https://doi.org/10.1093/nargab/lqab089

Deep learning identifies erroneous microarray-based, gene-level conclusions in literature

Yanan Qin[1], Daiyao Yi[1], Xianghao Chen[1] and Yuanfang Guan[1,2,*]

[1]Department of Computational Medicine and Bioinformatics, University of Michigan Medical School, Ann Arbor, MI 48109, USA and [2]Department of Internal Medicine, University of Michigan Medical School, Ann Arbor, MI 48109, USA

Received February 10, 2021; Revised August 25, 2021; Editorial Decision September 12, 2021; Accepted September 17, 2021

ABSTRACT

More than 110 000 publications have used microarrays to decipher phenotype-associated genes, clinical biomarkers and gene functions. Microarrays rely on digital assaying the fluorescence signals of arrays. In this study, we retrospectively constructed raw images for 37 724 published microarray data, and developed deep learning algorithms to automatically detect systematic defects. We report that an alarming amount of 26.73% of the microarray-based studies are affected by serious imaging defects. By literature mining, we found that publications associated with these affected microarrays have reported disproportionately more biological discoveries on the genes in the contaminated areas compared to other genes. 28.82% of the gene-level conclusions reported in these publications were based on measurements falling into the contaminated area, indicating severe, systematic problems caused by such contaminations. We provided the identified published, problematic datasets, affected genes and the imputed arrays as well as software tools for scanning such contamination that will become essential to future studies to scrutinize and critically analyze microarray data.

INTRODUCTION

Since their invention >20 years ago, microarrays have had myriad applications, including discovering gene functions, biomarkers for diseases and biological pathways (1,2). Microarrays are widely used and an essential tool in all areas of biology and medicine. Today, >110 000 papers have been published using microarray data. In 2020 alone, ∼6800 microarray-based studies have been published. To the majority of biologists or bioinformaticians, transcriptomic data from microarrays are presented as a list of numbers for probes, transcripts or genes. The raw data format, a fluorescence image directly acquired from the microarray facilities, is rarely accessible to the users. As matter of fact when using microarray data to make biology discoveries, no one can be assured that the original fluorescence image is valid and meaningful.

A microarray measures gene expression by probes that bind to reversely transcribed RNAs (Figure 1A). The bound probes illuminate fluorescent light and the intensity is measured to represent the expression level of the genes (3). Each gene is mapped to multiple probes corresponding to several independent sampling regions of the gene (Figure 1B)—this property will turn out to be useful as independent support to the image-based analysis pipeline developed in this study. Fluorescence signals are subject to several sources of noise. One is cross-array batch effects, which are often corrected by adjusting the readout values by the overall distribution of the microarray chip (4–9). The other source of noise comes from defects localized at certain parts of the array. Methods have been developed to correct background noises based on the estimation of intensity of the neighboring pixels (10–13), with the assumption that the surrounding areas of the noise spot are correct. As we will see in the analysis later, such assumptions are not complete, and cannot account for large quantities of contaminations.

In this study, we carried out a retrospective examination of 37 724 microarray samples. We reconstructed the fluorescence images for these microarray data from probe readouts, and hand labeled defective areas. This allowed us to develop a deep learning model that automatically detects the contaminations in such images and associated microarray data. Reanalyzing the expression data reveals less coordinance across probes of the same gene affected by contamination. We found that 4.80% of the microarrays, corresponding to 26.73% of the studies, are affected by such defects, while literature mining showed a disproportionate enrichment of reporting of significant findings of the genes in the affected area. Overall, 28.82% of the gene-level conclusions reported in these publications were in fact based on measurements falling into the contaminated area, while such contamination only occupies 2.78% in area on these images. This implies that a large quantity of microarray-based publications were making conclusions based on contamination rather than biology. We provided this microar-

[*]To whom correspondence should be addressed. Tel: +1 734 764 0018; Email: gyuanfan@umich.edu

