models and took their averages as the final estimations. The ensemble approach produced a better predictive performance than individual models, even the best one in each cell density (Fig. 2(b)). Due to its more robust and accurate predictions we applied this approach in combination with the ensemble estimation in all subsequent cases.

3.3. Microbial assembly governed by spatially heterogeneous parameters

We extended our analysis to a larger-size domain where the previous assumption that interaction and growth parameters are invariant in space may not hold. For this purpose, we considered a 450 μm × 450 μm size domain and additionally performed 50 dynamic growth simulations by accounting for spatial heterogeneity of the parameter set in the agent-based model. We divided the computational domain into five subdomains such that the growth and interaction dynamics in each subdomain is governed by independent parameter sets; in each simulation, we randomly assigned five sets of model parameters (see Section 2.1.3 and Supplementary Fig. S1). This configuration sets up simple proof-of-concept simulations for the extended testing of deep learning models. The heterogeneous agent-based model properly generated composite spatial patterns as configured using distinct parameter sets in subdomains (Fig. 3(a)). Depending on how the parameters (randomly determined) were partitioned, both population densities and spatial patterns across subdomains may or may not be distinctive.

Our goal was to infer spatial variation of interactions using the previously developed top three ResNets without any additional training. This requires a special approach because deep learning networks were trained using the data generated under a completely different configuration. Thus, we extracted local interactions from a predefined 100 μm × 100 μm-sized window that slides through the space to scan the entire domain (see Section 2.2.5). With the step size of 25 μm, the 450 μm × 450 μm domain is filled with 225 (=15 × 15) smaller windows (e.g., Fig. 3(b)). Due to overlapping among windows, each grid is assigned with multiple values of interaction parameters (inferred from deep learning networks). We determined the grid-specific interaction parameters by taking an average of a set of values obtained from neighbor windows.

The sliding window approach successfully inferred interaction parameters from composite spatial patterns (averaged R^2: 0.79) (Supplementary Fig. S6). The ability to predict the spatial variation of interaction parameters was remarkable considering that deep learning networks were never previously fed with those data. Comparison of interaction parameter distributions demonstrates the performance of our approach in a more dramatic way. The example shown in Fig. 4 includes four different types of interactions: amensalism (subdomains 1 and 5), parasitism (subdomain 2), commensalism (subdomain 3), and competition (subdomain 5), all of which were correctly inferred from the sliding window method. Our approach also enabled visually identifying the borderlines across subdomains that are governed by distinct sets of parameters because they are characterized by relatively higher variations of predicted interaction coefficients as shown in Fig. 4. The performance of deep learning networks was consistently verified through all other examples (Supplementary Fig. S7).

3.4. The shifts in microbial interactions inferred from real microscopic images

To demonstrate the applicability of the deep learning method for real microscopy images, we co-cultured GFP tagged *P. fluo-*

