ONLINE METHODS

1 METHODS

1.1 MODEL OVERVIEW

For CCIGAN, we use SPADE residual blocks (Park et al., 2019) as our generative backbone and DCGAN's discriminator's architecture (Radford et al., 2015). Park et al. (2019) have shown SPADE to be an effective way to inject conditioning into a generative model. The SPADE normalization layer serves as a replacement for previous layer normalization techniques. Instead of learning a universally shared per channel affine transformation, like in Batch Normalization (Ioffe & Szegedy, 2015) or Instance Normalization (Ulyanov et al., 2016), SPADE learns to predict affine transformations based on segmentation maps; each feature is uniquely transformed based on its cell type, size, and neighboring cells. The ability for SPADE to modulate activations based on the context of adjacent cell segmentations allows the network to effectively model the behaviors and interactions of cells. The input of CCIGAN is a noise vector $\boldsymbol{z} \in \mathbb{R}^{128}$ and a segmentation map $\mathbf{S}$. f denotes a linear layer $\mathbb{R}^{128} \mapsto \mathbb{R}^{2048}$. $\mathbf{R}^i$ are feature map representations from SPADE resblocks and $\mathbf{X}$ denotes the final output of M cell expressions. Below, each layer's output dimensions are given next to their respective equations. Further details such as kernel size, activation functions, training regimen, and model interpretability are given in Supplementals 1.

$$\mathbf{Z} \in \mathbb{R}^{(128,4,4)} = f(\boldsymbol{z}) \quad (1)$$

$$\mathbf{R}^1 \in \mathbb{R}^{(128,8,8)} = \text{SPADE\_RESBLK}(\mathbf{Z}, \mathbf{S}) \quad (2)$$

$$\mathbf{R}^2 \in \mathbb{R}^{(128,16,16)} = \text{SPADE\_RESBLK}(\mathbf{R}^1, \mathbf{S}) \quad (3)$$

$$\mathbf{R}^i \in \mathbb{R}^{(128,32,32)} = \text{SPADE\_RESBLK}(\mathbf{R}^2, \mathbf{S}) \quad (4)$$

$$\mathbf{R}^4 \in \mathbb{R}^{(128,64,64)} = \text{SPADE\_RESBLK}(\mathbf{R}^3, \mathbf{S}) \quad (5)$$

$$\mathbf{X} \in \mathbb{R}^{(M,64,64)} = \text{ATTENTION}(\mathbf{R}^4, \mathbf{S}) \quad (6)$$

1.2 ATTENTION MODULE

Our architectural contribution is a protein marker dependent attention module in the final output layer. The goal of the attention module is to condition the final output of a channel on a protein marker m and $\mathbf{S}$'s cell types. For example the protein marker, pan-keratin m_{pk}, is expressed exclusively in tumor cells but not in other cells. Appropriately, an attention mechanism should attend to tumor cells and ignore irrelevant cells in $\mathbf{S}$ for m_{pk}. To replicate a marker searching for specific cell types that express it, we define a learned persistent vector for each marker denoted by $\boldsymbol{s}_{m \in M} \in \mathbb{R}^8$ that undergo a series of operations with the final feature map representation attending to m's specific cell types. It is also worthwhile to note that these persistent vectors $\boldsymbol{s}_m$ offer a degree of model interpretability that mimic real world markers. The current input dimensions to the attention module are $\mathbb{R}^{(128,64,64)}$ following the last resblock $\mathbf{R}^4$ and m indexes from $1, .., M$.

$$\mathbf{O} \in \mathbb{R}^{(M,64,64)} = \text{CONV2D}(\mathbf{R}^4) \quad (7)$$

$$\mathbf{C} \in \mathbb{R}^{(KM,64,64)} = \text{CONV2D}(\mathbf{O}) \quad (8)$$

$$\mathbf{C}_i \in \mathbb{R}^{(K,64,64)} = \text{SPADE}(\mathbf{C}_{K(i-1):Ki,:,:}, \mathbf{S}) \quad (9)$$

$$\mathbf{A}_i \in \mathbb{R}^{(|\mathbf{s}| \times K,64,64)} = \mathbf{C}_i \otimes \boldsymbol{s}_m \quad (10)$$

$$\mathbf{B}_i \in \mathbb{R}^{(1,64,64)} = \sigma(\text{CONV2D}(\mathbf{A}_i)) \quad (11)$$

$$\mathbf{X} \in \mathbb{R}^{(M,64,64)} = \mathbf{O} + \mathbf{B}_{1,..,M} \quad (12)$$

