LI Detector: a framework for sensitive colony-based screens regardless of the distribution of fitness effects

Saurin Bipin Parikh, Nelson Castilho Coelho, and Anne-Ruxandra Carvunis

Department of Computational and System Biology, University of Pittsburgh School of Medicine, Pittsburgh, Pennsylvania, 15213

Pittsburgh Center for Evolutionary Biology and Medicine, University of Pittsburgh School of Medicine, Pittsburgh, Pennsylvania, 15123

Keywords: adaptive, bioinformatics, fitness, high throughput, phenomics

ABSTRACT

Microbial growth characteristics have long been used to investigate questions of biology. Colony-based high-throughput screens enable the parallel analysis of thousands of individual strains by measuring microbial colony growth on agar plates as a proxy for fitness. In these assays, colony growth is affected by spatial biases, including uneven nutrient distribution, agar surface irregularities, batch effect, and other technical factors that cannot be predicted or controlled for *a priori*. To correct for these spatial biases, several analytical methods have been developed which rely on the following assumptions: i) that fitness effects are normally distributed, and ii) that most genetic perturbations lead to minor changes in fitness. These assumptions, although reasonable for many genome-wide screens, are not always warranted and intrinsically reduce the ability to detect small fitness effects. Beneficial fitness effects, in particular, are notoriously difficult to detect using these assumptions. Here, we developed the linear interpolation-based detector (LI Detector) framework to enable sensitive colony-based screening without making prior assumptions about the underlying distribution of fitness effects. The LI Detector introduces a grid of reference colonies on each agar plate to predict and correct spatial biases. It assigns a relative fitness to all colonies as a measure of their size compared to the neighboring reference colonies. We show that the LI Detector can identify both increases and decreases in fitness effects as low as 5% with more than 95% sensitivity and specificity. LI Detector can be utilized to develop and refine gene-gene and gene-environment interaction networks of all colony-forming organisms, including yeast, by increasing the range of colony size differences that can be reliably detected.

INTRODUCTION

Colony-based high-throughput screens (CBHTS) of microbes are increasingly used for basic scientific, biomedical, and industrial research[1-4]. These screens involve growing manually or robotically placed grids of microbial colonies on solid agar plates that are used to record growth using imagery. The images are analyzed to generate a quantitative output of colony size that is used as a proxy for the fitness of the organism. The wide availability of such methodologies, combined with the growing number of artificial gene constructs for microbial model organisms such as *Saccharomyces cerevisiae*, have provided a large-scale controlled approach to experimentally determine the effects of genetic perturbations on the fitness of an organism. CBHTS have been previously used to explore genetic interactions[5, 6], protein-protein interactions[7-9], chemical-genetic interactions[10, 11], protein localization[12], and microbial pathogenicity[13].

The benefit of conducting CBHTS, especially for an organism like *S. cerevisiae*, is that its entire genome can be investigated within a single experiment. These experiments can fast track discovery due to the scale at which they are conducted. However, the scale also amplifies the spatial bias in colony size that is introduced during the experimental process (**Supplementary Fig. S1**). These biases result in systematic colony size differences that arise due to technical reasons as opposed to the perturbations being investigated in the study. The extent and proportion of these spatial biases cannot be predicted a priori, making their identification and correction a substantial computational challenge. Spatial biases include artifacts related to edge effects[14], local competition[15, 16], batch effects[17], source plate memory[14], light artifacts[18, 19], agar surface and nutrient heterogeneity[14-16], and even environmental factors such as humidity[20]. A variety of tools like the HT colony grid analyzer[21], Colonyzer[18], ScreenMill[19], ScanLag[22], SGATools[15], Balony[16], MATLAB Colony Analyzer Toolkit[14]

and Scan-o-matic[23] exist that implement normalization algorithms to correct for spatial biases (**Supplementary Table S1**). Most of these tools rely on the assumption that the colony sizes in an experiment are normally distributed, and that genetic manipulations only rarely cause fitness deviation from wildtype. Although these assumptions are often reasonable for some cases like genome-scale Synthetic Genetic Arrays (SGA), they work less well in other contexts.

The first assumption is violated in small-scale experiments with fewer plates and mutants[17, 24]. Even large-scale experiments done in conditions producing polarizing fitness effects would not follow this assumption. Lastly, the ability to reliably observe beneficial fitness effects is also limited because there is usually an abundance of neutral and deleterious genetic perturbations leading to a left skew in the fitness distribution[25-27]. The second assumption limits the use of colony-based screens for observing small changes in fitness as these are often ignored or corrected against. Both these assumptions reduce the applicability of colony-based screens to a wide variety of scientific interrogations.

The linear interpolation-based detector (LI Detector or LID) framework for conducting CBHTS is designed to avoid making any *a priori* assumptions of the underlying fitness distribution. This two-part, experimental, and analytical method utilizes a reference colony grid on every plate of the experiment to predict and correct for spatial biases. The reference grid is an isogenic population of colonies that are evenly distributed over the agar surface to act as an internal control[23]. The spatial bias correction is done by assuming that this reference colony grid is representative of the typical growth pattern that is to be expected within the given spatial context. Our results show that the LI Detector's reference colony based linear interpolant can successfully remove spatial bias. It has very high specificity and sensitivity and can identify fitness effects as low as 5% with more than 95% sensitivity. LI Detector is also equally sensitive in finding fitness increases and

85 decreases. Its sensitivity is directly related to the number of references per plate and replicates per strain.

MATERIALS AND METHODS

90 **Validation experiment using an isogenic population**

A method validation experiment was conducted using an isogenic population that was mocked as either references or mutants.

Yeast strain, medium and robotic equipment

A previously characterized strain, FY4[28], was used to conduct experiments in YPD medium (1%
95 w/v yeast extract, 2% w/v peptone, 2% w/v dextrose, and 2% agar in the case of solid medium). A single colony of FY4 selected from a streak out was used to inoculate liquid YPD medium and grown overnight at 30°C. This culture was used to create four 384-well glycerol stocks with wells containing 18μL of 50% glycerol and 42μL of culture media. Two to five wells in each stock were left empty to create gaps in the colony grid when pinned on solid medium. The stocks were stored
100 at -80°C before use. The bench-top RoToR HDA robotic plate handler (Singer Instruments Co Ltd, Roadwater, UK) was used for plate to plate cell transfer.

Pin-copy-upscale

The LI Detector experimental pipeline follows a pin-copy-upscale protocol when starting from frozen glycerol stocks, where the copy-upscale steps are repeated until the desired colony density
105 is reached (**Fig. 1a**). The four glycerol stock plates were pinned at 384-density to working copy agar plates. This process was performed using the RoToR HDA robot with default settings (Supplementary Table S2). One working copy was chosen at random to represent the mock reference population. Every colony on the other three working copies was mocked as a unique mutant strain. The working copies were incubated at 30°C for 60 hours to reach saturation. These
110 were then copied 1-to-1 to make transition plates (#1) using default RoToR HDA settings (Supplementary Table S2) and incubated at 30°C for 48 hours. Distinct combinations of the four transition plates (#1) were then condensed to make four 1536-density upscale plates (#1) using default RoToR HDA settings (Supplementary Table S2). The distinct combinations make sure

that colony grids from each plate occupy different positions on the four higher-density plates. At
115 this stage, each colony that was present on the working copy has four technical replicates, and 1/4th of all colonies on every plate are references. The upscale plates (#1) were incubated at 30°C for 30 hours, after which they were copied 1-to-1 to transition plates (#2) using custom RoToR HDA settings (Supplementary Table S2). The overshoot setting value at the target plate was increased to compensate for the unevenness of the agar surface and the smaller pin size of the
120 higher density pin pads. These plates were incubated at 30°C for 30 hours. Four upscale plates (#2) were then made by condensing the four transition plates (#2) in distinct combinations using default RoToR HDA settings (Supplementary Table S2). These were incubated at 30°C until they reached saturation. The upscale plates (#2) have 16 technical replicates for every colony that was present in the working copy, and 1/4th of all colonies are references. **Supplementary Figure S2**
125 provides a simplified visual representation of the plates at all these steps.