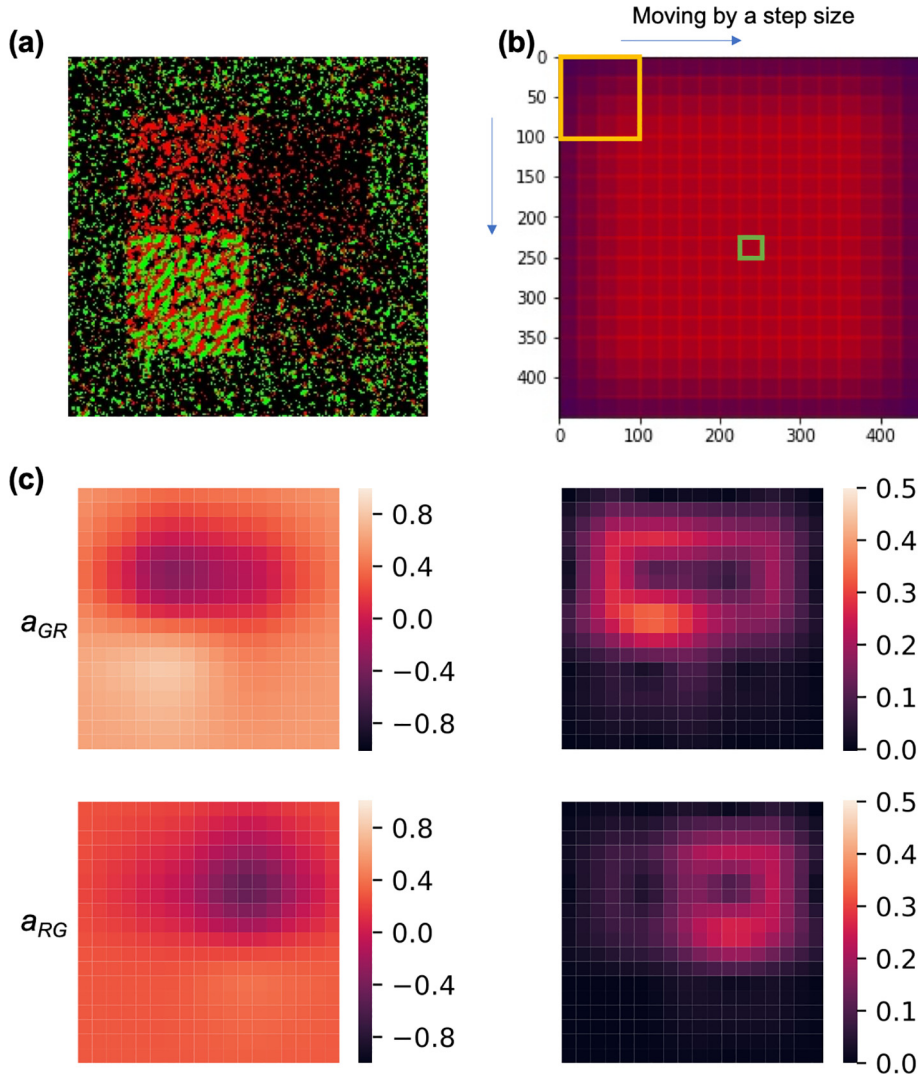

Fig. 3. Illustration of (a) an example of the assembly patterns governed by a composite set of 5 different interaction parameters and (b) the sliding windows of image size = 450 (n_i = 100, n_s=25). There are 225 sliding windows in 15 rows and 15 columns. (c) shows the heatmaps of true interaction distributions when the step size is 25. The orange box and green box in (a) represent the sliding window ($n_i \times n_i$) and the grid block ($n_s \times n_s$), respectively. (c) shows the heatmap of averaged interactions (left) and standard deviations of neighboring interactions (right). a_{GR} and a_{RB} represent the effect of R(red) on G(green) and the effect of G on R, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