After $\mathbf{R}^4$, a bottleneck convolution is applied to match the original data's dimension as $\mathbf{O}$ (step 1), which is used in a residual manner with the final output. Intuitively at this stage, $\mathbf{O}$'s feature maps resemble the target $\mathbf{Y}$, but we wish to further refine the output channels. We convolve $\mathbf{O}$ into MK channeled features for each protein marker where $K = 8$. Considering each $\mathbf{C}_i$ where $i \in \{1, ..., M\}$ as a group of K channels, the model spatially adaptive normalizes each $\mathbf{C}_i$ and computes an outer product with the corresponding persistent vector $\boldsymbol{s}_i$ and $\mathbf{C}_i$. The resulting matrix is flattened and convolved (with a kernel size of 1 on the pixel level) from $\mathbf{A}_i \in \mathbb{R}^{(|\mathbf{s}| \times K,64,64)} \mapsto \mathbb{R}^{(1,64,64)}$ followed by a sigmoid $\sigma(\cdot)$ activation. Lastly, the attentions $\mathbf{B}_{1,..,M}$ are added to $\mathbf{O}$ to obtain the output $\mathbf{X}$.

Initially, the model has no priors over the interaction of protein markers and cell types. The proposed outer product attention layer (outer product and 1×1 convolution) excels at modeling these relationships and interactions between specific markers and cell types. By using an outer product, the model forces attention at a pairwise pixel level comparison for all combinations of elements between $\boldsymbol{s}_m$ and $\mathbf{A}_i$. As training progresses, both the learned features over segmentation patches and the learned persistent vectors $\boldsymbol{s}_m$ improve, in turn allowing the learned 1×1 convolution to reason about positive or negative relationships from the pairwise pixel combinations.

2 DATA AND DATA PROCESSING

We trained CCIGANon two types of cell data, MIBI-TOF (multiplexed ion beam imaging by time-of-flight) and t-CyCIF (tissue cyclicimmunofluorescence) data. While these two types of multiplexed image data were obtained through different procedures, they share structural similarities.

Multiplexed cell images display multiple protein marker expression levels. These images are represented as high dimensional tensors $\mathbf{T} \in \mathbb{R}^{(M,H,W)}$, M being the number of markers, H being the height of the image, and W being the width. Each of these markers $m \in \{1, ..., M\}$, are given as a channel taking on real values continuous in $[0, 1]$ at each (x, y) coordinate, indicating the expression level at a given protein. Protein markers' particular expression levels (either separately or in conjunction with other protein markers) demarcate different cellular subtypes and furthermore, are indicative of the functional properties of a cell. By simultaneously imaging over multiple protein markers, multiplexed images are able to identify cell type as well as provide detailed information of sub-cellular structure, cell neighbors, and interactions in the tumor microenvironment across these different marker settings.

2.1 MIBI-TOF

MIBI-TOF images are represented in $\mathbf{T} \in \mathbb{R}^{(M,2048,2048)}$. These images are then further processed at a cell by cell basis into $\mathbf{Y} \in \mathbb{R}^{(M,64,64)}$ patches, where a cell is at the center of the patch along with its neighbors. Next, we construct semantic segmentation maps $\mathbf{S} \in \mathbb{R}^{(C+1,64,64)}$, where a vector $\mathbf{S}_{:,i,j}$ is one-hot encoded based on a cell type $C = 17$, and the $C + 1$-th channel denotes empty segmentation space. The data is train-test split at a 9:1 ratio at the MIBI-TOF image level to avoid cell neighborhood bias.

Data obtained through MIBI-TOF characterized tissue samples were collected from triple-negative breast cancer (TNBC) patients. MIBI-TOF images over 36 protein markers, but $M = 24$ markers were used in our training. A description of the technology and full list of these markers is given in Supplements 2.1.

2.2 T-CYCIF

T-CyCIF (tissue cyclicimmunofluorescence) images of primary lung squamous cell carcinoma are represented in $\mathbf{T} \in \mathbb{R}^{(M,H\approx 12000,W\approx 14000)}$. Segmentation patches are constructed in an similar fashion. The main salient difference from MIBI is the processed patch size $\mathbf{Y} \in \mathbb{R}^{(M,128,128)}$, and the segmentation patch size $\mathbf{Y} \in \mathbb{R}^{(C+1,128,128)}$. This was intentionally done to demonstrate the scalability of CCIGAN. 44 markers were imaged in t-CyCIF, however we excluded background cell markers to yield $M = 37$ markers. A description of the technology and full list of these markers is given in the S2.2.

Unlike MIBI, a significant amount of data processing was done in order to analyze the data. Full treatment of data is given in Supplements 2.2.

