A Landmark-based Common Coordinate Framework for Spatial Transcriptomics Data

Alma Andersson*[1], Žaneta Andrusivová[1], Paulo Czarnewski[1], Xiaofei Li[2], Erik Sundström[2], and Joakim Lundeberg*[1]

[1]Department of Gene Technology, KTH Royal Institute of Technology, Science for Life Laboratory, Solna, Sweden
[2]Division of Neurogeriatrics, Department of Neurobiology, Care Sciences and Society, Karolinska Institutet, Stockholm, Sweden

February 2, 2022

1 Abstract

The increasing amount of spatial transcriptomics data prompts for means to amalgamate observations from distinct experiments, especially attractive is to cast quantities from different sources into a common coordinate framework (CCF) to relate signals across space. We here present a method that enables transfer of information from multiple samples to a reference representing a CCF, and show its utility by analyzing an assortment of real and synthetic data sets.

2 Main

During the last years, there's been an ever increasing amount of interest in the field of spatially resolved transcriptomics, epitomized by its "Method of the Year 2020" award.[1, 2] The field has also experienced a trend of democratization, where techniques have spread beyond the groups originally developing them, a phenomenon reflected by the growing corpus of spatial transcriptomics data. Indeed, some of the spatial transcriptomics techniques have already been adopted as commercial products and embraced by the scientific community, thus facilitating the production of consistent high-quality data by a diverse set of labs. Spatial transcriptomics is also more frequently appearing as a modality of interest in ambitious international initiatives such as the Human Cell Atlas.[3] While quantity is key to delineate the many nuances of transcriptomics data, it also brings with it certain challenges; perhaps most notably the need to integrate observations from multiple sources.

For single cell transcriptomics data the concept of integration is often strongly associated with the process of constructing a shared space based on gene expression, to then embed the data therewithin. However, in contrast to single cell data, spatial transcriptomics data possess an inherent low-dimensional space, being the physical domain from which it's collected. Thus, when building spatial transcriptomics atlases or summarizing larger studies, the idea of integration should be extended beyond elimination of unwanted batch effects. More specifically, it ought to encompass the transfer of data to a shared reference, where observations from different samples can be

related in physical space. Such references are commonly referred to as common coordinate frameworks (CCFs), a concept which Rood et al. thoroughly discuss in their perspective.[4]

Considering this need for spatially aware integration methods, we here present a landmark-based approach to transfer spatial transcriptomics data to a defined reference. Our method relies on Gaussian Process (GP) regression, which previously has been successfully applied to identify spatially variable genes and cell interactions.[5, 6] With this method we seek to overcome both the limitations of traditional alignment methods relying on linear transformations (e.g., rotation and translation) as well as the need for an extensive preexisting reference system to which the data can be registered. We also provide an implementation of our method as a Python package, named "*effortless generic GP landmark transfer*", or *eggplant* for short. To promote easy incorporation into already existing workflows and increase accessibility, *eggplant* is designed to be compatible with the popular analysis framework *scanpy* and its derivatives.[7]

To be more precise, our method focuses on the specific task of transferring observed spatial features from one coordinate system to a given reference system, using a set of shared spatial landmarks. The reference can be any arbitrary structure that represents a spatial domain onto which one seeks to transfer information, see Methods. Meanwhile, we define a spatial landmark as a feature that can be consistently located with fairly high precision across individuals. Samples where spatial landmarks (for brevity, we hereafter drop the prefix "spatial") have been identified will be referred to as "charted". Landmarks can be derived from any – to the tissue – associated information including morphological and molecular structures (e.g., gene expression or protein signals). Furthermore, the charting process can be manual, unsupervised (using computational methods) or a mixture of both; since our method is agnostic to this choice, we consider a deeper discussion regarding landmark annotation and identification to be outside the scope of this work. We also assume that the spatial data has been appropriately normalized and had eventual batch effects corrected for.