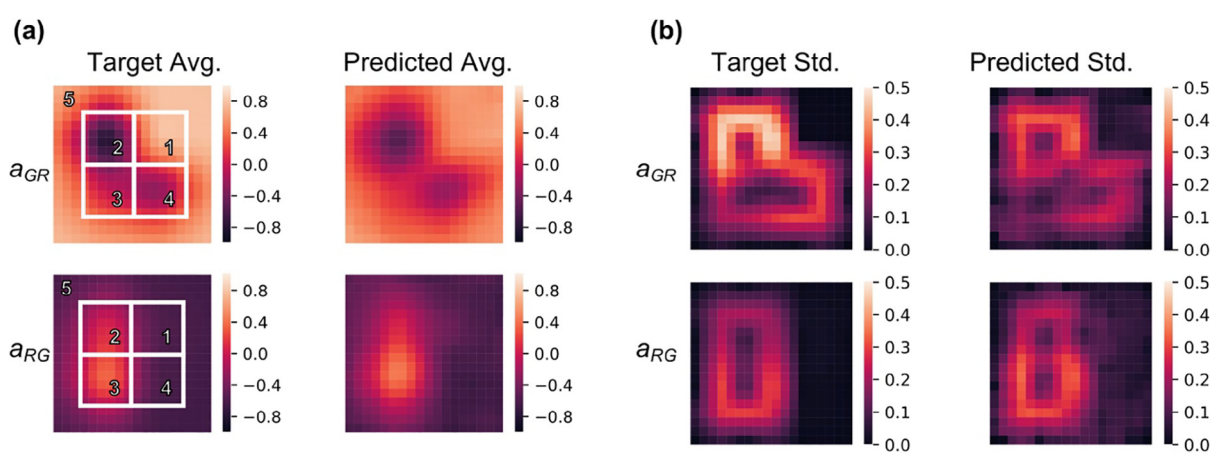

Fig. 4. An example of interaction heatmaps, systematically built by aggregating all prediction results from the sliding window estimation, which represented the interaction governing a locally assembled community at microscale (See 2.2.5). Each grid (an overlapping block) can represent (a) the averaged interactions of the neighboring windows and (b) their standard deviations. The original assembly pattern was shown in Supplementary Fig. S9 and the left panels of (a) show the boundaries of five subdomains governed by different interactions. This visualization facilitates intuitive inspection how similar the target and predicted interaction patterns are. The grid locations of the interaction heatmaps in (a) and (b) corresponds to the coordinate of the input image. Each grid in (a) and (b) respectively shows the average values and standard deviations of interaction coefficients of the neighboring sliding windows that overlie the corresponding grid.