3 EVALUATION

To conduct fair experiments, all models were optimized, tuned, and set with similar parameters. They were also taken from their official online implementations and trained for 120 epochs or until convergence (max 150). CCIGAN is identical to our designed SPADE comparison baseline with the exception of the attention module.

3.1 IMAGE EVALUATION AND RECONSTRUCTION

First, we use the following evaluation metrics in order to compare with baseline results: adjusted L_1 and MSE score, L_1 and MSE score, structural similarity (SSIM) index (Wang et al., 2004) and cell based mutual information (MI) shown in Table 1. Bolded scores indicate the best scores. Equations and motivation are given in S3.1.

Metrics	CCIGAN	SPADE	Pix2PixHD	CycleGAN
Adjusted L_1 Score	**0.613**	**0.618**	0.875	4.745
L_1 Score	**0.594**	**0.602**	0.745	3.959
Adjusted MSE Score	**0.026**	0.031	0.061	1.841
MSE Score	**0.026**	0.031	0.055	1.523
SSIM	**0.810**	**0.802**	0.709	0.394
Cell Mutual Information	**10.46**	10.25	9.26	7.96

Table 1: Comparison of conventional reconstruction metrics between different models.

Three evaluation metrics were then used to conduct experiments and validate the trained model's utility in generating biologically meaningful cellular proteins in the tumor microenvironment and ability to recapitulate and *quantify* previously established biological phenomena. Each subsection provides additional relevant information.

3.2 CENTER OF MASS (COM)

For a generated cell image, its weighted centroid, or center of mass, is the mean position of all the points in the cell weighted by a particular channel expression. Given a cell image $\boldsymbol{X} \in \mathbb{R}^{(H,W)}$, with indices of the segmented cell $V \subseteq \{1,\ldots,H\} \times \{1,\ldots,W\}$, the COM $\bar{\boldsymbol{p}} = (\bar{x}, \bar{y})$ is defined as $\bar{x} = \frac{\sum_{(x,y)\in V} x\boldsymbol{X}_{x,y}}{\sum_{(x,y)\in V} \boldsymbol{X}_{x,y}}$ and $\bar{y} = \frac{\sum_{(x,y)\in V} y\boldsymbol{X}_{x,y}}{\sum_{(x,y)\in V} \boldsymbol{X}_{x,y}}$.

For example, in the PD-1/PD-L1 experiment, we compute the COM of the CD8 T cell (cell of interest) weighted by PD-1 expression, given as $\bar{\boldsymbol{p}}^{\text{CD8}}$, and the COM of all tumor cells weighted by PD-L1 expression, given as $\bar{\boldsymbol{p}}^{\text{Tumor}}$. Since T cells located within the tumor microenvironment often have upregulated expression of PD-1, we assume that $\bar{\boldsymbol{p}}^{\text{CD8}}$ should have the roughly the same PD-L1 COM of all its surrounding tumor cells $\bar{\boldsymbol{p}}^{\text{Tumor}}$. The center of mass score is defined below as the relative distance between $\bar{\boldsymbol{p}}^{\text{CD8}}$ and $\bar{\boldsymbol{p}}^{\text{Tumor}}$, where N is defined as the number of patches:

$$\text{COM}_{\text{projection}} = \frac{1}{N}\sum_{i=1}^{N} \|\bar{\boldsymbol{p}}_i^{\text{CD8}} - \text{Proj}_{\text{CD8}}(\bar{\boldsymbol{p}}_i^{\text{Tumor}})\|_2 \tag{13}$$

The projection function $\text{Proj}(\cdot)$ is used to project $\bar{\boldsymbol{p}}^{\text{Tumor}}$ onto the CD8 T cell to ensure the expected COM of the tumor cells is inside of the CD8 T cell. As a reference we choose a random position $\bar{\boldsymbol{p}}^{\text{Random}}$ in the CD8 T cell (PD-1) which replaces $\bar{\boldsymbol{p}}^{\text{CD8}}$ in Eq. 13 and compute the random COM score to show the effectiveness of the result. An example illustration is given in the figure in Supplements section 3.1.

3.3 EARTH MOVER'S DISTANCE

Earth Mover's Distance (EM) is a measurement of the dissimilarity between two distributions. For our Pan Keratin/CD8 experiment, we used EM to evaluate the shift in protein localization between CD8 T cells and tumor cells. EM can be generalized to experiments involving other cell-cell interactions, but for the purposes of clarity, below we describe our EM approach in the context of the Pan Keratin/CD8 experiment.