Our method is simple in its design and can be described in a few steps, see Figure 1A for a schematic overview. As input it requires charted spatial transcriptomics data containing one or more features of interest (FOI) together with a reference. The reference represents the domain to which the FOI's distribution should be transferred and should also be charted. Next, the domain of the observed data is transformed to make landmark distances match those of the reference. The transformation can either be linear (multiplication with a scaling factor), or non-linear (using thin plate splines) if one suspects a non-homogeneous distortion of the spatial domain. Finally, we formulate a multivariate regression problem where the value of the FOI is considered a function of the distance to respective landmark. We employ a GP framework, commonly described as a distribution over functions, to learn the relationship between feature value and distances. A transfer of any FOI to the reference is seamless once the relationship is established; the function is simply applied to each location in the reference to obtain an estimate of the FOI value. Evidently, multiple samples can be transferred to the same reference, either one-by-one or jointly. Notably, there is no need for alignment or further processing once the samples have been charted. We also provide a strategy to determine a lower bound for the number of landmarks to be used in the process, see Methods.

To demonstrate our method, we first apply it to a set of synthetic data containing eight samples from different time points

in a dynamically changing system. The samples represent the same physical domain, but – like real data – exhibit differences in structure and orientation. Expression from each time point was transferred to a reference with the help of nine landmarks, Figure 1B. For this, and all subsequent analyses, we used non-linear landmark adjustment. This transfer of data to a CCF permits a multitude of downstream analyses, of which we will give two examples below.

The first example focuses on characterization of the system's underlying spatiotemporal dynamics. The dynamical model used to generate the synthetic data is a two-compartment system, in which expression fluctuates according to a set of ordinary differential equations (ODEs). For the sake of simplicity, we assume that the model's structure is known prior to the analysis, and therefore only aim to estimate the model's parameters. The two compartments between which expression varies (C1 and C2) are defined in our reference, allowing us to approximate the total amount of expression in each compartment at every time point. From this aggregated data, we estimated the ODE-model parameters; the corresponding dynamics are shown in Figure 1C where they are also compared to the ground truth values. With the system dynamics established, we could also reconstruct the exchange of expression between the two compartments, see Supplementary Figure 1. In a biological system, this type of flux-analysis could for example elucidate how cells migrate between different regions in a tissue.

In a second example of downstream analysis, we leverage the fact that all data now inhabits the same reference, thus making coordinates comparable between time points. This allows us to perform "spatial arithmetics" from which information about local up-or downregulation of features between time points or conditions can be deduced and tested, see Figure 1D.

For additional evaluation of our method, a second set of synthetic data was generated to assess the influence of the number of landmarks on its performance and compare it to alternative strategies. In short, a non-homogeneous distortion was applied to a collection of spatial observations and associated landmarks, see Supplementary Figures 2A-B. We then assessed how well each strategy could recover the original spatial distribution of the distorted signals, where our approach exhibited the best performance, Supplementary Figure 2C. As expected, for landmark-based approaches, the number of landmarks was positively correlated with performance; however, this trend quickly diminished as the number of landmarks increased.

Having established confidence in our method, we next analyzed several sets of real spatial transcriptomics data. In the first analysis, we examined twelve first generation Spatial Transcriptomics (ST1K) samples of the mouse olfactory bulb (MOB), collected from different individuals and sexes.[8] Here we chose 14 landmarks, identified by morphological cues in the accompanying Hematoxylin and Eosin (HE) images, and charted the corresponding sites in our reference. Having prepared the data, we applied our method and transferred the expression of three genes to the reference: *Nrgn, Apoe* and *Omp*, see Figure 2A and Supplementary Figure 3 and 4. We also assembled "composite" expression profiles for each of the aforementioned genes, allowing us to represent information from all twelve samples jointly. We also conducted a "spatial differential expression analysis" (sDEA) between the three genes, to examine how their local expression differed. The composite representations and the sDEA results are both presented in Figure 2B.