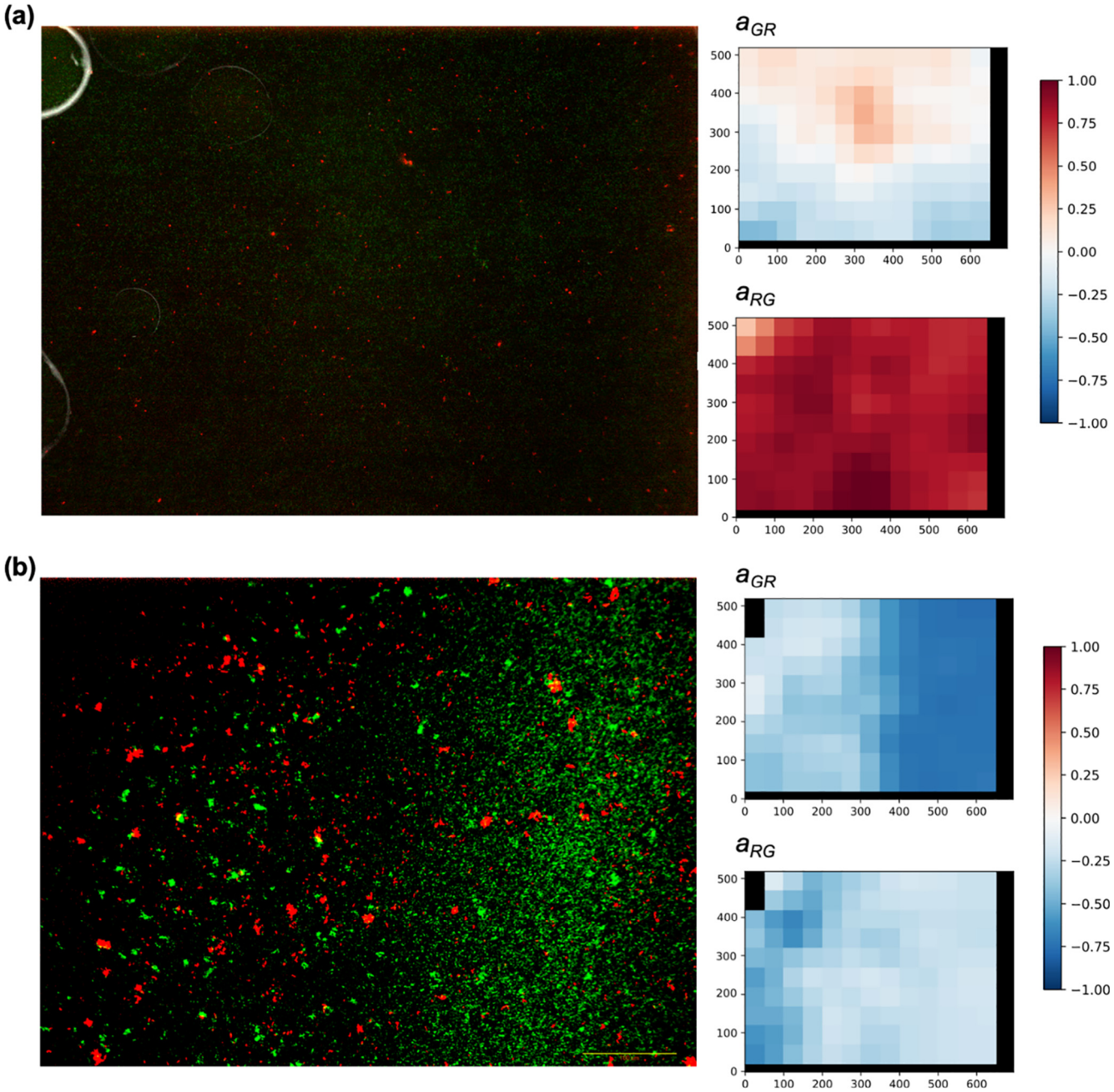

Fig. 5. Interaction heatmaps automatically predicted from species level assembly patterns (i.e., real microscopy images) influenced by various trophic dynamics. These images were taken from the co-culture experiments of *P. fluorescens* (Green fluorescent) and *E. coli* △chiA (Red fluorescent) in (a) chitin beads and (b) chitin monomer N-acetyl glucosamine (See 2.3). (a) Indicates that, when cultured with chitin, *P. fluorescens* positively affects *E. coli* △chiA growth since the heatmap in the bottom right panel shows a reddish color that stands for the strong positive interaction of the *P. fluorescens* on *E. coli*. In contrast, the top right panel of (a) shows *E. coli* has no significant effect on *P. fluorescens* growth (interaction coefficients reside between −0.2 to +0.2). In (b), the negative effects of *P. fluorescens* (a_{RG}) became a bit stronger than its counterpart in the left side. However, as it went to the right, the negative effects of *E. coli* (a_{GR}) did get much stronger than the other. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

rescens and RFP tagged *E. coli* △chiA (chitinase gene chiA null) with either a chitin polymer or N-acetyl glucosamine and captured images after 36 h of incubation by fluorescent confocal microscopy (see Section 2.3). These substrates were chosen because chitin polymers must first be enzymatically hydrolyzed to oligomers and monomers before the sugar can be utilized for energy and growth and is thus ideally suited for studying metabolic co-dependencies between populations. Because *E. coli* △chiA is incapable of degrading chitin, *E. coli* will depend on the chitinolytic activity of *P. fluorescens* to acquire chitin hydrolysis products (e.g. monomers and oligomers of N-acetyl glucosamine) in the microcosms supplied with chitin as the sole supplied carbon source. Therefore, resource accessibility would be determined by the species spatial organization in a given environment (Supplementary Fig. S8(a)). On the other hand, both species will compete for resources in the microcosms fed with the readily accessible chitin monomer, N-acetyl glucosamine (Supplementary Fig. S8(b)).

Using the sliding window method described in the previous section, we were able to extract the local variation of interaction parameters from real images (because it is difficult to assume that

spatial distributions of nutrient concentrations and population densities are perfectly uniform), while the level of their variation was insignificant in our experimental settings (Fig. 5). Overall predictions were consistent with our expectations, i.e., the deep learning networks predicted a degrader-cheater relationship between *P. fluorescens* and *E. coli* △chiA when grown on chitin polymers (Fig. 5(a)), but a competitive relationship when grown on their hydrolysis products (Fig. 5(b)). The latter case was clearly indicated by negative values of interaction coefficients, both a_{GR} (the influence of *E. coli* on *P. fluorescens*) (right top panel of Fig. 5(b)) and a_{RG} (the influence of *P. fluorescens* on *E. coli*) (right bottom panel of Fig. 5(b). In the case of growth on chitin polymers, the influence of *E. coli* on *P. fluorescens* was predicted to qualitatively vary in space (i.e., positive in the upper area; negative in the bottom area) (right top panel of Fig. 5(a)), while the effect of *P. fluorescens* on *E. coli* was predicted to be positive throughout the entire domain (right bottom panel of Fig. 5(a)). While local, the prediction of the positive influence of *E. coli* on *P. fluorescens* (a_{GR}) in the upper area mismatched our expectation. Unlike Fig. 5(b), the image in Fig. 5(a) shows that *P. fluorescens* is predominant while the presence of *E. coli* is minor. Thus, two possibilities are indicated: 1) our deep learning network was not sufficiently trained against those situations, or 2) images (dominated by one species) pose an intrinsic challenge in accurately predicting the influence of the minor species on its partners. Despite this uncertainty, we stress that our deep learning model demonstrated unprecedented performance in inferring microbial interactions with acceptable accuracy, while predicting specific interspecies interactions from snapshot images is far from intuitive.