For a segmentation map, we add T tumor cells around one CD8 T cell. The COM for the t-th tumor ($t \in \{1, ..., T_i\}$) is defined as as $\bar{\boldsymbol{p}}_t^{\text{Tumor}}$. Similarly, the PD-1 COMs of the CD8 T cell by adding the t-th tumor is defined by $\bar{\boldsymbol{p}}_t^{\text{CD8}}$. Initially when there are no tumor cells, $\bar{\boldsymbol{p}}_0^{\text{CD8}}$ is the centroid of the CD8 T cell.

We proceed to define vector a vector $\boldsymbol{v}_t$ which points from $\bar{\boldsymbol{p}}_0^{\text{CD8}}$, to the COM of the t-th tumor cell $\bar{\boldsymbol{p}}_t^{\text{Tumor}}$. We also define vector $\boldsymbol{u}_t$ which points from the previous COM $\bar{\boldsymbol{p}}_{t-1}^{\text{CD8}}$ to the current COM $\bar{\boldsymbol{p}}_t^{\text{CD8}}$ of the CD8 T cell. We define θ_t as the angle between $\boldsymbol{v}_t, \boldsymbol{u}_t$.

If $\cos\theta_t > 0$, that is to say if the cosine similarity is positive, the COM of a CD8 T cell $\bar{\boldsymbol{p}}_t^{\text{CD8}}$, moves correctly towards the COM of the added tumor cell $\bar{\boldsymbol{p}}_t^{\text{Tumor}}$. An illustration of the points and vectors is given in the figure in section 3.2 of Supplements.

Formally:

$$\boldsymbol{v}_t = \bar{\boldsymbol{p}}_t^{\text{Tumor}} - \bar{\boldsymbol{p}}_0^{\text{CD8}},\ \boldsymbol{u}_t = \bar{\boldsymbol{p}}_t^{\text{CD8}} - \bar{\boldsymbol{p}}_{t-1}^{\text{CD8}},\ \cos\theta_t = \frac{\boldsymbol{u}_t \cdot \boldsymbol{v}_t}{\|\boldsymbol{u}_t\| \cdot \|\boldsymbol{v}_t\|} \tag{14}$$

After obtaining the directional information, we use EM (Rubner et al., 2000) to evaluate the changes in PD-1 expression of the CD8 T cell. Earth Mover's Distance, which measures the dissimilarity of two distributions, is used in this context to measure the protein localization shifts in PD-1 before and after adding a tumor cell. We consider each cell $\boldsymbol{X}$ in polar coordinates (r, θ) with respect to its centroid, integrate its expression along the radius coordinates, and evaluate the resulting histogram $\text{hist}(\boldsymbol{X})$ along the angle coordinate. The 2nd figure in Supplements section 3.2 shows an example histogram of cells by coordinate location.

This allows for the definition of distance for moving one histogram to another, i.e. $\text{em}(\boldsymbol{X}_i^t, \boldsymbol{X}_i^{t-1}) = d_{\text{EM}}(\text{hist}(\boldsymbol{X}_i^t), \text{hist}(\boldsymbol{X}_i^{t-1}))$, for the generated PD-1 expression of the CD8 T cell $\boldsymbol{X}_i^t$ when adding the t-th tumor cell.

The final EM score is defined as:

$$\text{EM Score} = \frac{1}{\sum_{i=1}^{N} T_i} \sum_{i=1}^{N} \sum_{t=1}^{T_i} \mathbb{1}(\|\boldsymbol{X}_i^t\| > \|\boldsymbol{X}_i^{t-1}\|) \cdot \text{em}(\boldsymbol{X}_i^t, \boldsymbol{X}_i^{t-1}) \cdot \cos\theta_{t,i} \tag{15}$$

where the indicator function $\mathbb{1}(\cdot) = 1$ if and only if $\|\boldsymbol{X}_i^t\| > \|\boldsymbol{X}_i^{t-1}\|$, otherwise $\mathbb{1}(\cdot) = 0$. This ensures that the biological constraint of PD-1 expression increasing as a response to added tumor cells is met. Recall, if $\cos\theta_t > 0$, $\bar{\boldsymbol{p}}_t^{\text{CD8}}$ has moved in the direction of $\bar{\boldsymbol{p}}_t^{\text{Tumor}}$, implying the shift in PD-1 expression is correct, and in turn increases the EM score. By contrast, the EM distance score decreases when $\bar{\boldsymbol{p}}_t^{\text{CD8}}$ moves in the opposite direction.

This can be adapted and used for any two channel protein interactions (such as in the β-catenin and dsDNA experiment for analyzing Wnt pathways).

3.4 Protein Expression and Cell Surface Area Trend Experiments

In this experiment, we used a Student's t-test as the statistical hypothesis test to evaluate the correlations of the protein expression of a specified cell as a function of the area/number of surrounding cells in the specified cell's microenvironment.