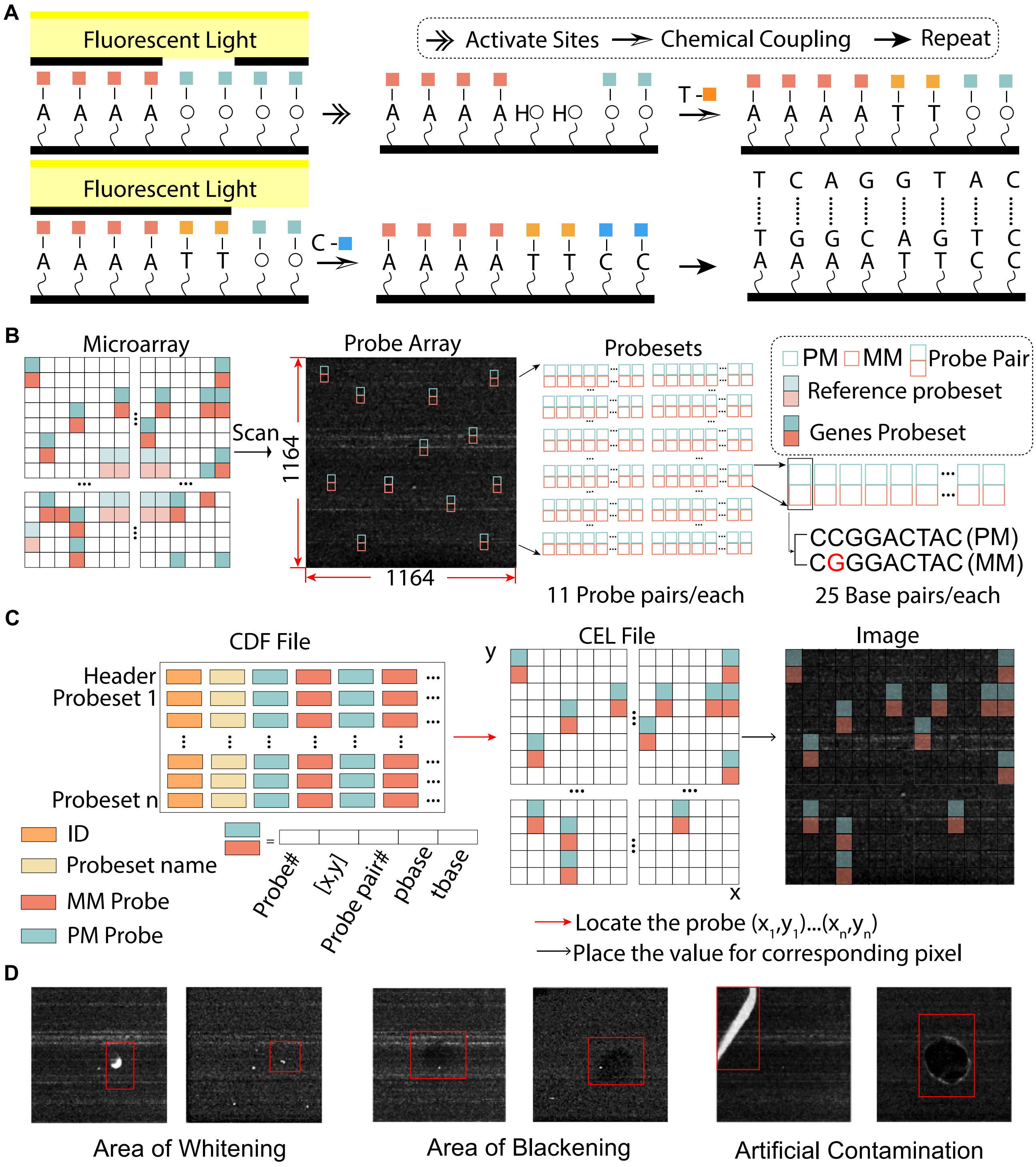

Figure 1. Reconstruction of microarray images generated by Affymetrix GeneChip. (**A**) Illustration of microarray sample processing and hybridization. (**B**) HG-U133_Plus_2 GeneChip microarray distribution: one microarray represents over 20 000 genes, each of which has at least one corresponding probeset. Each probeset contains 11 probe pairs [perfect match (PM):mismatch (MM)], and the probe pairs are scattered on the whole microarray. There is one PM probe cell and one MM probe cell in each pair. (**C**) Mapping CEL files to images according to CDF annotation. (**D**) Reconstruction of the microarray images reveals different types of contamination on the original microarray chips.

ray contamination information as well as re-imputed microarray data using uncontaminated surrogate probes as a resource to the research community, and we also provided the code to scan for such contamination, in the hope that this can inform the trustability of the relevant studies and can be useful to future studies that reanalyze these datasets.

MATERIALS AND METHODS

Recovering of microarray images from CEL files

We reconstructed and hand labeled 37 724 published Human Genome U133 Plus 2.0 Array (HG-U133_Plus_2) image files (see the 'Data Availability' section). These images came from a total of 3165 studies. Although microarray data are presented as gene- or probe-level expression levels, the original data were acquired from reading the fluorescent level on the microarray chips. Retrospectively reconstructing the microarray images from CEL files was a critical and challenging step in this study. We accomplished it by writing an API to the Affymetrix software developer's kit that is internally used by Affymetrix; this API maps the value of each probe to a unique position on the original image (Figure 1C). This API is shared at https://github.com/GuanLab/Microarray.

For the HG-U133_Plus_2 platform, it is a 1164 × 1164 matrix, corresponding to a total of 1 354 896 possible probe positions (Figure 1B). The HG-U133_Plus_2.cdf file provided by Affymetrix gives a total of 54 675 probesets. For each probeset, it gives a list of associated probes and their corresponding x, y positions in the 1164 × 1164 matrix (Figure 1B). This allows us to remap the probe-level readout values into an image, typically with a value between -3.30×10^{38} and 6.08×10^{37}. On average, each probeset contains 22 probes, and this value ranges between 16 and 138 (Supplementary Figure S1). This information is useful for us to reconstruct and estimate the values in contaminated areas later. Each probe is 25 bp. The HG-U133_Plus_2 platform has 62 reference probesets used to adjust microarray facility and readouts, and the remaining probesets correspond to genes. Each gene typically has a couple of probesets (average 2, range 1–22) associated with it, corresponding to different DNA positions in the gene (Supplementary Figure S1). Visually inspecting a normal microarray image will immediately reveal the position of a group of reference probesets that form a bright spot at about the middle position of the microarray image. However, not all reference probes are located at this spot. It is a typical design for the probes of a gene to be scattered across images as well, which is intended to minimize the effect of contamination.

The probes in one probeset are always designed in a series of pairs. One has a single base pair change from the reference genome, the other is a perfect match of the original sequence, and the two probes of the pair are immediately adjacent to each other (Figure 1B). For example, probe 200097_s_at, mapped to the gene HNRNPK, contains a total of 22 probes. These probes are grouped into 11 pairs positioned at (row column) [(144 259); (144 260)], [(1051 1101); (1051 1102)], [(257 901); (257 902)], [(146 95); (146 96)], [(883 997); (883 998)], [(675 829); (675 830)], [(856 597); (856 598)], [(916 249); (916 250)], [(397 327); (397 328)], [(1000 741); (1000 742)] and [(856 1099); (856 1100)] (Figure 1B). For the probe at (144 259), it has a single base pair change from G→C compared to the reference genome and the probe at (144 260) (Figure 1B). This design is intended to correct noises in the image. The inference of a probe value takes into account the ratio of the readout of the perfect match over the non-perfect match.