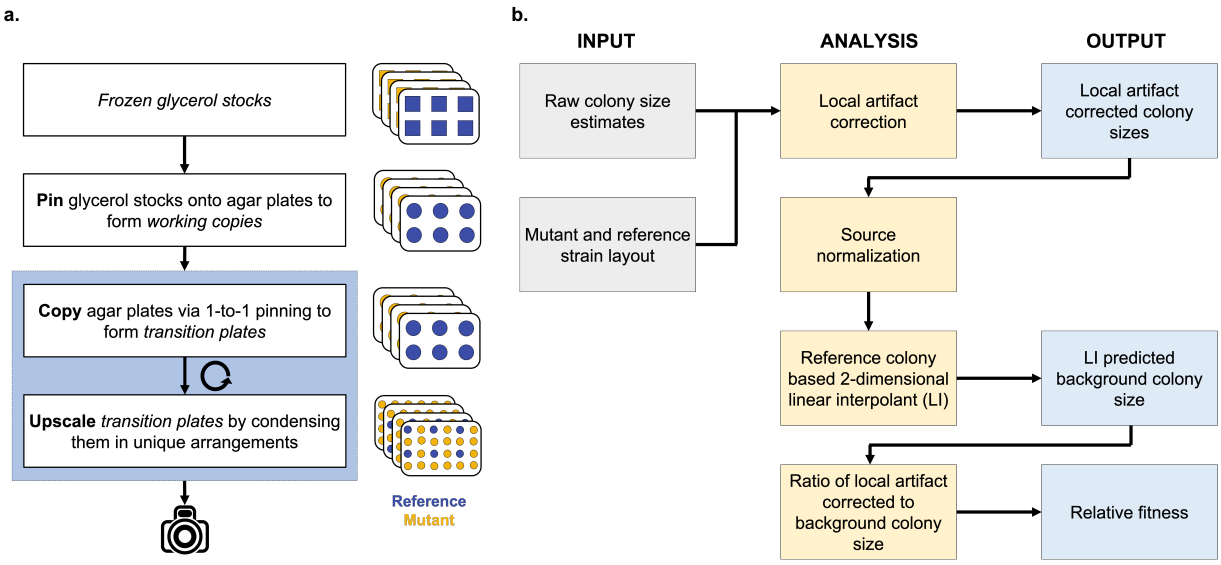

Figure 1. The LI Detector framework consists of an experimental and analytical pipeline. **a**. The flow chart represents the pin-copy-upscale experimental pipeline that goes from frozen glycerol stocks (top) to
130 imaging (bottom). Each box represents a pinning step that can be performed manually or robotically. The steps within the sky-blue highlighted portion can be repeated until the desired colony density is reached. A reference population is introduced on every plate during the first upscale step. This population is used by the analytical pipeline for spatial bias correction. Illustrations to the right of the flowchart is a simplified

representation of four experimental plates. Squares illustrate wells, circles are colonies, and colors represent different colony types. **b.** Workflow of the analysis pipeline where columns from left to right represent user inputs, analytical steps, and outputs. User inputs consist of raw colony size estimates, and strain layout of the plates. Analytical pipeline performs local artifact correction, source normalization, reference-based background colony size prediction using a 2-dimensional linear interpolation, and normalizing spatial bias by dividing the observed with the background colony sizes. The outputs include local artifact corrected colony sizes, reference predicted background colony sizes, and spatially corrected relative fitness.

Colony size estimation

A custom-made lightbox with an overhead camera mount was built to acquire high-resolution images using a commercially available SLR camera (18Mpixel Rebel T6, Canon USA Inc., Melville, NY, USA). The 6144-density final screen plates were imaged at 12 time points beginning right after pinning, till the colonies reach saturation, which was around 11 hours later. Saturation was determined as the point at which the colonies would run into each other if the plates are incubated for any longer. The images were analyzed in bulk using the "analyze_directory_of_images()" function of the MATLAB Colony Analyzer Toolkit (MCAT)[14] with default parameters (window size 9, threshold 1.25) to provide colony size estimations (https://github.com/sauriiiin/lidetector/blob/master/imageanalyzer.m).

Spatially cognizant colony size database

Every possible colony position across the different plates of the experiment was given a unique identifier called position id (pos). Each pos was linked to plate density, plate number, column number and row number, and stored in the position to coordinate table (pos2coor). A position to mutant name (pos2orf_name) table was used to store information on which colony position was occupied by what mutant. The colony size estimations, along with the pos2coor table, were used to store the colony sizes in a spatially cognizant manner. **Supplementary Figure S2** is a visual representation of the plate maps made using the pos2coor and pos2orf_name tables.

LI Detector analytical pipeline

165 The LID analytical pipeline (**Fig. 1b**, https://github.com/sauriiiin/lidetector/blob/master/lid.m) is designed to make fitness assessments using local reference colony information.

Border colony removal

Border colonies tend to grow larger because of increased access to nutrients. To remove this artifact, we reject colony size estimations from 1, 2, and 4 border rows and columns from 384,

170 1536, and 6144-density plates, respectively.

Local artifact correction (AC)

The first step of the analytical pipeline is local artifact correction, which is inspired by the 'competition correction' feature present in previous tools[15, 16]. An 'artifact score' is assigned to every colony on a plate as a ratio of its colony size compared to its present and past neighbors.

175 The present neighbors are a colony's eight immediate neighbor colonies, and the past neighbors are eight neighboring colonies that come from the same source plate. The artifact scores of the reference population is used to determine outliers using two median adjusted deviations. The outliers would be the colonies growing disproportionately big or small as compared to their neighboring colonies. Outliers that occur as a localized group of three or more neighbors of both

180 big and small colonies are considered for correction. The less abundant outlier in the group is expected to have driven the phenotype. For example, a single small or dead colony would increase the relative access to nutrients for all its neighbors, which would all then be expected to grow bigger than usual and vice-versa. Raw colony sizes of all immediate neighbors of the driver are median normalized using the median reference population colony size for the plate. User has

185 the option to skip this correction.

Source normalization (SN)

LI Detector uses a source-based computational deconstruction of high-density plates into their four sources to correct for the source related colony size differences that are introduced during the upscaling process (**Supplementary Fig. S3**). This correction is adopted from MCAT's

interleave filter[14]. Each source-deconstruct is individually normalized in the later steps, hence also making it necessary for the penultimate density plates to have a reference population grid. User has the option to skip this correction.

Reference-based normalization

A two-dimensional linear interpolant is applied to the reference population grid to predict background colony sizes on the entire colony grid. The background colony sizes represent the predicted reference colony growth on every position of the grid conditioned upon its spatial context. Relative fitness is assigned to colonies as the ratio of the observed colony size to the background colony size. Hence, giving a direct measure of how well a mutant is growing as compared to the reference while maintaining the spatial context.