Given a generated protein channel $\boldsymbol{X}_i \in \mathbb{R}^{(H,W)}$ and the segmentation map channel for the surrounding cells $\boldsymbol{S}_i \in \mathbb{R}^{(H,W)}$, we compute the total area of the cells $a_i = \sum_{h=1}^{H} \sum_{w=1}^{W} \boldsymbol{S}_i$, and the total expression level of the specified cell $e_i = \sum_{h=1}^{H} \sum_{w=1}^{W} \boldsymbol{X}_i$. We then regress $\{e_i\}_{i=1}^N$ on $\{a_i\}_{i=1}^N$ and assess significance of the slope using a t-test against the null of no change in expression as a function of surrounding cells.

Additional figures of other models and explanations for pan-keratin and CD 8 are given in Supplements 3.4. Additional figures for t-CyCIF experiments are given in Supplements 3.6 and 3.7.

3.5 Tumor Infiltrated and Compartmentalized Microenvironments

CCIGAN was used to compare protein localization in tumor infiltrated versus compartmentalized microenvironments. We used CCIGAN to predict on 200 directly manipulated mixed and non-mixed

tumor environment segmentation patches. For each experiment, we challenge the cell in question with an opposing cell in a microenvironment with macrophages (for example T Cell is challenged with Tumor cell to result in increased PD-1 expression in the T Cell) and use endothelial cells as control cells to show our result has biological significance. Similar to Section 3.4's experimental settings, we compute the average expression of a specific marker for the cells of interest for all patches.

Data resulting from the experiment is located in S3.5. The increase in PD-L1 for the above tumor and macrophage scenarios (S3.5 table 5) indicate that CCIGAN has appropriately captured previously reported biological outcomes and is capable of quantifying these phenomena at single cell levels. Furthermore, the model is adaptable to various different types of tumor architecture depending on its training set to produce different hypothesis testing environments.

4 SEARCH ALGORITHM

Here we provide a modular search algorithm framework to try to discover further cell-cell interactions in other channels. As a high level overview, the algorithm uses CCIGAN to automate and measure a specific cell's change in a specified protein's expression level due to user specified microenvironment changes. The algorithm allows a user to change and specify such changes as cell type, shape, size, and quantity. If the change in expression is greater than a user specified input, then the particular instance is logged. It is important to note that this tool is meant to guide and search for particular interesting interactions and still susceptible to issues such as noisy segmentations.

Algorithm 1: Search Algorithm

Input: Cell segmentation list $\{S_i\}_{i=1}^{n}$, the cell type c_f for the cell of interest, the protein index m that is measured, potential neighbors and their specified types $c_i \in C_{types}$ fixed noise δ, threshold β, a chosen metric d_y, and the generator G.

Randomly chose initial cell index $i_0 \in \{1, ..., n\}$;
Input segmentation $S^{\text{INPUT}} = S_{i_0}$;
Assign cell type S^{INPUT} to c_f;
Mask for the initial cell $U = \sum_{j=1}^{n} S_{i_0}[j, :, :]$;
Generated image $X_0 = G(\delta, S^{\text{INPUT}})$;
Specified channel $M_0 = U * X_0[m]$;
Show S^{INPUT} and M_0;
for $k = 1$ **to** $n - 1$ **do**
    Random index $i_k \in \{1, ..., n\} / \cup_{j=0}^{k-1} \{i_j\}$;
    Random index $c_i \in C_{types}$;
    Assign S_{i_k} to cell type c_i;
    $S^{\text{INPUT}} = S^{\text{INPUT}} + S_{i_k}$;
    $X_k = G(\delta, S^{\text{INPUT}})$;
    $M_k = U * X_k[m]$;
    Show S^{INPUT} and M_k;
    $E_k = d_y(M_{k-1}, M_k)$;
    if $E_k > \beta \sum_{i,j} U_{i,j}$ **then**
        Log Significance;
        if $\sum_{i,j} M_{k-1,i,j} < \sum_{i,j} M_{k,i,j}$ **then**
            Log Increase;
        else
            Log Decrease;

REFERENCES

Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv e-prints*, art. arXiv:1502.03167, Feb 2015.

Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. *CoRR*, abs/1903.07291, 2019. URL `http://arxiv.org/abs/1903.07291`.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv e-prints*, art. arXiv:1511.06434, Nov 2015.

Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99–121, 2000.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance Normalization: The Missing Ingredient for Fast Stylization. *arXiv e-prints*, art. arXiv:1607.08022, Jul 2016.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.