Web interface-based annotation tool

We first scaled the values of the images to 0–255 to allow viewing on a computer screen. Hand-annotating tens of thousands of microarray images is a tedious task. To facilitate efficient annotation, we developed an annotation tool, which allows fast annotation of the images through dragging the mouse. Using this tool, we can load a predefined number of images on a single screen. After labeling each page, the next screen is loaded by refreshing the webpage. The unlabeled images will be then recycled after every image is seen, and thus allowed us to double label the collection of the microarray images. If a wrong annotation is made, the annotation tool allows unsetting the annotation. All annotations are then dumped into a database and ready to be trained.

The labeling procedure was carried out below: because the annotation tool allows fast viewing of many images at a time (typically thousands in an hour) through mouse operation. The labelers first went through the images once to observe the overall pattern of the images. This procedure helped us to recognize several distinct patterns (characterized by stripes) likely produced by different microarray machine production batches. These are not defects. Then, the labelers proceeded with hand labeling of the images. We double labeled the images by two passes and discussed ambiguous cases together. We used the union of the two as the gold standard, as it is much easier to miss out defect regions than to find false positives.

Deep learning training loss and procedures

We used a representative U-Net structure (Figure 2A). The nonlinear activation after each convolution layer is a rectified linear unit. At the output layer, we used sigmoid activation. A combination of cross-entropy (CE) and mean square error (MSE) loss was used for model training and it is defined as

$$\mathrm{CE} = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i \log\left(\widehat{y_i}\right) + (1 - y_i)\log\left(1 - \widehat{y_i}\right)\right),$$

where N is the number of test cases, y is the true value and $\hat{y}$ is the predicted value that was clipped by [1e−7, 1 − 1e−7] due to the use as log loss.

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \widehat{y_i}\right)^2$$

and

$$\text{combined loss} = \mathrm{CE} + \mathrm{MSE}.$$

Training and testing loss is included in Supplementary Figure S3.

The whole training process was carried out by Keras and TensorFlow. In each cross-validation fold, we carried out

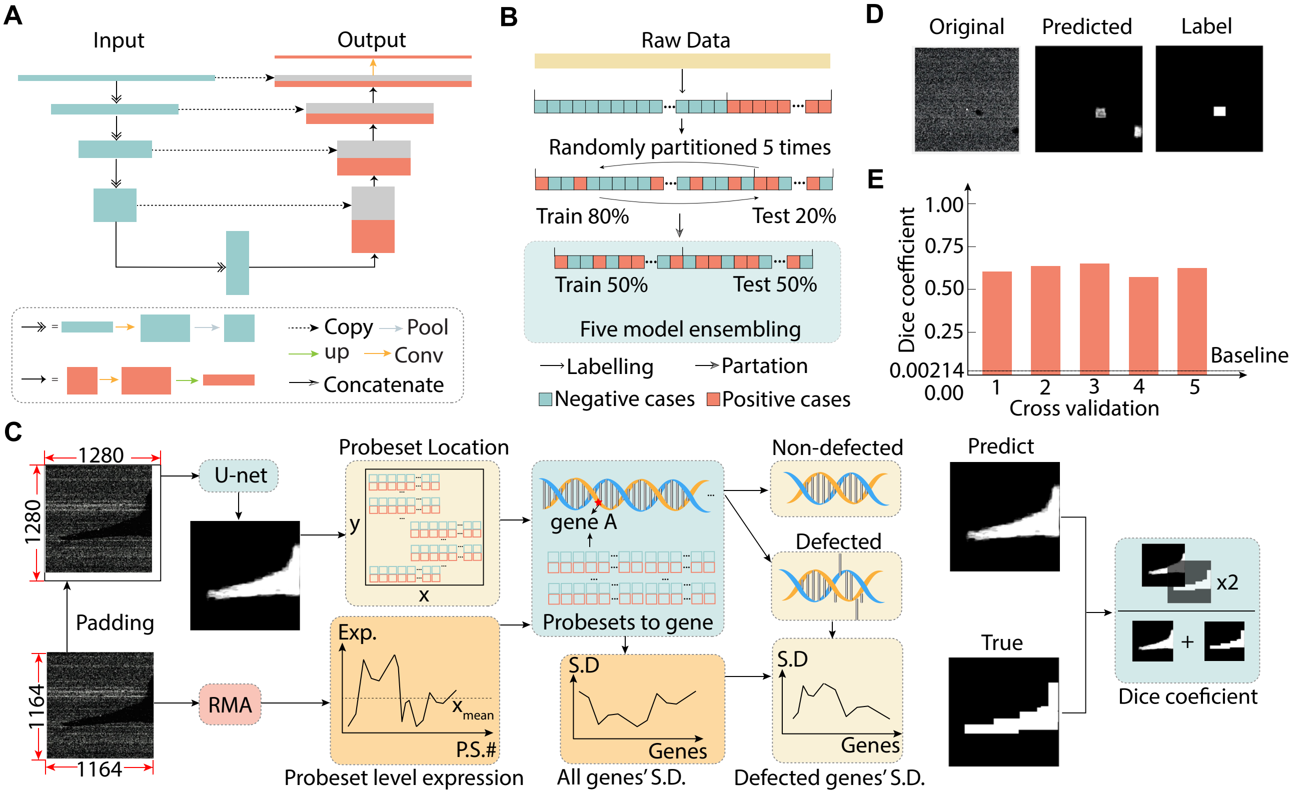

Figure 2. Overview of the workflow of the algorithm. (**A**) U-Net model structure. (**B**) Partition of images into training, validation and test sets. (**C**) Image preprocessing, U-Net model training and result evaluation. (**D**) Examples of model output, compared with human labels: our model can detect the defective areas that were initially identified and missed by human labeling. White regions in the output/label images indicate the defected areas. (**E**) Dice coefficients for each fold of cross-validation are 0.6054, 0.6356, 0.6493, 0.5722 and 0.6244.