In a second analysis, to show cross-platform

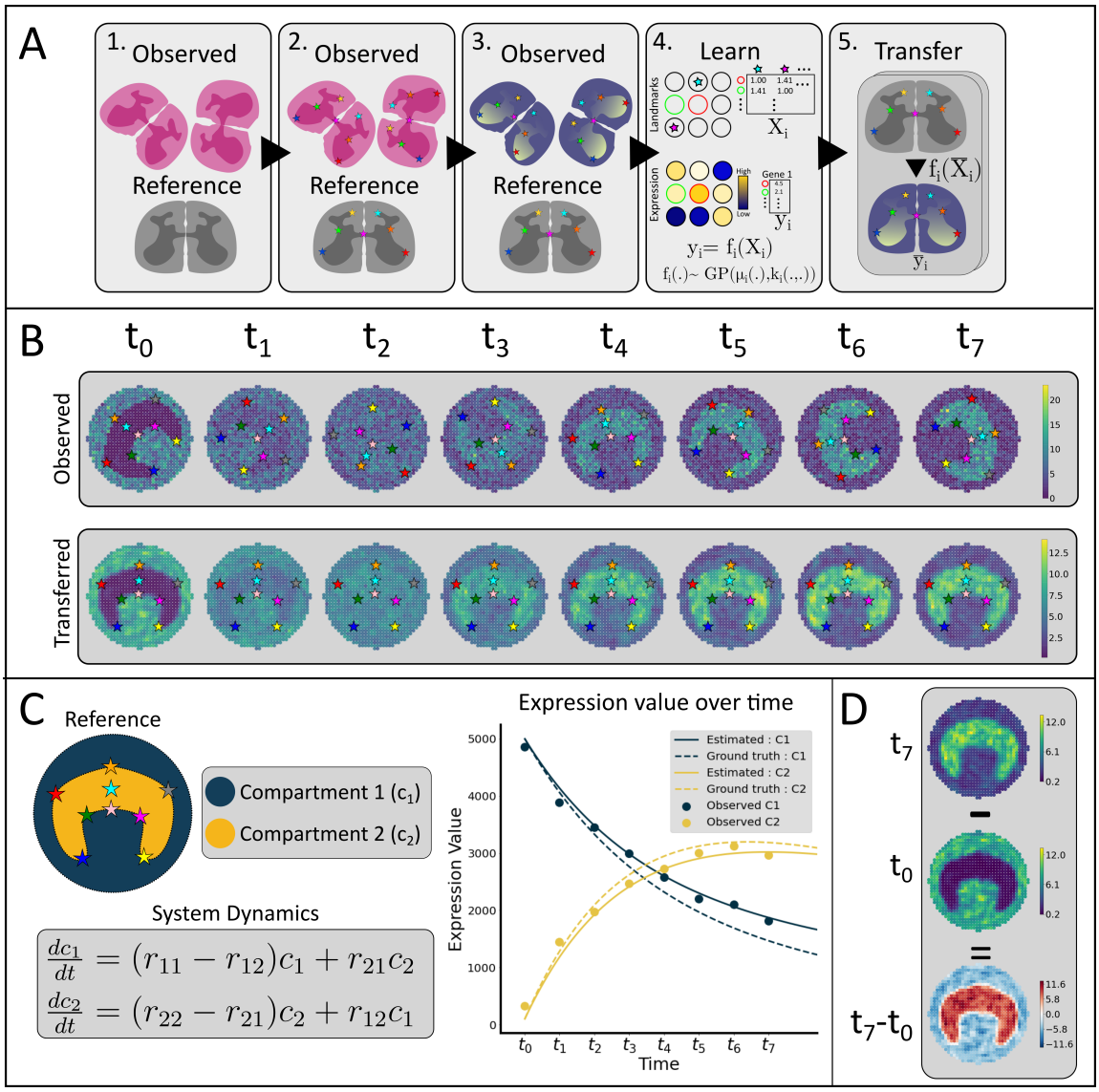

Figure 1: ***A)*** *Schematic overview of the method. 1. We, select a number (here two) of samples representing the same spatial domain together with a reference. 2. We then chart the samples and reference (annotating landmarks). Here, landmarks are represented by colored markers. 3. Next, a feature of interest (FOI) that should be transferred to the reference is selected. 4. We learn the function that relates FOI values to landmark distances by using Gaussian Process (GP) Regression. 5. Finally, the FOI is transferred to reference using the learnt relationship between expression and landmark distances.* ***B)*** *Top : Observed synthetic data across eight different time points. Bottom : Results from transferring the observed data to a reference using our method.* ***C)*** *Spatiotemporal analysis of material (gene expression) transfer between the two compartments in the reference, the graph shows how the expression varies in each compartment as a function of time.* ***D)*** *An example of spatial arithmetics, subtraction of values at t_0 from values at t_7 shows local up-and downregulation of the feature between the two time points.*