Going from colony size to relative fitness using different modes of normalization

The LI Detector analytical pipeline is applied to colony size estimates to remove spatial bias and measure relative fitness as described above. The analytical pipeline is applied as is (LID), without local artifact correction (LID-AC), and without source-normalization (LID-SN) to measure the impact of these components on the downstream analysis.

Fitness estimates were also made using the MCAT' s[14] Interleave filter along with the SpatialMedian normalization with window size 9 (https://github.com/sauriiiin/sau-matlab-toolkit/blob/master/image2resBEAN.m). This normalization utilizes a moving window based median filter on interleaved colony size data to calculate a background colony size for every position on the plate. The observed colony sizes are divided by the background colony size to calculate the normalized colony sizes. This measure was utilized as a fitness equivalent.

Raw observed colony size estimates were also used as is, as 'fitness' measurements without performing any normalization (NO-NORM).

Measuring spatial bias and accuracy of background colony size

The coefficient of variance of fitness or colony size estimations per time point was used as a measure of spatial bias. Ten random observations were picked, with replacement, 2000 times to measure the coefficient of variance as a percentage of the mean (CV%). CV% distributions for LID, LID-AC, LID-SN, MCAT, and NO-NORM were compared using the Wilcoxon rank-sum test.
220 Accuracy of background colony size was measured using linear regressions models, and root mean square error (RMSE) estimation. The linear regression model between observed colony size and background colony size for all time points combined was made using the *lm* function in R[29]. The R^2 results were used to infer the accuracy of the background colony sizes. RMSE of the background colony size was calculated as a percentage of the average observed colony size
225 per time point. A random colony size predictor (RND) was used as a null model for background colony sizes used for spatial bias correction. The RND generated random colony sizes from a normal distribution using the *rnorm* function in R[29] using the mean and standard deviation of observed colony sizes per hour. Wilcoxon rank-sum test was used to compare RMSE results from LID, LID-AC, LID-SN, MCAT, and RND.

230

Empirical p-values and phenotype classification

Replicate wise mean relative fitness is calculated for every mock strain after removing outliers based on three median adjusted deviations. The reference strain mean relative fitness distribution is used to assign empirical p-values to all the mutant strains according to their rank in this
235 distribution (https://github.com/sauriiiin/lidetector/blob/master/lid.m). A p-value of 0.05 or below would mean that the mutant's mean relative fitness is in the top or the bottom 2.5^{th} percentile. The phenotype of mutant strains that are significantly different than the reference population is classified as 'beneficial' or 'deleterious' depending on increased or decreased relative fitness. The remaining mutant strains that do not have a significant change in fitness are classified as having
240 a 'neutral' phenotype.

Measuring the impact of the number of references and replicates

Reference proportion was sequentially reduced from 25% to 18.75% to 12.5% to 6.25% by masking 1/4th of the existing reference grid each time. This reduction was achieved by masking colonies on the 384-density mock reference plate and then propagating those masked colonies through the other densities. The number of replicates per strain was reduced in parallel by masking the nth replicate of every mock strain. Replicates were reduced from 16 to 2 in deprecations of 2. This process was repeated ten times to cover a variety of replicate combinations that are being masked (https://github.com/sauriiiin/paris/blob/master/techPowA.m).

Measuring specificity and sensitivity

Condition negative dataset

Colony size data from a particular time point represents the condition negative dataset as the mock reference, and the mutant strains are isogenic in reality. The proportion of mock mutant strains that are successfully called neutral by the LI Detector represents the true negative rate or specificity. The specificity was calculated for a sequentially increasing p-value cut-offs from 0 to 1 for colony size data available from the 12 time points.

Condition positive dataset

Two sets of virtual plates were created to generate a condition positive dataset. The first set of virtual plates contained a bimodal distribution of colony sizes (**Supplementary Fig. S4**) where colony size estimations for reference and mutant colony positions came from two different time points while maintaining their topological context (https://github.com/sauriiiin/paris/blob/master/techPowA.m). For example, if the colony size estimations for the mock references are taken from 11.0 hours and those for the mock mutants from 2.9 hours, then we have created a virtual plate where the mutant strains are smaller by design (**Supplementary Fig. S5**). The virtual plates retain the overall structure and spatial organization of the original plate by maintaining the topological context of the colonies. Since we

had a total of 12 time points, there can be 132 (12 times 12 minus the 12 where reference and mutant strains are at the same point) virtual plates. Each of these bimodally distributed virtual plates can be represented as a 12x12 fitness effect matrix (**Fig. 2**), where the fitness effect is the percentage change in the mutant median colony size as compared to the reference median colony size. Mock mutants that are successfully called beneficial or deleterious in these virtual plates are used to estimate the true positive rate or sensitivity of the LI Detector (https://github.com/sauriiiin/adaptivefitness/blob/master/scripts/4CX/4C_POWDY.R). For the virtual plates, a p-value cut off that controls the false positive rate at 5% was used to make across method examination of sensitivity comparable between LID, LID-AC, LID-SN, MCAT, and NO-NORM. False positive rate was controlled for each method by choosing a p-value cut-off at which 5% of the mock mutant strains were called significantly different than the mock reference.

The second set of virtual plates had randomly distributed colony sizes, where the reference colony size estimations were taken from a single time point, but the mutant colony size estimations were randomly selected from all the other time points (**Fig. 2**). In addition to maintaining the topological context, colony size estimates for replicates came from the same time point (https://github.com/sauriiiin/adaptivefitness/blob/master/scripts/4CX/4C_MESSUP.R). There were 11 such virtual plates. Since this set of virtual plates had both condition positive deleterious and beneficial mock mutants, the phenotype classification was further subdivided into all possible combinations stated in Supplementary Table S3.