nested training to generate five models to ensemble into the final parameters. For each model, training samples were resampled to balance the positives (with defects) and negatives (without defects) by bootstrap resampling and then were further divided, respectively, into the true training (50%) and validation sets (50%) randomly for five times (Figure 2B). Models were trained for five times using different random seeds. The validation sets were used to call back the best model. Network structure is presented in Figure 2A, and small variations of this structure by filter sizes and network depth did not result in substantial changes in performance. We trained with a batch size of 2, Adam optimization with learning rate of 3e−5 and the combination of cross entropy loss and MSE loss. We iterated for five epochs corresponding to over 100 000 batches of samples, and the training loss was fairly stable at the end of the training process.

Evaluation procedure

The Dice coefficient was used for model evaluation by measuring the overlap between the predicted area (A) and gold standard annotation (B). The Dice coefficient is defined as

$$\mathrm{Dice}_{A,B} = \frac{2\,|A \cap B|}{|A| + |B|}.$$

In order to further confirm the contamination and the effectiveness of our model, we compared the expression of genes with and without contamination. Specifically, we compared the deviation of probeset expression levels of the two kinds of genes. First of all, we set a threshold = 0.5: for every pixel of a predicted segmentation mask of defected regions, if it is ≥0.5, we consider it contaminated and otherwise not. By extracting the location-to-probeset mapping information from the CDF file and the probeset-to-gene mapping information from the corresponding annotation on R Bioconductor, we were able to map probe locations to genes and know which genes have contamination and which do not. In the meantime, we have CEL files for each sample and calculated robust-multiarray average (RMA) for each dataset. The resulting information is the transformed expression level for each probeset in every sample. For each sample, due to the factors about microarray design mentioned above, we corrected the affinity biases across probes by matrix-wisely dividing the mean value of each probeset across all microarrays without contamination. Combining the resulting corrected expression on probeset level, probeset-to-gene mapping formation and gene identity (defected versus non-defected), we calculated standard errors for all probesets in defected genes and non-defected genes. The average levels of the deviation from two groups of genes were then compared.

Procedure to recover contaminated microarray data

We extracted the probe intensities of each sample using R package affxparser from the corresponding CEL file and resulting 1164 × 1164 matrix. We selected 35 043 samples without predicted contamination, positive label or missing values and calculated the average intensity matrix for reference. One thousand seven hundred three contaminated samples were recovered following the logistics described in the 'Results' section. In order to make our recovered data more user friendly, we extracted the probe location information from the CDF file and saved the recovered intensities mapped to unique probe indices in TXT files. Of note, in each TXT file there are 1 208 516 probes since other probes do not have corresponding probeset annotations.

RESULTS

Pixel-level segmentation of contaminated regions on reconstructed microarray images by translational convolutional neural networks

We downloaded a total of 37 724 microarray data from GEO, and reconstructed the images by the described localization identification procedure in the 'Materials and Methods' section (Figure 1). As no ground truth of contamination currently exists for these published microarray data, we normalized the ranges of microarray readout values for each image between 0 and 255, which allowed us to visually inspect each one of them and hand label the defected or contaminated areas. We identified several prominent types of contamination: (1) recognizable large areas of whitening, likely caused by sudden increase in mRNA materials in that area (Figure 1D); (2) recognizable large areas of blackening, likely caused by the material not covering the entire microarray chips (Figure 1D); and (3) artificially induced contaminations such as fingerprints, written characters on the chips (Figure 1D, Supplementary Figure S2). The above types of contaminations are not expected to be corrected by neighboring paired probes, and we are interested in investigating alternative methods based on reconstructed images to detect these regions.

The above reconstructed microarray images and labels of contaminated regions provided ground truth to develop a deep learning model for automatically detecting contaminated regions in a public microarray dataset. An interesting development in the deep learning field is a method that translates an input image to an output image. This translation can be colorization of a black-and-white image, segmentation of regions of interest or changing an image to a specific artistic style (14–16). For example, in a translational network that changes black-and-white images to colored images, the input is an image with a single channel representing pixel intensity, and the output is an image with three channels representing the values for red, green and blue, respectively. In this context of segmenting defected regions in microarray, the input is a microarray chip's image intensity, and the translated image is the segmentation mask of the defected region.

A variety of such pixel-value translational deep learning architectures have been developed, including fully convolutional networks (FCNs) and U-Net (14,17). Both architectures share a common global layout with an encoder that extracts hidden features from the original image and a decoder that reconstructs the translated image. In FCNs, the decoder is relatively shallow, usually with several layers of deconvolution. In U-Net, the encoder and decoder form a symmetrically structured convolutional neural network. We deployed the U-Net structure in this study (Figure 2A).

We carried out five-fold cross-validation to evaluate whether we are able to identify the contaminations. Briefly, the data were partitioned into five parts, and in each round, four out of the five parts were used as the training set, and the other part was used as the test set to evaluate the ability of the models in recovering ground truth labels (Figure 2B). As this is a segmentation task, we evaluated the performance using Sørensen–Dice coefficient, which is the overlap between the predicted region and the ground truth over the union of the two. The random baseline Dice coefficient for this dataset is 0.00214, i.e. the percentage of contaminated regions by the hand label. The model achieved an average Dice coefficient of 0.61703 in the cross-validation, 288 times over random baseline (Figure 2C–E).