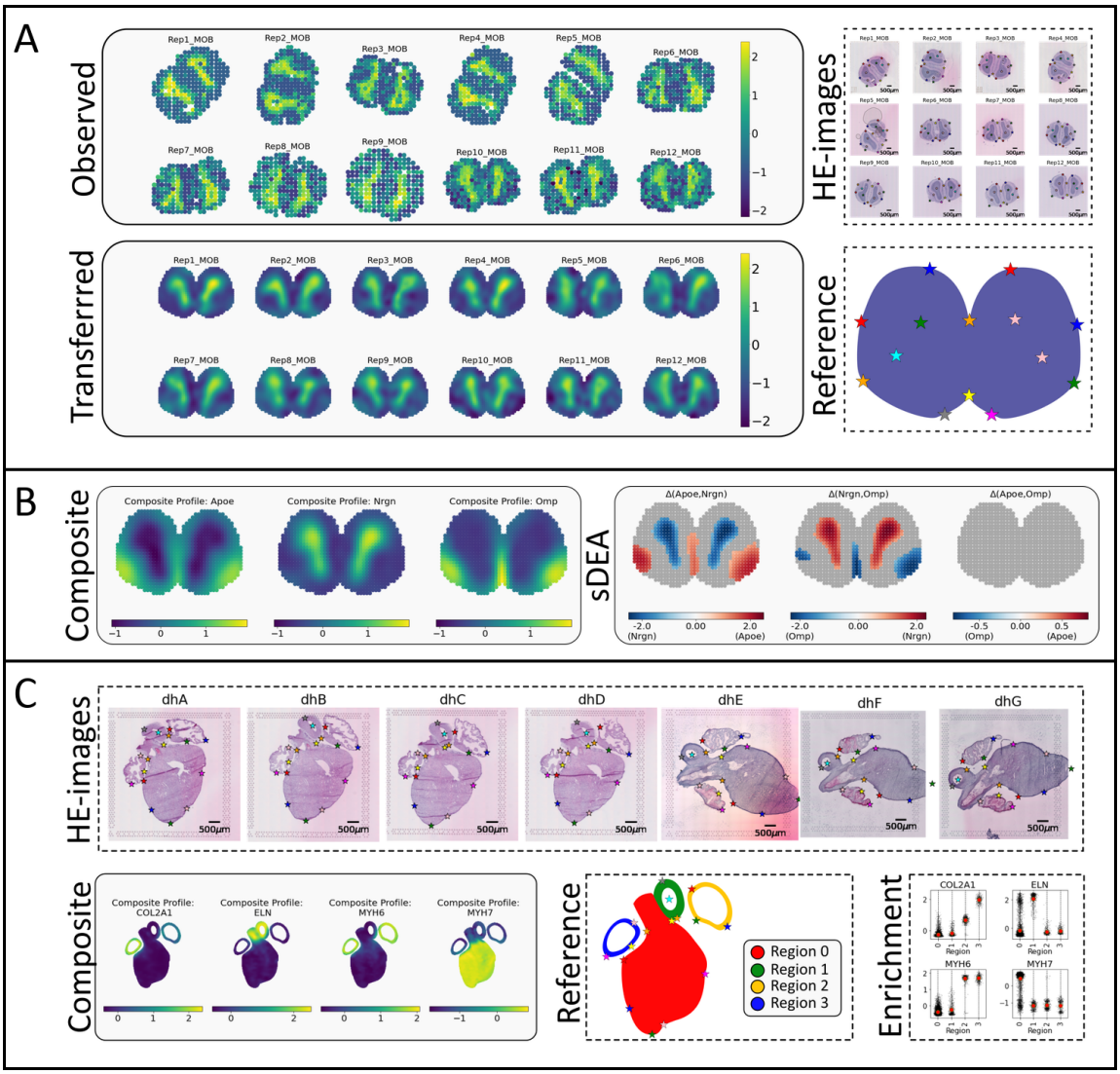

Figure 2: ***A)*** *Top left: observed spatial gene expression of Nrgn in the mouse olfactory bulb (MOB) sections (n=12). Top right : charted HE-images of the MOB sections, landmarks are indicated by colored markers, for a larger image see Supplementary Figure 10. Bottom left: results from transferring the observed MOB data to a common reference. Bottom right : the charted reference to which the MOB expression data is transferred.* ***B)*** *Left : Composite profiles for each of the three genes Apoe, Nrgn and Omp. The composite expression profiles are formed by computing the location-wise mean across all twelve MOB sections, see Methods. Right : spatial differential expression analysis (sDEA) between the three genes, see Methods. Gray areas indicate locations where there's no differential expression between the two compared genes. At locations with differential expression, the values for comparison $\Delta(g_1, g_2)$ are obtained by subtracting the composite profile of g_2 from g_1.* ***C)*** *Results related to the human developmental heart data. The "HE-images" panel shows the charted HE-images, landmarks are represented with colored markers. The 'Composite" panel gives the composite representation (across samples, n = 7) of the transferred data for each gene. The "Reference" panel shows the reference to which data was transferred together with the four different regions, landmarks are indicated with colored markers. The "Enrichment" panel depicts the predicted expression values of each transferred sample (black dots) within respective region. Mean values are represented with a red marker.*