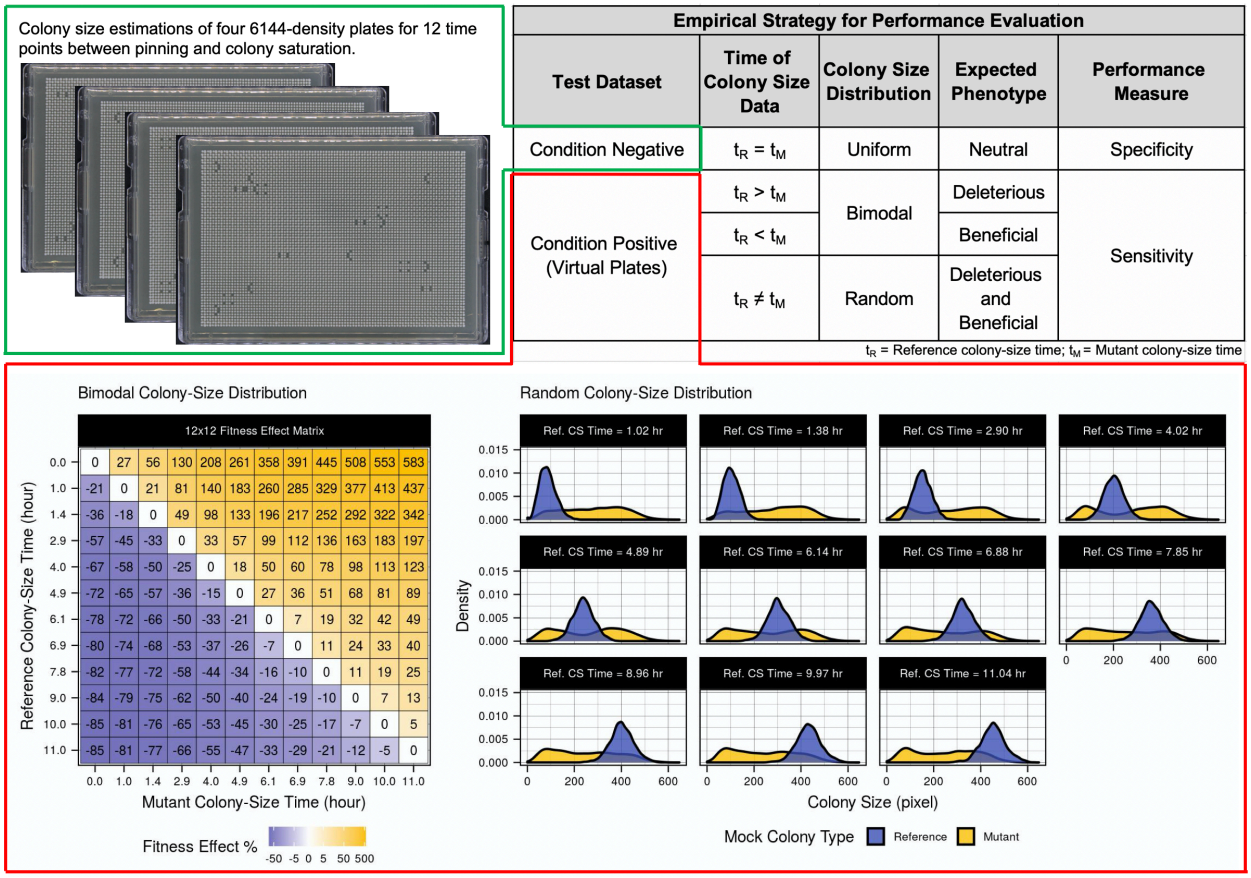

Figure 2. Empirical strategy for LI Detector performance evaluation. The condition negative dataset consisted of colony size estimations of four 6144-density plates that resulted from the validation experiment (top left). These plates were imaged at 12 different time points between pinning and colony saturation. The mock mutant phenotype at each time point is expected to be neutral and be utilized for measures of specificity. Two sets of virtual plates were created using a combination of mock reference and mutant colony size data taken from different time points (bottom). Virtual plates with bimodal colony size distribution had mock reference and mutant colony-size data at two separate time points. All mutants are either expected to be deleterious or beneficial. The 12x12 fitness effect matrix (bottom left) shows the effect sizes between the median reference and mutant colony size for each virtual plate. Virtual plates with random colony size distribution were made by picking mutant colony sizes from random time points that are different than that of the reference colony size estimations. The colony size distribution for each of these plates is shown in the bottom right. Both sets of virtual plates maintained the original spatial layout of the strains. Virtual plates were used to infer sensitivity.

Data availability

All data generated/analyzed in this study is available in the main text, in the Supplementary Figures and Tables, and as Supplementary Data files. All supplementary data are also on GitHub: https://github.com/sauriiiin/lidetector.

Code availability

The LI Detector analytical pipeline is written in MATLAB and provides an option between using SQL server databases or .csv files as input/output. It can be used in combination with any existing tools that generate colony size estimation from images. The user needs to provide pos2coor and pos2orf_name information along with the colony size estimation for LI Detector to work (see Materials and Methods). The pipeline is automated and goes from colony size estimations to empirical p-value calculations. The user has an option to choose if they want to use normalization features like local artifact correction and source normalization. These features are included by default. The code is available to download at https://github.com/sauriiiin/lidetector, along with instructions on how to use it.

Image processing, relative fitness estimations, and analysis are available at https://github.com/sauriiiin/sau-matlab-toolkit. All images within the main article and supplementary data were generated using code available at https://github.com/sauriiiin/adaptivefitness/tree/master/scripts/lid_paper.

RESULTS

Development of a new CBHTS framework

The LI Detector framework (**Fig. 1**) consists of complementary experimental and analytical pipelines that are specifically designed to eliminate spatial bias, to detect small changes in fitness, and to be able to identify increases as well as decreases in fitness. The experimental pipeline (**Fig. 1a**) follows a pin-copy-upscale protocol that serves two purposes. First, it reduces colony size differences that arise when colonies are pinned from glycerol stocks, and when they are upscaled. Second, it introduces a reference colony grid on every plate that is utilized by the analytical pipeline to correct spatial bias. The analytical pipeline (**Fig. 1b**) consists of three main steps - local artifact correction, source normalization, and reference-based normalization.

The local artifact correction step draws inspiration from the competition correction feature implemented by previous methods [15, 16]. It is designed to reduce spatially localized differences in colony size that are arising due to differential access to nutrients. The source normalization step controls for differences in colony sizes that arise as a result of the upscaling process. It was adopted from the interleaving feature of MATLAB Colony Analyzer Toolkit (MCAT [14]). Briefly, it deconstructs the colony size estimates of the higher density plates into subsets corresponding to the source plates used for the upscaling (**Supplementary Fig. S3**). Both local artifact correction and source normalization are provided as optional steps in the analytical pipeline.

For any location on the agar surface, the spatial bias can be expected to affect the reference and mutant colonies to an equal extent. The reference-based normalization step employs a 2-dimensional linear interpolant to predict background colony growth on every position of the plate based on the observed colony growth in the reference colony grid. Relative fitness is then assigned to every colony as a ratio of the observed colony size to the reference-predicted

background colony size. This estimate of relative fitness corrects for the local spatial bias without
350 making any assumptions of the underlying distribution of fitness effects.

In what follows, we compare the performance of LI Detector with one of the most versatile and robust tools available for correcting spatial bias, MCAT[14]. The overall workflow adopted for testing the performance of the LI Detector is described in **Figure 2** and **Supplementary Figure**
355 **S6**. This included performing a validation screen using an isogenic population using the LID experimental pipeline (see Materials and Methods). A randomly chosen subset of colonies was used as reference for spatial bias correction and the rest of the colonies were classified as mock mutants. The final 6144-density plates were imaged at 12 time points from initial pinning to colony saturation. Colony size estimates for each of these time points represented the condition negative
360 dataset where none of the mutants should be significantly larger (beneficial) or smaller (deleterious) than the reference. Two sets of virtual plates were created to represent the condition positive dataset. The first set was made by combining colony size estimates of the references and mutants from two different time points. These plates had a bimodal colony size distribution. The second set was made by randomly choosing mutant colony sizes from a time point other than
365 that of the reference and had a random colony size distribution. For both sets of virtual plates the spatial architecture of the colonies remained unchanged. The condition negative and positive datasets were utilized for spatial bias correction using LI Detector (LID), LI Detector without source normalization (LID-SN), LI Detector without local artifact correction (LID-AC) and MCAT. Doing this not only helped us evaluate LID's performance with MCAT but also investigate the
370 impact of various stages of normalization employed in the LI Detector. Results from the condition negative and positive dataset provided measures of specificity and sensitivity respectively.

LI Detector can accurately estimate background colony sizes and eliminate spatial bias

375 Variation in the colony sizes observed at any given time point would be coming from the natural biological variation in colony sizes that is expected for an isogenic population and the spatial bias. Hence, the fitness estimates obtained after spatial bias removal should only contain the biological variability, and the noise ensued from the statistical manipulation. We used the coefficient of variance percentage (CV%) of the colony sizes and fitness estimates per time point to see if LI
380 Detector could have a significant reduction in the variability (see Materials and Methods). LID, LID-AC, and MCAT showed a significant reduction in CV% as compared to NO-NORM, while LID-SN did not (**Supplementary Fig. S7**). This finding indicates that LID can reduce spatial bias and, source-normalization plays an important role in doing so.