Compared to the initial bounding box labeling, predictions made by deep learning cover the contaminated regions more accurately (Figure 2D). Further inspection of the results shows that the imperfect Dice score mainly comes from overprediction, and such overprediction almost invariably identifies defected regions that were initially missed by human labeling. We randomly surveyed 500 such new predictions, manually rescrutinized the images and found 455 out of them are indeed correct predictions and can be seen by human eyes. For the other 45 images on which we could not visually observe defects, their predicted masks showed that only a tiny part of them have defects. This observation supports the application of the models to be used to detect contaminated regions in microarray data, and correct the expression data using this information.

Expression data in contaminated regions detected by reconstructed images show less coordinance within probesets

We further explored whether expression patterns of the contaminated regions and contaminated microarrays showed less coordinance compared to non-contaminated ones. To do so, we made predictions of defective areas for all reconstructed microarray images through cross-validation procedure. Specifically, when a microarray image belongs to the test set, we record its predicted contaminated areas for follow-up experiment.

To examine the coordinance of probes within probesets, we must consider several important factors in microarray design. First, the binding affinity of each probe is different. This will cause probe readouts for the same gene not equivalent, even without any type of noise or contamination. Thus, normalization for each probe is needed. Second, probes of some probesets have naturally larger deviations than others.

Considering the above factors, we first identified all microarray images that are predicted to have no contamination at all. We inferred the RMA transformed expression level of each probeset using software R for all microarrays. We next calculated the mean expression value of each probe across all microarrays without predicted or labeled

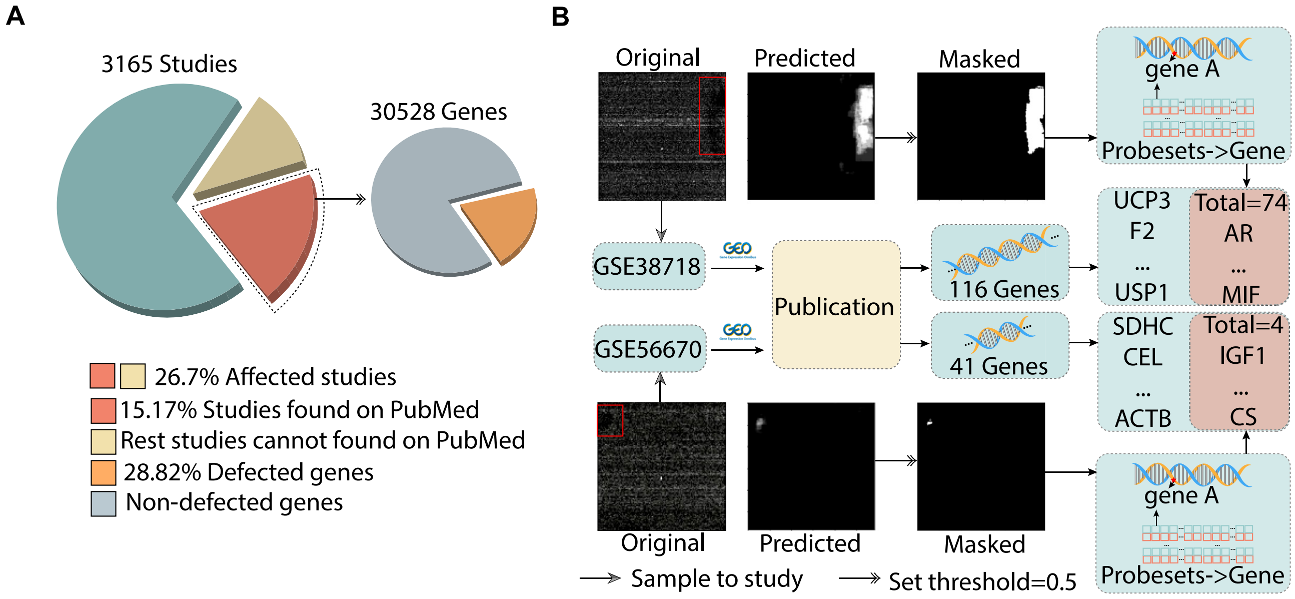

Figure 3. Post-experiment evaluation. (**A**) Proportions of affected studies and genes. 26.7% of the total of 3165 studies were affected. 15.17% ($n = 480$) of the associated publications can be downloaded through PubMed. These 480 studies mentioned 30 528 genes in total, and 28.82% of them were located in the defective area. (**B**) Overview of literature mining and examples of GSM948634 and GSM1366662. We identified their corresponding series number and publication on the GEO website and grepped all genes mentioned in the text. Prediction results were used to filter out contaminated genes.

contamination, X_{mean}, and used X_{mean} as a reference to correct the affinity biases across probes by vector element-wise division (Figure 2C). That is, we created new expression values for each probe of datasets by dividing the original value against the corresponding value in the reference matrix.

The above normalized data allowed us to carry out one type of evaluation. First, for specific genes, we compare whether the ones that are partly or wholly covered in contaminated regions show higher intragene probeset-level variance compared to the ones that are completely in non-contaminated regions. Specifically, suppose we have a total of N microarray images and a total of P genes, and each gene j corresponds to n_j probesets, whose expression level is denoted as x_{ijk}, where i refers to the index of the image and k refers to the index of the probeset. For each image i, we calculated the standard error of all probesets in a gene:

$$\text{s.d.}_{ij} = \sqrt{\frac{1}{n_j - 1}\sum_{k=1}^{n_j}\left[\frac{x_{ijk}}{X_{\text{mean}\,jk}} - \left(\frac{\sum_{k=1}^{n_j}\frac{x_{ijk}}{X_{\text{mean}\,jk}}}{n_j}\right)\right]^2}.$$

We examined whether overall contaminated areas are more likely to show higher deviation than uncontaminated areas at the image level. Thus, for each image that contains contamination, we grouped the genes into {s.d.contaminated} and {s.d.uncontaminated}, and compared the overall distribution of the two. If a gene has at least one probe falling into the contaminated regions, it is grouped into {s.d.contaminated}. We then calculated the above standard error across all probesets in a gene. We found out of 96.0% of the images, {s.d.contaminated} showed higher standard error overall than the {s.d.uncontaminated} group, supporting that such areas of contamination are indeed a problem for microarray data analysis. This result also corroborated the validity of the deep learning models and necessity to correct these errors in microarray data analysis.