compatibility, we also transfer gene expression in the mouse hippocampal area from data collected using both the Slide-seqV2 and Visium platforms. The Visium sample was charted with the help of the associated HE-image, while we relied on total UMI-counts for the Slide-seqV2 data, exemplifying how both morphology and molecular information may be used in the charting process (see Supplementary Figure 11). As shown in Supplementary Figures 13 (Observed) and 14 (Transferred), data from the two platforms were successfully integrated while preserving the intricate structure of the expression patterns.

Finally, we produced a new set of 10x Genomics Visium data consisting of seven sections (A-F) à two individuals from human developmental heart (dh) tissue (collected at the tenth postconceptional week). We then transferred the expression profiles of four genes (*COL2A1, ELN, MYH6* and *MYH7*) from all seven sections in this data set to a single reference. Despite vast inter-individual differences in the structure, the transferred data correlated well between patients; the mean between-individual correlation was 0.88 , while the mean within-individual correlation was 0.96 for individual 1 and 0.91 for individual 2, see Supplementary Figure 15. We also generated gene-specific composite profiles, see Figure 2C. Separate representations of each combination of gene and section pairs are found in Supplementary Figure 5 (Observed) and 6 (Transferred). We also segmented the reference into four distinct spatial regions, which allowed us to assess region-specific enrichment of genes. Importantly, the enrichment analysis does not require any additional annotation of the original tissue samples, and the regions can be redefined without any need to repeat the transfer process. As expected, *MYH6* expression was highest in the atrial regions (Region 2 and 3), *MYH7* expression was elevated in the ventricular body (Region 0), and *ELN* in the outflow tract (Region 1).[9] The atria also were enriched for *COL2A1* but we, interestingly, observed a preserved and statistically significant left-right asymmetry in its expression ($p_{value} < 0.05$, two-sided permutation test).

Gene expression may be the primary information that spatial transcriptomics techniques produce, but there's now a panoply of methods to infer second order insights from said data. Thus, to demonstrate the flexibility of our method, we transferred inferred (by the tool *stereoscope*) cell type proportion values between two Visium sections of human breast cancer, see Supplementary Figure 7.[10]

In this study we have presented a new and general method to transfer spatial transcriptomics data from multiple samples to a shared reference, something that previously only has been conceptually described. We look at a varied set of tissue types, the olfactory bulb and hippocampus are more symmetrically organized tissues, the heart is a complex asymmetric tissue, and cancer in general possess a more random spatial structure than healthy organs. Using these tissues we show how eggplant generalize well and is applicable to a broad set of targets. The method is versatile and effortless to use. Furthermore, the implementation leverages the GPyTorch framework, which supports GPU acceleration together with efficient algorithms to reduce the inference's complexity.[11] Our tool is useful for visualization purposes, but also prepares the data for more extensive analysis, such as spatiotemporal modeling, spatial arithmetics, and regional enrichment. We are currently relying on manual identification of landmarks, but see a great opportunity for future research to explore different venues for unsupervised landmark detection. Taken together, we consider this an important first step towards harmonizing and integrating spatial transcriptomics data in a common coordinate framework, with particular relevance for the collaborative Human Cell Atlas initiative.

3 Methods

3.1 Code Availability

An implementation of our method is provided as a Python package named *eggplant*, short for ***e**ffortless **g**eneric **GP** **lan**dmark **t**ransfer*. The package can be accessed at the GitHub repository `https://github.com/almaan/eggplant`. The repository also contains a set of Jupyter notebooks outlining all the presented analyses as well as generation of the synthetic data associated with this study. The repository also contains scripts to download and curate the public data that we've used. We have also deposited a clone of the repository together with the charted data at Zenodo, accessible via `https://doi.org/10.5281/zenodo.5659093` .