4. Conclusions

Here, we proposed supervised deep learning as a new network inference method to predict interspecies interactions from non-conventional data, i.e., spatial patterns of microorganisms. Due to the unsuitability of real image data for training deep learning models, we employed agent-based models for data generation. Agent-based modeling is commonly used to simulate the self-organization of interacting microorganisms in space and time as governed by specific interaction and growth rules. Our use of the agent-based model was exactly the opposite, i.e., we used their simulation results to train deep learning network models for computational inference of interspecies interactions (i.e., for reverse modeling). Through case studies considered in this work, we demonstrated that: 1) agent-based modeling can serve as an effective source of *in silico* images (for training deep learning network models); 2) in combination with the sliding windows technique, the deep learning network trained on spatially homogeneous environments can extract local variances of interaction parameters in heterogeneous environments; and 3) in the application to real images, the resulting deep learning network can correctly differentiate context-dependent microbial interactions.

Challenges could arise in extending the developed computational method to predict interspecies interactions in complex, natural microbial communities that contain diverse microbial species including microeukaryotes as well as bacteria, which are mostly unculturable. For example, the predictive power of deep neural networks is dependent on the accuracy of *in silico* data, which is an intrinsic issue with all machine learning methods that use model-generated data for training. Currently, reliable simulation of spatial patterns formed by multiple interacting species remains challenging, often due to the lack of fundamental knowledge of growth characteristics of individual species in environmental communities. However, this gap is expected to be minimized through rapid advancement in simulation methods [47,48], as well as cultivation techniques [49,50] that enable building structurally tractable synthetic consortia derived from complex communities to provide molecular omics data that can inform agent-based modeling for more accurate simulations.

Our work provides a generalizable pipeline for developing deep learning models from image data, which can serve as a basic platform for further development by combining with strategic data generation through robust design of experiments and transfer learning and fine-tuning using pre-trained models [51] (e.g., to overcome lack of experimental image data [22]). Co-design of experimental and modeling studies following the procedures proposed in this work is our key contribution that can facilitate to reveal key but previously unknown interaction mechanisms in complex microbial communities that have been underexplored to date.

Author statement

J.-Y.L., N.C.S, and H.-S.S. designed the study and wrote the manuscript. J.-Y.L. trained and tested deep learning networks. H.-S.S. performed agent-based model simulations to generate *in silico* images. N.C.S. performed experiments to collect microscopic images. R.G.E. transformed *P. fluorescens*, and C.R.A., K.S.H., and J.K.J. contributed to the design of experiments. All authors read and approved the final manuscript.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was supported by the Department of Energy Office of Biological and Environmental Research (BER) and is a contribution of the Scientific Focus Area "Phenotypic response of the soil microbiome to environmental perturbations" at Pacific Northwest National Laboratory (PNNL). A portion of the research described in this paper was also performed in the Environmental Molecular Sciences Laboratory (EMSL). PNNL is operated for the DOE by Battelle Memorial Institute under Contract DE-AC05-76RLO1830.

Research data for this article

All datasets and numerical codes used in this study can be freely downloaded from https://github.com/hyunseobsong/csbjDeepLearning.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.csbj.2020.05.023.

References

[1] Bradford MA, McCulley RL, Crowther TW, Oldfield EE, Wood SA, Fierer N. Cross-biome patterns in soil microbial respiration predictable from evolutionary theory on thermal adaptation. Nat Ecol Evol 2019. https://doi.org/10.1038/s41559-018-0771-4.
[2] Schlesinger WH, Andrews JA. Soil respiration and the global carbon cycle. Biogeochemistry 2000. https://doi.org/10.1023/A:1006247623877.
[3] Cai P, Sun X, Wu Y, Gao C, Mortimer M, Holden PA, et al. Soil biofilms: microbial interactions, challenges, and advanced techniques for ex-situ characterization. Soil Ecol Lett 2019. https://doi.org/10.1007/s42832-019-0017-7.