385 The ability of LID to remove spatial bias is dependent on the accuracy with which it can measure background colony sizes on the entire plate by using the reference colony sizes. We used linear regression models to determine the quality of fit between the background and observed colony sizes for LID, LID-AC, LID-SN, MCAT, and RND (See Materials and Methods). LID and LID-AC had the highest R^2 (~0.99), signifying a good fit. MCAT was a close second with R^2 values barely
390 under 0.99, LID-SN had intermediate results (R^2 = 0.92), and RND performed poorly (R^2 = 0.81) (**Fig. 3a-e**). Given these findings, the root mean square error (RMSE) between background and observed colony sizes per time point was calculated as a percentage of the mean colony size at the time point. LID, LID-AC, and MCAT RMSE% were indistinguishable for the higher time points when the colonies begin to saturate (**Fig. 3f**, Supplementary Data: RMSESTAT.csv). RMSE% for
395 LID-SN was significantly higher than LID ($p = 0.00019$, Wilcoxon rank-sum test) again, indicating the impact of performing source-normalization.

Then we wanted to know if source-normalization was effective in reducing source-based bias. To do that, we compared raw colony size and fitness results of the source-deconstructs for the plates
400 at saturation (time = 11.04 hours, **Fig. 3g-k**). The raw colony size estimates of the four source-

deconstructs (**Fig. 3g**) show a significant difference in their distribution (Kruskal-Wallis, $p < 2.2E-16$). This difference in distribution is the result of the source plate bias that needs to be corrected. As expected, LID-SN could not get rid of the source plate bias (**Fig. 3i**, Kruskal-Wallis, $p < 2.2E-16$). Whereas, fitness distributions resulting from LID-AC, LID, and MCAT (**Fig. 3i-k**) do not show any significant difference between the source-deconstructs. These results indicate that LID and LID-AC are comparable to MCAT for removing source-based bias.

Overall, these findings show that both local and global spatial context is crucial to consider when correcting for spatial bias. The LI Detector, just like MCAT[14], is good at integrating global and local bias.

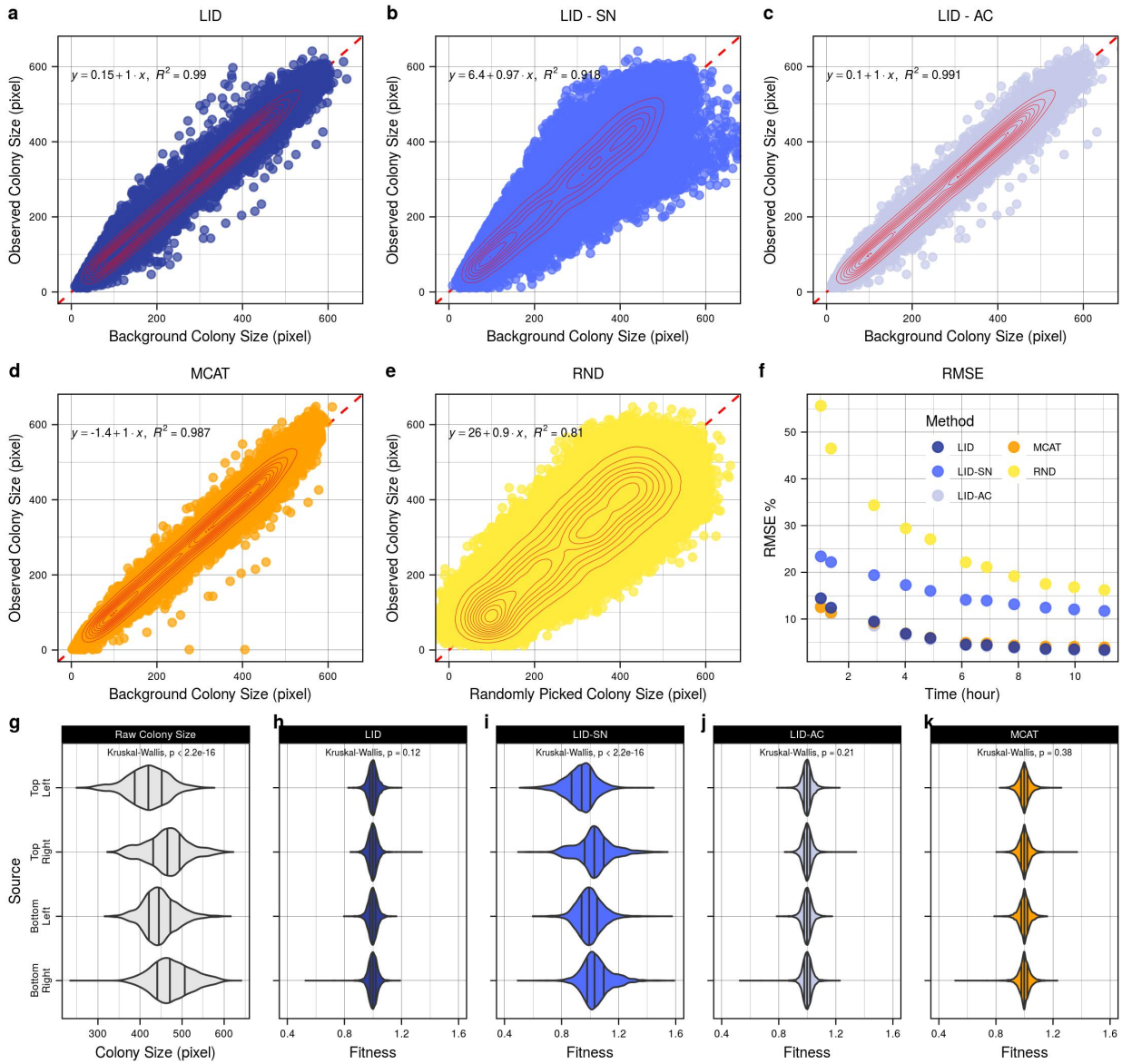

Figure 3. Accuracy of background colony-sizes. Effectiveness of spatial bias correction is dependent upon the background colony-sizes. **a-f.** We use the condition negative set to compare the background and observed colony-sizes for LID, LID-SN, LID-AC, MCAT, and RND. **a-e** Scatter plots showing the relationship between background and observed colony size (pixel count). The scatter plots are overlaid with a 2-dimensional density plot (solid red lines) to visualize spread. Linear models that describe the relationship along with their R^2 metric are shown in the top left of each plot. The background colony size should be as close to the observed colony size as possible for optimal spatial bias correction. **f.** Root mean square error (RMSE) of the background colony sizes as compared to the observed colony size as a percentage of mean observed colony size per time point. Colors represent different modes of normalization. **g-k.** Source-wise violin plot of raw colony sizes and LID, LID-SN, LID-AC, and MCAT normalized fitness at saturation (time = 11.04 hours). Solid lines in the violin plot indicate median and the 25th and 75th percentile. The source-wise distributions are compared using a non-parametric (Kruskal-Wallis) test, and the results are displayed at the top of each plot. LID = LI Detector, LID-AC = LI Detector without local artifact correction, LID-SN = LI

425 Detector without source normalization, MCAT = MATLAB Colony Analyzer Toolkit, RND = Random prediction of colony sizes.