Over a quarter of the biological conclusions were drawn based on contaminated regions in microarray studies

We crawled down 480 publications that are directly accessible from PubMed that can be associated with 846 microarray datasets (out of 3165 datasets) predicted to be affected by contamination. This is 15.17% of the total affected studies we were able to download an associated publication. This is a tiny fraction of the entire publication repertoire based on microarray, but can reflect the overall biases.

We text-mined the genes that were mentioned in each of these papers and discovered an accumulative 30 528 genes mentioned in total. We found 8797 of the cases (28.82%) are within the contaminated areas (Figure 3A). This indicates around a quarter of the publications (26.73%) are using microarray data relying on data with defects, and around one quarter of conclusions of them were drawn based on contaminated regions (28.82%). Importantly, only 2.78% of genes are affected by the contamination. If contamination does not affect what we discover in biology, we would expect around 2.78% of the discoveries made for these genes. However, 28.82% of the reported discoveries of genes came out of the 2.78% of the genes affected by contamination. Thus, the 'discoveries' made on contaminated regions are disproportionately high, strongly supporting that many conclusions were reporting contamination rather than biology.

We would like to highlight the damaging effects of such contaminations for our understanding of science using two examples. By no means that this serious issue is limited to the examples we listed here, again it is widespread and estimated to affect a quarter of the studies. Sample

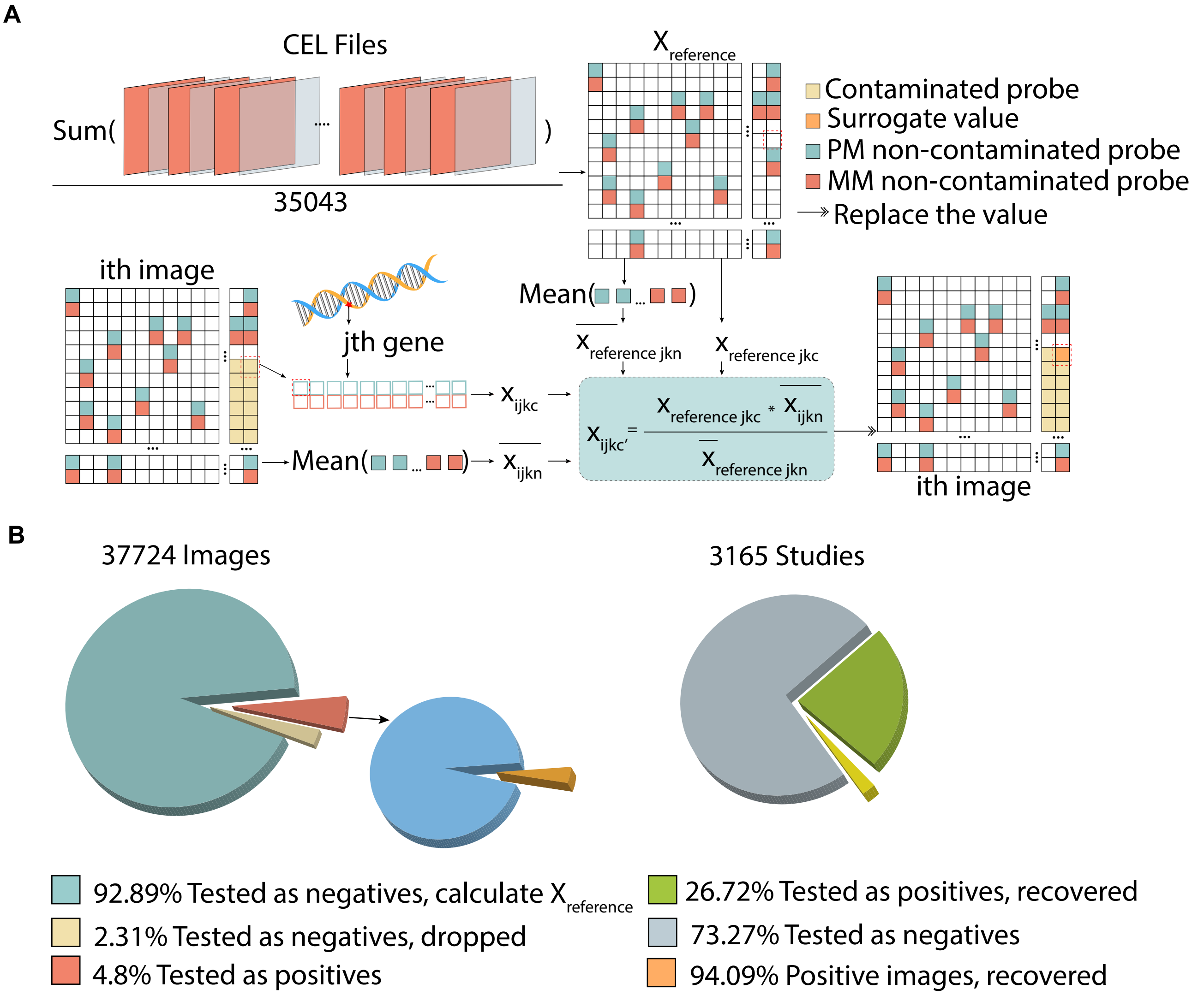
A
CEL Files
Sum(
)
35043
X reference
Contaminated probe
Surrogate value
PM non-contaminated probe
MM non-contaminated probe
Replace the value
ith image
jth gene
Mean(
)
X reference jkn
X reference jkc
x ijkc
Mean(
)
x ijkn
x ijkc' = X reference jkc * x ijkn / X reference jkn
ith image
B
37724 Images
3165 Studies
92.89% Tested as negatives, calculate X reference
2.31% Tested as negatives, dropped
4.8% Tested as positives
26.72% Tested as positives, recovered
73.27% Tested as negatives
94.09% Positive images, recovered