[4] Vos M, Wolf AB, Jennings SJ, Kowalchuk GA. Micro-scale determinants of bacterial diversity in soil. FEMS Microbiol Rev 2013. https://doi.org/10.1111/1574-6976.12023.
[5] Cordero OX, Datta MS. Microbial interactions and community assembly at microscales. Curr Opin Microbiol 2016. https://doi.org/10.1016/j.mib.2016.03.015.
[6] Liu W, Jacquiod S, Brejnrod A, Russel J, Burmølle M, Sørensen SJ. Deciphering links between bacterial interactions and spatial organization in multispecies biofilms. ISME J 2019. https://doi.org/10.1038/s41396-019-0494-9.
[7] Valm AM, Mark Welch JL, Rieken CW, Hasegawa Y, Sogin ML, Oldenbourg R, et al. Systems-level analysis of microbial community organization through combinatorial labeling and spectral imaging. Proc Natl Acad Sci U S A 2011. https://doi.org/10.1073/pnas.1101134108.
[8] Massalha H, Korenblum E, Malitsky S, Shapiro OH, Aharoni A. Live imaging of root-bacteria interactions in a microfluidics setup. Proc Natl Acad Sci U S A 2017. https://doi.org/10.1073/pnas.1618584114.
[9] Lambert BS, Raina JB, Fernandez VI, Rinke C, Siboni N, Rubino F, et al. A microfluidics-based in situ chemotaxis assay to study the behaviour of aquatic microbial communities. Nat Microbiol 2017. https://doi.org/10.1038/s41564-017-0010-9.
[10] Raynaud X, Nunan N. Spatial ecology of bacteria at the microscale in soil. PLoS ONE 2014. https://doi.org/10.1371/journal.pone.0087217.
[11] Song HS, Cannon WR, Beliaev AS, Konopka A. Mathematical modeling of microbial community dynamics: A methodological review. Processes 2014. https://doi.org/10.3390/pr2040711.
[12] Faust K, Raes J. Microbial interactions: From networks to models. Nat Rev Microbiol 2012. https://doi.org/10.1038/nrmicro2832.
[13] Song HS, Lee JY, Haruta S, Nelson WC, Lee DY, Lindemann SR, et al. Minimal interspecies interaction adjustment (MIIA): Inference of neighbor-dependent interactions in microbial communities. Front Microbiol 2019. https://doi.org/10.3389/fmicb.2019.01264.
[14] Lee J-Y, Haruta S, Kato S, Bernstein HC, Lindemann S, Lee D-Y, et al. Prediction of Neighbor-dependent Microbial Interactions from Limited Population Data. Front Microbiol 2020.
[15] Goodfellow I, Bengio Y, Courville A. Deep Learning – whole book. Nature 2016. https://doi.org/10.1038/nmeth.3707.
[16] Bishop CM. Pattern Recognition and Machine Learning. New York: Springer; 2016.
[17] Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, et al. Human-level control through deep reinforcement learning. Nature 2015. https://doi.org/10.1038/nature14236.
[18] Lecun Y, Bengio Y, Hinton G. Deep learning. Nature 2015. https://doi.org/10.1038/nature14539.
[19] Zou J, Huss M, Abid A, Mohammadi P, Torkamani A, Telenti A. A primer on deep learning in genomics. Nat Genet 2019. https://doi.org/10.1038/s41588-018-0295-5.
[20] Esteva A, Robicquet A, Ramsundar B, Kuleshov V, DePristo M, Chou K, et al. A guide to deep learning in healthcare. Nat Med 2019. https://doi.org/10.1038/s41591-018-0316-z.
[21] Reichstein M, Camps-Valls G, Stevens B, Jung M, Denzler J, Carvalhais N, et al. Deep learning and process understanding for data-driven Earth system science. Nature 2019. https://doi.org/10.1038/s41586-019-0912-1.
[22] Chapelais-Baron M, Goubet I, Péteri R, Pereira M de F, Mignot T, Jabveneau A, et al. Colony analysis and deep learning uncover 5-hydroxyindole as an inhibitor of gliding motility and iridescence in Cellulophaga lytica. Microbiology 2018;164:308–21. https://doi.org/10.1099/mic.0.000617.
[23] Hellweger FL, Clegg RJ, Clark JR, Plugge CM, Kreft JU. Advancing microbial sciences by individual-based modelling. Nat Rev Microbiol 2016. https://doi.org/10.1038/nrmicro.2016.62.
[24] Kim M, Or D. Individual-based model of microbial life on hydrated rough soil surfaces. PLoS ONE 2016. https://doi.org/10.1371/journal.pone.0147394.
[25] Tecon R, Ebrahimi A, Kleyer H, Levi SE, Or D. Cell-to-cell bacterial interactions promoted by drier conditions on soil surfaces. Proc Natl Acad Sci U S A 2018. https://doi.org/10.1073/pnas.1808274115.