LI Detector identifies small fitness effects with high specificity and sensitivity

430 To evaluate LID's ability to detect neutral, beneficial and deleterious fitness effects, we estimated its specificity and sensitivity using our condition negative and positive datasets, respectively (**Figure 2**). Specificity was estimated as the proportion of mock mutant strains that were correctly classified at neutral for varying p-value cut-off using the condition negative dataset. LID's specificity was more than 98% for a p-value cut off of 0.05 and remained above 95% when that 435 cut-off was increased to 0.1 (**Fig. 4a**). For comparison, MCAT showed a maximum specificity of 94.5% for a p-value cut off of 0.05 using the same condition negative dataset (**Supplementary Fig. S8**).

Sensitivity was estimated as the proportion of mock mutant strains correctly classified as either 440 beneficial or deleterious at a false positive rate of 5% using the condition positive dataset. LID's sensitivity was more than 95% for beneficial and deleterious fitness effects of 5%, and reached 100% for fitness effects of about 7% (**Fig. 4b**). These findings show that, not only is LID highly sensitive in observing small fitness effects, but it is equally sensitive towards both increases and decreases in fitness. This result was dependent on the mode of normalization with LID performing 445 significantly better than LID-SN and NO-NORM (**Supplementary Fig. S9-10**). We also analyzed the same data using MCAT and found that while it was 80% sensitive in detecting 5% fitness decreases, it was only 40% sensitive when it came to 5% fitness increases (**Fig. 4c**).

Next, we wanted to observe how the number of references per plate and the number of replicates 450 per strain affects the sensitivity of observing small fitness effects. To do this, we computationally masked portions of the reference-colony grid and replicates, and then reanalyzed the condition

positive dataset (see Materials and Methods). We observed that the LID's sensitivity in detecting 5% fitness effects was directly related to the number of reference colonies per plate and the number of replicates per mutant (**Fig. 4c**). This observation, in part, is consistent with the finding that RMSE% is inversely related to the number of reference colonies per plate (**Supplementary Fig. S11**). We reasoned that a higher proportion of references would increase the accuracy of the reference-based background colony size prediction, thus improve spatial bias removal and increase sensitivity. The increased number of replicates would randomize technical bias by being spread across plates and locations on the colony grid.

LID's specificity, on the other hand, is independent of the mode of normalization, the proportion of references per plate, and the number of replicates per mutant and always remains above 95% (**Supplementary Fig. S12**).

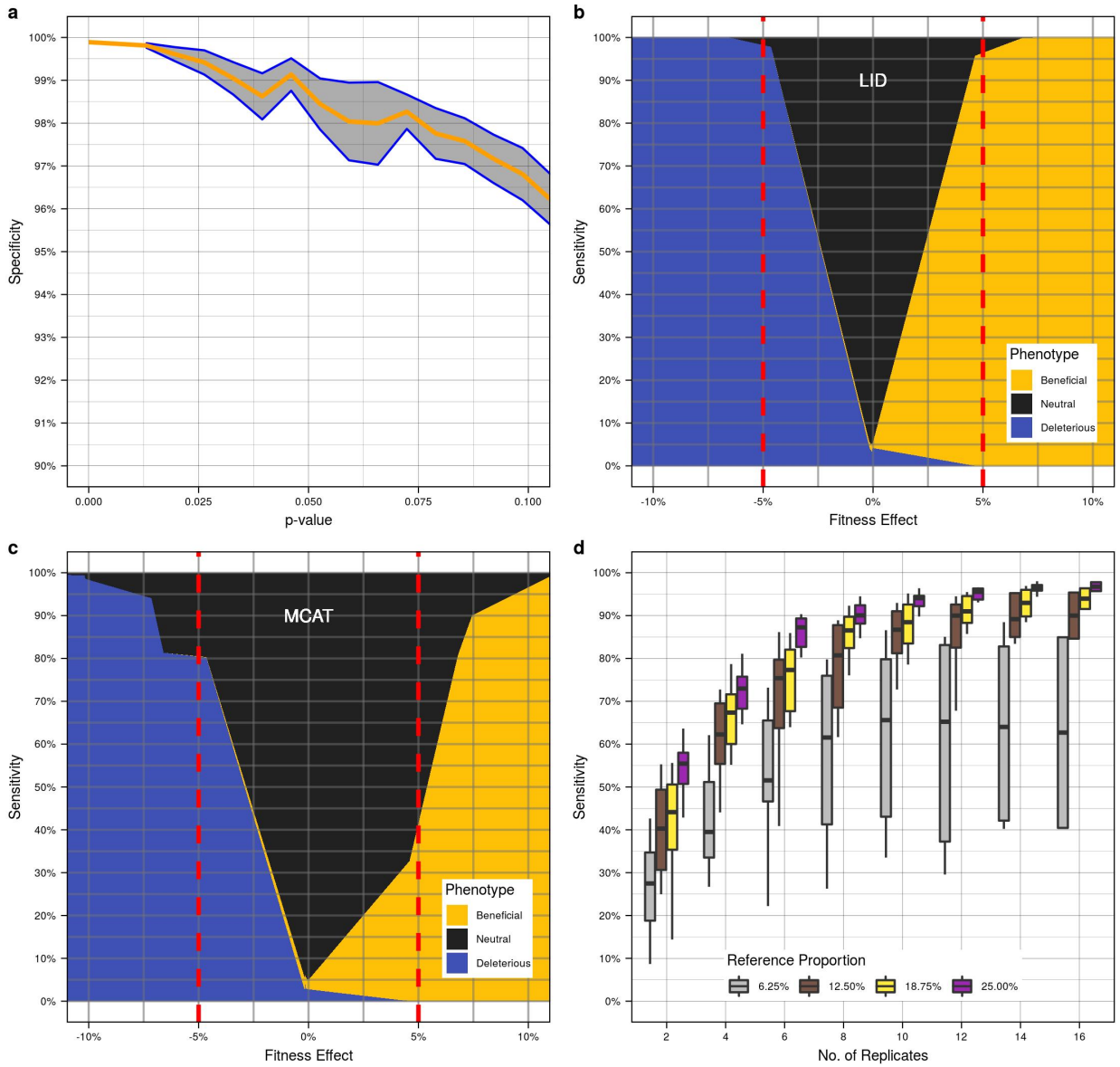

Figure 4. LID has high specificity and sensitivity. a. Average specificity (solid yellow line) and standard error (solid blue lines) for all time points at various p-value cut-offs. LID has more than 95% specificity for p-values less than 0.1. **b.** Phenotype classification results at a false positive rate of 5% from the virtual plates with bimodal distributions are arranged according to increasing fitness effects. They show that LID can observe fitness effects as low as 5% with more than 95% sensitivity (red dashed line). **c.** MCAT phenotype classification results from the same data as **b.** show 80% and 40% sensitivity in observing 5% fitness decreases and increases respectively. **d.** Sensitivity directly related to the number of replicates and references**.** Box plot of the sensitivity to detect fitness effects of 5% or less at 5% false positive rate for varying number of replicates per strain and proportion of references per plate. Sensitivity is at its highest when there are 16 replicates per strain, and 1/4th of all colonies are references.