Figure 4. Recovery of contaminated CEL files. (**A**) Recovery algorithm. (**B**) Proportion of images tested as positive and images recovered. Among the 37 724 samples, 1810 samples have contaminations. One thousand seven hundred three out of 1810 samples were recovered. For the remaining 35 914 negative samples, 35 043 samples were used to calculate $X_{\text{reference}}$.

GSM948634 (GEO sample ID) belonging to the study GSE38718 (GEO series ID) has defects as shown in Figure 3B (18). There are 116 genes mentioned in the associated publication, but 74 of them are in contaminated areas. Sample GSM1366662 in study GSE56670 has defects as shown in Figure 3B (19). There are 41 genes mentioned in the paper and 4 of them are in contaminated areas. For another example, study GSE38718 was cited by a publication (18), in which researchers found that ‘genes involved in lipids storage, such as SCD, GPAM, and PPARG (Table 3), were significantly upregulated with aging in women, but only minor or nonsignificant changes were observed in men’. However, the expression levels of all three genes in the young man’s sample (Muscle_Young Men_Rep5, GSM948634) were affected by contaminations. The contamination is misleading for comparing transcript levels between young men and older men. This highlights how published conclusions are erroneously made according to microarray data contamination, and suggests the necessity of an algorithm to scan and potentially correct the microarray data.

Software can detect defects and recover 99% of the contaminated microarray signals by surrogate probe-based imputation

We sought to correct the contaminated microarray experiments and provide it as a resource for the community for the follow-up studies and rescrutinization of the discoveries generated by previous studies (Figure 4). Toward this goal, we read the probe intensities from the CEL files of 35 043 samples without labeled or predicted contamination or *NaN* and calculated the average intensity matrix $X_{\text{reference}}$. Let us suppose we have probe intensity x_{ijkc} that resides in a contaminated region, where *i* refers to the *i*th image, *j* refers to the *j*th gene, *k* refers to the *k*th probeset and *c* refers to the *c*th probe in the probeset. We identify all the probes for this probeset that reside in the non-contaminated region, and calculate the surrogate value for kc by

$$x_{ijkc'} = \frac{X_{\text{reference } jkc} \times x_{ijkn}}{X_{\text{reference } jkn}} \quad \{n \in \text{non-contaminated}\}\,.$$

Essentially, the values of the contaminated probes are replaced by the non-contaminated ones normalized by the reference of the average matrix obtained from the non-contaminated images. Because for each probe set the probes are designed to be scattered across the images, 99.09% of the gene-associated probes in the contaminated regions can be replaced using this approach. The remaining 0.91% will be left as missing values, as in traditional microarray data analysis; rare missing values can be imputed by algorithms such as KNNimpute (20).

We provided the corrected microarray data, the hand-labeled contamination arrays and associated genes and the predicted contaminations and genes as a public resource at https://osf.io/g4qxu/?view_only=3aaf0f0469744e54befbc4f86143ab47 on Open Science Framework for researchers to revisit and reanalyze the published experimental results. Overall, 26.7% of the published microarray studies are affected by this correction. We also provided the software (see the 'Code Availability' section) to scan for such contaminations, so that future studies can avoid similar errors.

DISCUSSION

Microarray data have been used in >100 000 publications to date. Numerous important genes and significantly enriched pathways have been reported based on these types of data. Yet, this retrospective reconstruction of the original microarray images and inspection efforts reveal that ∼26.7% of the published microarray datasets are affected by contamination. Among this set, 28.8% of the conclusions on genes have been drawn on the defective areas. This spurred us to reflect how, as end users, we should use, scrutinize and trust this and other genomic technologies.

These defects include possible chip defects, uncovered areas and human-induced artifacts such as writings and fingerprints on the microarray chips. More alarmingly, these defects are significantly associated with important gene-wise conclusions and discoveries reported in the literature. To correct these errors and screen the trustable results based on microarray data retrospectively, we developed a deep learning framework that reconstructs images, and identifies and corrects defects. We applied it to all microarray images we have at hand, and validated the predictions by both expression patterns and visual inspection. These predictions are now provided as a community resource, which we think will become a reference to facilitate retrospective analysis and scrutinization of these datasets. The model for reconstruction and prediction is certainly directly useful for scrutinizing new, ongoing microarray experiments.

Despite a tremendous number of studies relying on microarray data, no approach has been streamlined to detect defects among them at the original chip imaging level. As a matter of fact, end users such as biologists or bioinformaticians do not typically have access to such images. Utilizing a set of Affymetrix development kits, we managed to reconstruct these images by probe level readouts. For the first time, we revealed a surprising fraction of the studies (26.73%) affected by such studies, indicating many of the discoveries we now see are simply results of such contaminations. For many other genomics tools, such as RNA-sequencing and single cell sequencing, what the end users obtain are similarly processed numbers, rather than original readout signals by machines. The result of this study suggests that careful scrutinization of the biases in large-scale datasets generated by genomic tools is perhaps in general necessary.

DATA AVAILABILITY

These source data are available at https://zenodo.org/record/5236430#.YSPxpVNKi3I (doi:10.5281/zenodo.5236430).