3.2 Data Availability

Except for the synthetic and developmental heart data, we used publicly available data sets in this study. We thus refer to the original data sources for access, which we list below:

- Synthetic data: `https://github.com/almaan/eggplant`
- MOB data: `https://www.spatialresearch.org/resources-published-datasets/doi-10-1126science-aaf2403/`
- Mouse hippocampus (Visium): `https://support.10xgenomics.com/spatial-gene-expression/datasets/1.1.0/V1_Adult_Mouse_Brain`
- Moue hippocampus (Slide-seqV2, Puck_200115_08): `https://singlecell.broadinstitute.org/single_cell/study/SCP815/highly-sensitive-spatial-transcriptomics-at-near-cellular-resolution-with-slide-seqv2`
- bcA: `https://support.10xgenomics.com/spatial-gene-expression/datasets/1.1.0/V1_Breast_Cancer_Block_A_Section_1`
- bcB: `https://support.10xgenomics.com/spatial-gene-expression/datasets/1.1.0/V1_Breast_Cancer_Block_A_Section_2`
- Single cell HER2 data : `https://zenodo.org/record/4739739#.YPF2D5KxVhE`

For the human developmental heart, raw sequencing data can be accessed at the Gene Expression Omnibus (GEO) with access code GSEXXXXXX (*). All processed data together with the presented results are available at the GitHub and Zenodo repositories associated with this manuscript.(*)

**Note*: all new data will be publicly available upon publication of the manuscript.

3.3 Data Acquisition

3.3.1 Human Developmental Heart

After collection, the human developmental heart tissue was fresh-frozen and embedded in Tissue-Tek (OCT). The tissues samples were cryosectioned at 10 μm thickness and placed on 10X Visium spatial gene expression slides, to then be stored at $-80^{o}C$ prior the library preparation. Libraries were generated from the samples using Visium Spatial Gene Expression kit from 10x Genomics. Every barcoded Visium array contains four capture areas á 4992 spots, where each spot contains probes consisting of: a spatial barcode, an UMI sequence, and a poly-dT-VN sequence enabling mRNA capture. Sections were fixed for 30 min in Methanol, stained with Hematoxylin and Eosin and imaged using Metafer Slide Scanning system (Metasystem, Altlussheim, Germany). The 10x Genomics Visium Tissue Optimization Kit was used to determine the optimal permeablization time, rendering an estimate of 20 mins. The generated libraries were sequenced using the Illumina Platform. The lengths for read 1 and read 2 were 28 bp respectively 120 bp. The sequencing data was processed with *spaceranger* v.1.2.0.

3.4 Data Processing

In the Slide-seqV2 data, we removed all beads with less than 100 UMI's and then subsampled the remainder to 20% of its size. For Visium and first generation Spatial Transcriptomics (ST1K) data, we used all spots

identified to be under the tissue (for public data sets we used the original annotations).

When analyzing gene expression data, we applied a simple normalization strategy compiled of functions from the *scanpy* (v. 1.8.1) package, the recipe is given below:

1. *scanpy.pp.filter_genes(...,min_cells = 0.1)*
2. *scanpy.pp.normalize_total(...,1e4, exclude_highly_expressed = True)*
3. *scanpy.pp.log1p(...)*
4. *scanpy.pp.scale(...)*

When cell type proportions acted as the feature of interest we only used standard scaling (subtraction by mean and division by standard deviation).

Working with the older ST1K data, we also added a *spatial smoothing* step to the above recipe (as a last step), to counteract "holes" caused by tears or ruptures of the tissue as well as steep gradients and variation in the capture efficacy across the tissue. The spatial smoothing is a form of weighted average of the feature values observed in a given location's neighborhood. The neighborhood of spot s is denoted as $\mathcal{N}(s)$ and contains said spot together with its four nearest neighbors. If y_s is the prior feature value associated with spot s, then the smoothed equivalent y_s^{smooth} is defined as:

$$\begin{aligned} y_s^{\text{smooth}} &= \textstyle\sum_{s' \in \mathcal{N}(s)} w_{s'} y_{s'} \\ w_{s'} &= \frac{\bar{w}_{s'}}{\sum_{k \in \mathcal{N}(s)} \bar{w}_k} \\ \bar{w}_{s'} &= \exp(-||\boldsymbol{u}_{s'} - \boldsymbol{u}_s||/\sigma) \end{aligned} \tag{1}$$

Where $\boldsymbol{u}_s$ are the coordinates of spot s and $||.||$ represents the L2-norm (euclidean distance). In our analysis we used $\sigma = 50$.