[26] Deutsch A, Dormann S. Cellular automaton modeling of biological pattern formation: characterization, applications, and analysis. Model Simul Sci Eng Technol 2005.
[27] Momeni B, Brileya KA, Fields MW, Shou W. Strong inter-population cooperation leads to partner intermixing in microbial communities. Elife 2013. https://doi.org/10.7554/eLife.00230.
[28] Gillespie DT. Exact stochastic simulation of coupled chemical reactions. J Phys Chem 1977. https://doi.org/10.1021/j100540a008.
[29] Gillespie DT. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. J Comput Phys 1976. https://doi.org/10.1016/0021-9991(76)90041-3.
[30] Higham DJ. Modeling and simulating chemical reactions. SIAM Rev 2008. https://doi.org/10.1137/060666457.
[31] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. Conf Learn Represent ICLR 2015 Conf Track Proc 2015.
[32] Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 2012.
[33] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2016. https://doi.org/10.1109/CVPR.2016.90.
[34] He K, Zhang X, Ren S, Sun J. Identity mappings in deep residual networks. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics) 2016. https://doi.org/10.1007/978-3-319-46493-0_38.
[35] Nair V, Hinton GE. Rectified linear units improve Restricted Boltzmann machines. ICML 2010 – Proceedings, 27th Int. Conf. Mach. Learn., 2010.
[36] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. 32nd Int Conf Mach Learn ICML 2015 2015.
[37] Kohavi R. A study of cross-validation and bootstrap for accuracy estimation and model selection. Int Jt Conf Artif Intell 1995.
[38] Kingma DP, Adam Ba JL. A method for stochastic optimization. Conf Learn Represent ICLR 2015 – Conf Track Proc 2015.
[39] Ruder S. An overview of gradient descent optimization algorithms 2016.
[40] Prechelt L. Automatic early stopping using cross validation: quantifying the criteria. Neural Networks 1998. https://doi.org/10.1016/S0893-6080(98)00010-0.
[41] Papageorgiou C, Poggio T. Trainable system for object detection. Int J Comput Vis 2000. https://doi.org/10.1023/A:1008162616689.
[42] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2001. https://doi.org/10.1109/cvpr.2001.990517.
[43] Baba T, Ara T, Hasegawa M, Takai Y, Okumura Y, Baba M, et al. Construction of Escherichia coli K-12 in-frame, single-gene knockout mutants: the Keio collection. Mol Syst Biol 2006. https://doi.org/10.1038/msb4100050.
[44] Schlechter RO, Jun H, Bernach M, Oso S, Boyd E, Muñoz-Lintz DA, et al. Chromatic bacteria – A broad host-range plasmid and chromosomal insertion toolbox for fluorescent protein expression in bacteria. Front Microbiol 2018. https://doi.org/10.3389/fmicb.2018.03052.
[45] Neidhardt FC, Bloch PL, Smith DF. Culture medium for enterobacteria. J Bacteriol 1974.
[46] Breiman L. Random forests. Mach Learn 2001;45:5–32.
[47] Wilmoth JL, Doak PW, Timm A, Halsted M, Anderson JD, Ginovart M, et al. A microfluidics and agent-based modeling framework for investigating spatial organization in bacterial colonies: the case of Pseudomonas aeruginosa and H1-Type VI secretion interactions. Front Microbiol 2018. https://doi.org/10.3389/fmicb.2018.00033.
[48] Ginovart Gisbert M. How a multi-agent programmable modelling environment like NetLogo can help to deal with communities or assemblages of bacteria on surfaces?. Explor Microorg Recent Adv Appl Microbiol 2018:256–60.
[49] Zegeye EK, Brislawn CJ, Farris Y, Fansler SJ, Hofmockel KS, Jansson JK, et al. Selection, Succession, and Stabilization of Soil Microbial Consortia. MSystems 2019;4. https://doi.org/10.1128/mSystems.00055-19.
[50] Khan N, Maezato Y, McClure RS, Brislawn CJ, Mobberley JM, Isern N, et al. Phenotypic responses to interspecies competition and commensalism in a naturally-derived microbial co-culture. Sci Rep 2018;8:297. https://doi.org/10.1038/s41598-017-18630-1.
[51] Yosinski J, Clune J, Bengio Y, Lipson H. How transferable are features in deep neural networks? Adv Neural Inf Process Syst 2014:3320-8. https://doi.org/https://dl.acm.org/doi/10.5555/2969033.2969197.