CODE AVAILABILITY

The API source is available at https://github.com/GuanLab/Microarray. Image annotation tool is available at https://github.com/GuanLab/owl.

SUPPLEMENTARY DATA

Supplementary Data are available at NARGAB Online.

ACKNOWLEDGEMENTS

Author contributions: Study design: Y.G. Experiment/coding: Y.Q., X.C. and Y.G. Figures: D.Y. and Y.Q. Manuscript: Y.G. All authors read and approved the manuscript.

FUNDING

National Institutes of Health [R35GM133346].

Conflict of interest statement. None declared.

REFERENCES

1. Kononen,J., Bubendorf,L., Kallioniemi,A., Bärlund,M., Schraml,P., Leighton,S., Torhorst,J., Mihatsch,M.J., Sauter,G. and Kallioniemi,O.P. (1998) Tissue microarrays for high-throughput molecular profiling of tumor specimens. *Nat. Med.*, **4**, 844–847.
2. Irizarry,R.A., Warren,D., Spencer,F., Kim,I.F., Biswal,S., Frank,B.C., Gabrielson,E., Garcia,J.G.N., Geoghegan,J., Germino,G. *et al.* (2005) Multiple-laboratory comparison of microarray platforms. *Nat. Methods*, **2**, 345–350.
3. DeRisi,J., Penland,L., Brown,P.O., Bittner,M.L., Meltzer,P.S., Ray,M., Chen,Y., Su,Y.A. and Trent,J.M. (1996) Use of a cDNA microarray to analyse gene expression patterns in human cancer. *Nat. Genet.*, **14**, 457–460.
4. Silver,J.D., Ritchie,M.E. and Smyth,G.K. (2009) Microarray background correction: maximum likelihood estimation for the normal–exponential convolution. *Biostatistics*, **10**, 352–363.
5. Sun,Z., Chai,H.S., Wu,Y., White,W.M., Donkena,K.V., Klein,C.J., Garovic,V.D., Therneau,T.M. and Kocher,J.-P.A. (2011) Batch effect correction for genome-wide methylation data with Illumina Infinium platform. *BMC Med. Genomics*, **4**, 84.
6. Lazar,C., Meganck,S., Taminau,J., Steenhoff,D., Coletta,A., Molter,C., Weiss-Solís,D.Y., Duque,R., Bersini,H. and Nowé,A. (2013) Batch effect removal methods for microarray gene expression data integration: a survey. *Brief. Bioinform.*, **14**, 469–490.

7. Luo,J., Schumacher,M., Scherer,A., Sanoudou,D., Megherbi,D., Davison,T., Shi,T., Tong,W., Shi,L., Hong,H. *et al.* (2010) A comparison of batch effect removal methods for enhancement of prediction performance using MAQC-II microarray gene expression data. *Pharmacogenomics J.*, **10**, 278–291.
8. Mezencev,R. and Auerbach,S.S. (2020) The sensitivity of transcriptomics BMD modeling to the methods used for microarray data normalization. *PLoS One*, **15**, e0232955.
9. Zindler,T., Frieling,H., Neyazi,A., Bleich,S. and Friedel,E. (2020) Simulating ComBat: how batch correction can lead to the systematic introduction of false positive results in DNA methylation microarray studies. *BMC Bioinformatics*, **21**, 271.
10. Okoniewski,M.J. and Miller,C.J. (2006) Hybridization interactions between probesets in short oligo microarrays lead to spurious correlations. *BMC Bioinformatics*, **7**, 276.
11. Wu,Z. and Irizarry,R.A. (2005) Stochastic models inspired by hybridization theory for short oligonucleotide arrays. *J. Comput. Biol.*, **12**, 882–893.
12. Irizarry,R.A., Wu,Z. and Jaffee,H.A. (2006) Comparison of Affymetrix GeneChip expression measures. *Bioinformatics*, **22**, 789–794.
13. Ritchie,M.E., Silver,J., Oshlack,A., Holmes,M., Diyagama,D., Holloway,A. and Smyth,G.K. (2007) A comparison of background correction methods for two-colour microarrays. *Bioinformatics*, **23**, 2700–2707.
14. Ronneberger,O., Fischer,P. and Brox,T. (2015) U-Net: convolutional networks for biomedical image segmentation. In: *Lecture Notes in Computer Science*. pp. 234–241.
15. Gatys,L.A., Ecker,A.S. and Bethge,M. (2016) Image style transfer using convolutional neural networks. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
16. Cheng,Z., Yang,Q. and Sheng,B. (2015) Deep colorization. In: *2015 IEEE International Conference on Computer Vision (ICCV)*.
17. Long,J., Shelhamer,E. and Darrell,T. (2015) Fully convolutional networks for semantic segmentation. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440.
18. Liu,D., Sartor,M.A., Nader,G.A., Pistilli,E.E., Tanton,L., Lilly,C., Gutmann,L., IglayReger,H.B., Visich,P.S., Hoffman,E.P. *et al.* (2013) Microarray analysis reveals novel features of the muscle aging process in men and women. *J. Gerontol. A Biol. Sci. Med. Sci.*, **68**, 1035–1044.
19. Killian,J.K., Miettinen,M., Walker,R.L., Wang,Y., Zhu,Y.J., Waterfall,J.J., Noyes,N., Retnakumar,P., Yang,Z., Smith,W.I., Jr. *et al.* (2014) Recurrent epimutation of SDHC in gastrointestinal stromal tumors. *Sci. Transl. Med.*, **6**, 268ra177.
20. Troyanskaya,O., Cantor,M., Sherlock,G., Brown,P., Hastie,T., Tibshirani,R., Botstein,D. and Altman,R.B. (2001) Missing value estimation methods for DNA microarrays. *Bioinformatics*, **17**, 520–525.