3.5 Model

The method we propose transfers a feature of interest from one coordinate system to a given reference system, below we describe the process in more detail.

Let Ω be the domain from which the observed data is collected, while Ω' represents the reference domain onto which we seek to transfer information. Similarly, $\mathcal{L} \subset \Omega$ is the set of landmarks in the observed data, while $\mathcal{L}' \subset \Omega'$ represents the landmark positions in the reference. Here, $|\mathcal{L}| = |\mathcal{L}'| = L$, where L is the number of landmarks and $|.|$ is the cardinality operator. Importantly, $\mathcal{L}$ and $\mathcal{L}'$ are ordered in the same way. We also define $U \subset \Omega$ and $U' \subset \Omega'$ as the sets of coordinate tuples containing the location of each observation ($\boldsymbol{u}_i$) and reference points ($\boldsymbol{u}'_j$). Every observation i has a target value y_i associated with it, and our primary objective is to find the corresponding values y'_j for the reference points.

First, we will transform the coordinate tuples in U and $\mathcal{L}$, to put distances between objects in the two sets at the same lengthscale as between their reference counterparts (U' and $\mathcal{L}'$). The transformation h can either be a simple linear scaling: $h(\boldsymbol{u}_i) = h_{\text{const}}(\boldsymbol{u}_i) = a \cdot \boldsymbol{u}_i$, or a more complex transformation relying on thin plate splines (TPS). In the case of the former, a will be given as the average of ratios between landmark-pair distances, that is:

$$a = \frac{2}{L^2 - L} \sum_{i}^{L} \sum_{j \neq i}^{L} \frac{||\boldsymbol{l}'_i - \boldsymbol{l}'_j||_2}{||\boldsymbol{l}_i - \boldsymbol{l}_j||_2}, \quad \boldsymbol{l}_i \in \mathcal{L}, \boldsymbol{l}'_i \in \mathcal{L}' \tag{2}$$

In the second case, h will be a composite function given as $h(\boldsymbol{u}_i) = h_{\text{TPS}}(h_{\text{const}}(\boldsymbol{u}_i))$, where h_{TPS} restricted to a family of TPSs parametrized by minimizing the cost C:

$$C = \sum_{i=1}^{L} ||\boldsymbol{l}'_i - h(\boldsymbol{l}_i)||_2 \tag{3}$$

The transformed versions of U and $\mathcal{L}$, obtained by applying h to every element in respective set, are referred to as U^* and $\mathcal{L}^*$. In our implementation, we use the Python package *Morphops* (v. 0.1.12) for the TPS warping .

Next, for all members of U^* and U', we compute the distances to $\mathcal{L}^*$ respectively $\mathcal{L}'$, forming the two new sets X and X' defined as:

$$\begin{aligned} \boldsymbol{x}_{ip} &= ||\boldsymbol{u}_i^* - \boldsymbol{l}_p^*||_2 \\ \boldsymbol{x}'_{jp} &= ||\boldsymbol{u}'_j - \boldsymbol{l}'_p||_2 \end{aligned} \tag{4}$$

We then seek a function φ that will allow us find the feature values associated with each location in our reference:

$$\varphi(X') = \boldsymbol{y'} \tag{5}$$

To learn φ, we use Gaussian Process Regression (see Section 3.6) where the observed data is used to learn said function.

3.6 Gaussian Process Regression

Gaussian Process (GP) Regression is fundamental to our method, and we will therefore briefly describe it in the context of our work. However, for a more elaborate account of GP regression we refer to any of the (many) already existing works on the subject, for example the canonical text by Rasmussen and Williams.[12]

A GP is defined as a collection of random variables, of which any finite subset have a joint Gaussian distribution. Hence, a GP may be interpreted as a distribution over functions that fit a certain set of points. We denote a function f that is distributed according to a GP as:

$$f(.) \sim GP(\mu(.), k(.,.)) \tag{6}$$

Here, $\mu(.)$ and $k(.,.)$ represent the mean respectively covariance function (also referred to as the kernel).

In our model, the function f relates landmark distances to the feature of interest's values. We represent the complete set of observed data as the tuple $(X, \boldsymbol{y})$, where $X \in \mathbb{R}^{M \times L}$ is the matrix representing the distances to each of the L landmarks for all of the M observations, and $\boldsymbol{y} \in \mathbb{R}^M$ is the value of the feature of interest associated with each observation. Distances and feature values are related via f, that is $f(X) = \boldsymbol{y}$. The distances from the locations to the landmarks (in the reference) are represented by X' while $\boldsymbol{y'}$ indicates the reference target values (which we seek to approximate).