LI Detector maintains high sensitivity even when colony size distribution is random

At this point, we wanted to evaluate LI Detector's performance when the underlying fitness distribution is random - *a priori* assumptions of colony size are difficult to make. Virtual plates with random colony size distribution were used for this evaluation. Eleven such plates were made by combining colony size estimates of reference from one time point with mutants from any other time point chosen at random. The overall colony size distribution of these virtual plates is shown in **Fig. 2**. The 11 plates combined had 45.98% condition positive beneficial mutants, and 54.02% condition positive deleterious mutants. LID was successful in calling 45.19% as true positive beneficial and 53.81% as true positive deleterious, which is 1% false negative rate (**Fig. 5a left panel**, paired Student's t-test on true positives, p-value = 0.335). We never observed a condition positive beneficial being called by LID as deleterious or vice-versa. In comparison, MCAT could only identify 34.89% as true positive beneficial and 42.99% as true positive deleterious (paired Student's t-test on true positives, p-value = 0.002), which was significantly lower than LID (paired Student's t-test on true positives, p-value = 0.015). MCAT also had a 21.64% false negative rate, and a small number (0.49%) of switched condition positive calls (**Fig. 5a right panel**). Plate-wise phenotype results show that MCAT's false negatives were spread all across them (**Fig. 5b**). We also notice that LID's false negatives had a much smaller fitness effect than that of MCAT's (**Fig. 5c**). Overall, these results indicate that LID specifically outshines existing methods of bias correction in scenarios where *a priori* assumptions of the underlying colony size distribution can prove to be limiting.

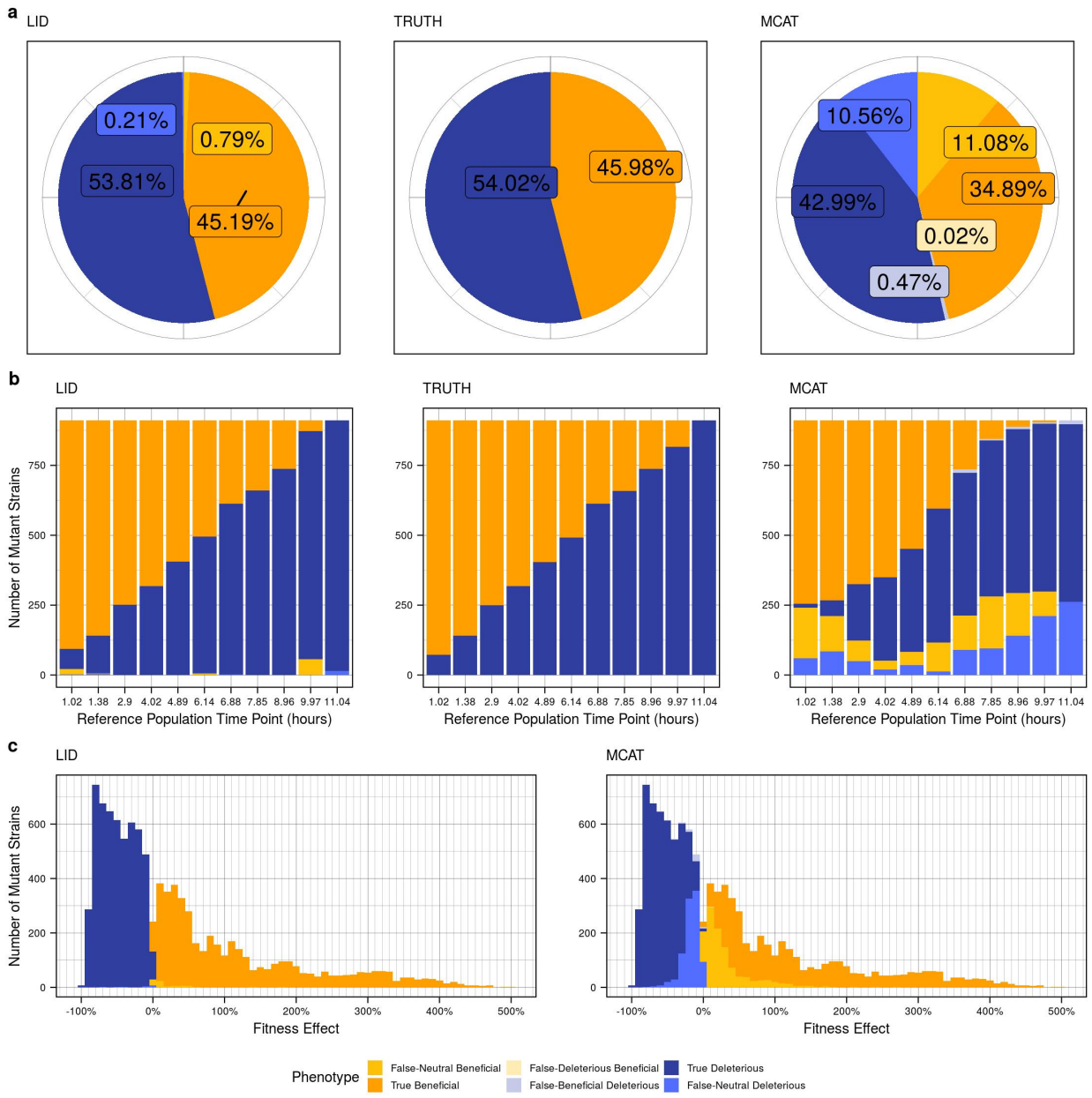

Figure 5. LID maintains high sensitivity even when underlying fitness distribution is random. Combined results for all the virtual plates with random colony size distribution. The middle panel is a pie chart of the condition positive mutants in the dataset, with 45.98% condition positive beneficial and 54.02% condition positive deleterious. The left and right panels show pie charts of the phenotype classification results from LID and MCAT, respectively. LID had a 1% false negative rate. MCAT had 21.64% false negative and 0.49% switched positive. **b.** Virtual plate-wise results of the phenotype classification. The plates are arranged according to their reference colony time point. The middle panel represents the ground truth, the left panel represents results from using LID, and the right panel represents results from MCAT. Each bar represents the number of mock mutants on the plate. MCAT false negative are distributed across all the plates. **c.** Bar graph of pooled results from all plates arranged according to the fitness effects. LID false negatives have smaller fitness effects as compared to MCAT. True deleterious and true beneficial categories are the true positives. False-neutral beneficial and false-neutral deleterious are the false

negatives of when a beneficial or deleterious strain is called neutral. False-deleterious beneficial and false-
515 beneficial deleterious are switched positive calls where a beneficial strain is called deleterious and vice-versa.

DISCUSSION

We present to you a CBHTS framework that can be adapted to any experiment and added on to existing analytical pipelines. With the help of a mock experiment involving an isogenic population and computational techniques like the virtual plates, we were able to thoroughly examine the performance of the LI Detector in a controlled manner. The utility of high-throughput screens has often been to pick out high effect targets considering their high false-positive and false-negative rates. LI Detector remedies this limitation by offering a highly specific and sensitive method (**Fig. 4a, b; Supplementary Fig. S12**), thereby giving a higher confidence in the results generated. LI Detector also increases the utility of CBHTS by being highly sensitive towards finding small fitness effects (**Fig. 4b; Supplementary Fig. S9-10**). It can detect fitness effects as low as 5% with more than 95% sensitivity (**Fig. 4b**). As opposed to existing methods like MCAT (**Fig. 4c**), LI Detector is equally sensitive towards detecting both increases and decreases in fitness (**Fig. 4b**).

The increased sensitivity, specificity, and ability to detect small fitness effects in both directions would be useful in providing highly granular phenotypic information from CBHTS. This feature can be especially valuable for improving the current gene-gene, gene-environment and, protein-protein interaction networks for colony-forming-microorganisms. LI Detector's ability to detect increases in fitness also makes it a favorable tool for screens that examine gain-of-function mutations, questions of evolutionary biology[30], and pharmacological screens of adaption and resistance. Another obstacle the LI Detector overcomes is that of the scale of CBHTS. Existing methods of spatial bias correction work best in genome-wide studies with a large number of plates and mutants[17]. Alternate methods of extracting phenotypic information have been developed to increase sensitivity at the small scale level[24]. LI Detector provides a flexible approach that can be applied to CBHTS regardless of their scale. LI Detector would work the same when used for a screen involving a handful of mutants of the active site of your favorite gene (YFG) made to figure

out the most critical residue, or a genome-wide synthetic genetic array (SGA) used to infer the genetic interactions of YFG.

One of the caveats for the effective use of LI Detector include that it requires both the final density plates and the penultimate density plates if source normalization is to be used, to have a proportion of colonies dedicated to a reference strain. We have shown that LI Detector's accuracy in predicting background colony sizes and its sensitivity in detecting small fitness effects is directly related to the proportion of reference colonies on a plate (**Supplementary Fig. S11; Fig. 4d**). A higher reference proportion can be used if the goal is to look for subtle fitness effects like in the deletion of non-essential genes. Alternatively, a lower reference proportion may be used where larger fitness effects are desired, like finding the most drug-resistant mutant. Regardless of the experiment, a portion of the plate will have to be sacrificed for references. This requirement can pose as an experimental and economic burden. Alternatively, if the goal is to identify larger fitness effects only then existing methods like MCAT may be employed instead of LID. Our results have also shown that MCAT performs as well as LID in scenarios where *a priori* assumptions of fitness distribution are reasonable.

In summary, the LI Detector framework experimentally introduces a reference population grid on plates whose colony size estimates are used to correct for spatial bias independently of the underlying distribution of fitness effects. It has the potential to expand the utility of CBHTS by making them independent of scale, sensitive towards small fitness effects, and equally sensitive in detecting increases and decreases in fitness. Although developed and validated using yeast, it can be applied to all colony-forming-microorganisms, including clinically relevant isolates. The LI Detector is free to download at https://github.com/sauriiiin/lidetector, along with instructions on how to use it.

570 **REFERENCES**

1. Saleski, T.E., et al., *Syntrophic co-culture amplification of production phenotype for high-throughput screening of microbial strain libraries.* Metab Eng, 2019. **54**: p. 232-243.
2. Zeng, W., et al., *High-Throughput Screening Technology in Industrial Biotechnology.* Trends Biotechnol, 2020.
3. Priola, J.J., et al., *High-throughput screening and selection of mammalian cells for enhanced protein production.* Biotechnol J, 2016. **11**(7): p. 853-65.
4. Rallis, C. and J. Bahler, *Cell-based screens and phenomics with fission yeast.* Crit Rev Biochem Mol Biol, 2016. **51**(2): p. 86-95.
5. Roguev, A., et al., *Conservation and rewiring of functional modules revealed by an epistasis map in fission yeast.* Science, 2008. **322**(5900): p. 405-10.
6. Costanzo, M., et al., *The genetic landscape of a cell.* Science, 2010. **327**(5964): p. 425-31.
7. Yang, F., et al., *Development and application of a recombination-based library versus library high- throughput yeast two-hybrid (RLL-Y2H) screening system.* Nucleic Acids Res, 2018. **46**(3): p. e17.
8. Uetz, P., et al., *A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae.* Nature, 2000. **403**(6770): p. 623-7.
9. Rajagopala, S.V. and P. Uetz, *Analysis of protein-protein interactions using array-based yeast two-hybrid screens.* Methods Mol Biol, 2009. **548**: p. 223-45.
10. Parsons, A.B., et al., *Integration of chemical-genetic and genetic interaction data links bioactive compounds to cellular target pathways.* Nat Biotechnol, 2004. **22**(1): p. 62-9.

11. Parsons, A.B., et al., *Exploring the mode-of-action of bioactive compounds by chemical-genetic profiling in yeast.* Cell, 2006. **126**(3): p. 611-25.

12. Vizeacoumar, F.J., et al., *Integrating high-throughput genetic interaction mapping and high-content screening to explore yeast spindle morphogenesis.* J Cell Biol, 2010. **188**(1): p. 69-81.

13. Butland, G., et al., *eSGA: E. coli synthetic genetic array analysis.* Nat Methods, 2008. **5**(9): p. 789-95.

14. Bean, G.J., et al., *Development of ultra-high-density screening tools for microbial "omics".* PLoS One, 2014. **9**(1): p. e85177.

15. Wagih, O., et al., *SGAtools: one-stop analysis and visualization of array-based genetic interaction screens.* Nucleic Acids Res, 2013. **41**(Web Server issue): p. W591-6.

16. Young, B.P. and C.J. Loewen, *Balony: a software package for analysis of data generated by synthetic genetic array experiments.* BMC Bioinformatics, 2013. **14**: p. 354.

17. Baryshnikova, A., et al., *Quantitative analysis of fitness and genetic interactions in yeast on a genome scale.* Nat Methods, 2010. **7**(12): p. 1017-24.

18. Lawless, C., et al., *Colonyzer: automated quantification of micro-organism growth characteristics on solid agar.* BMC Bioinformatics, 2010. **11**: p. 287.

19. Dittmar, J.C., R.J. Reid, and R. Rothstein, *ScreenMill: a freely available software suite for growth measurement, analysis and visualization of high-throughput screen data.* BMC Bioinformatics, 2010. **11**: p. 353.

20. Bean, G.J. and T. Ideker, *Differential analysis of high-throughput quantitative genetic interaction data.* Genome Biol, 2012. **13**(12): p. R123.

21. Collins, S.R., et al., *A strategy for extracting and analyzing large-scale quantitative epistatic interaction data.* Genome Biol, 2006. **7**(7): p. R63.

22. Levin-Reisman, I., et al., *Automated imaging with ScanLag reveals previously undetectable bacterial growth phenotypes.* Nat Methods, 2010. **7**(9): p. 737-9.

23. Zackrisson, M., et al., *Scan-o-matic: High-Resolution Microbial Phenomics at a Massive Scale.* G3 (Bethesda), 2016. **6**(9): p. 3003-14.

24. Herricks, T., et al., *One-Cell Doubling Evaluation by Living Arrays of Yeast, ODELAY!* G3 (Bethesda), 2017. **7**(1): p. 279-288.

25. Wloch, D.M., et al., *Direct estimate of the mutation rate and the distribution of fitness effects in the yeast Saccharomyces cerevisiae.* Genetics, 2001. **159**(2): p. 441-52.

26. Van den Bergh, B., et al., *Experimental Design, Population Dynamics, and Diversity in Microbial Experimental Evolution.* Microbiol Mol Biol Rev, 2018. **82**(3).

27. Joseph, S.B. and D.W. Hall, *Spontaneous mutations in diploid Saccharomyces cerevisiae: more beneficial than expected.* Genetics, 2004. **168**(4): p. 1817-25.

28. Winston, F., C. Dollard, and S.L. Ricupero-Hovasse, *Construction of a set of convenient Saccharomyces cerevisiae strains that are isogenic to S288C.* Yeast, 1995. **11**(1): p. 53-5.

29. Team, R.C., *R: A language and environment for statistical computing.* 2013.

30. Vakirlis, N., et al., *De novo emergence of adaptive membrane proteins from thymine-rich genomic sequences.* Nat Commun, 2020. **11**(1): p. 781.

31. Bischof, L., et al., *Spotsizer: High-throughput quantitative analysis of microbial growth.* Biotechniques, 2016. **61**(4): p. 191-201.

635 32. Wagih, O. and L. Parts, *gitter: a robust and accurate method for quantification of colony sizes from plate images.* G3 (Bethesda), 2014. **4**(3): p. 547-52.