Due to the properties of GPs, the joint distribution $p(\boldsymbol{y}, \boldsymbol{y'}|X, X'; \sigma)$ thus becomes:

$$\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{y'} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu(X) \\ \mu(X') \end{bmatrix}, \begin{bmatrix} k(X,X) + \sigma^2 I & k(X,X') \\ k(X',X) & k(X',X') \end{bmatrix} \right) \tag{7}$$

Where we account for noise in the training data according to the model : $\boldsymbol{y} = f(X) + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2)$. Using standard Gaussian identities and the assumption $\mu(.) = c \cdot \mathbf{1} = \boldsymbol{c}$, where c is some real number, the conditional distribution $p(\boldsymbol{y'}|X', X, y; \Theta)$ becomes:

$$\begin{aligned} &\boldsymbol{y'}|\boldsymbol{y}, X, X' \sim \mathcal{N}(\hat{\boldsymbol{\mu}}, \hat{k}) \\ &\hat{\mu} = \boldsymbol{c} + K_*(\boldsymbol{y} - \boldsymbol{c}) \\ &\hat{k} = k(X', X') - K_* k(X, X') \\ &K_* = k(X', X)(k(X, X) + \sigma^2 I)^{-1} \end{aligned} \tag{8}$$

With $(X, \boldsymbol{y})$ being given, we consider the conditional mean a function of X' and will use this as φ described in Section 3.5, that is:

$$\varphi(X') = \boldsymbol{c} + k(X', X)(k(X, X) + \sigma^2 I)^{-1}(\boldsymbol{y} - \boldsymbol{c}) \tag{9}$$

We support several different kernel functions but use the RQKernel as default, which is defined as:

$$[k_{RQ}(X, X')]_{ij} = \left(1 + \frac{1}{2\alpha}(\boldsymbol{x}_i - \boldsymbol{x}'_j)\Gamma^{-2}(\boldsymbol{x}_i - \boldsymbol{x}'_j)^T\right)^{-\alpha} \tag{10}$$

Where $\boldsymbol{x}_i$ and $\boldsymbol{x}'_j$ refers to the i:th respectively j:th row of X and X', while $\alpha \in \mathbb{R}$ and $\Gamma \in \mathbb{R}^L$ are kernel parameters. To find optimal values of the parameters $\Theta = [c, \sigma, \alpha, \Gamma]$, we optimize the marginal likelihood $p(\boldsymbol{y}|X; \Theta)$ using stochastic optimization. Once these parameters have been estimated, φ can be used to estimate $\boldsymbol{y'}$. Implementation-wise we leverage the GPyTorch (v. 1.5.0) framework for both inference and prediction.

3.7 Synthetic Data

Here we outline the process by which each synthetic data set was created, the time series data refers to the set analyzed in Figure 2 while the distortion data refers to the set presented in Supplementary Figure 2.

3.7.1 Time series data

Eight two-colored images (see Supplementary Figure 9) were used to generate the spatial domain for each time point. To convert images to array data, *eggplant*'s *reference_to_grid* function from the *preprocess* module was used, this also assigned each spot in the array to one of two groups (Compartment 1/C1 and Compartment 2/C2). The number of transcripts in each compartment was dictated by the dynamical system; from which expression values at select time points were extracted and rounded to the nearest integer value. The transcripts were then randomly distributed between array nodes in the associated compartment. Below we describe the dynamical model in more detail.

The dynamical model describes a two-compartment system governed by the following set of equations:

$$\begin{aligned} \frac{dc_1}{dt} &= (r_{11} - r_{12})c_1 + r_{21}c_2 \\ \frac{dc_2}{dt} &= (r_{22} - r_{21})c_2 + r_{12}c_1 \end{aligned} \tag{11}$$

Where c_1 is the amount of material in compartment 1 and c_2 the same but for compartment 2. From a given set of initial values, $(c_1(0), c_2(0))$, the system was then propagated in time for a pre-determined number of steps (T). Here the following parameter values – arbitrarily chosen – were used: $(r_{11}, r_{12}, r_{21}, r_{22}) = (0.2, 0.1, 0.8, -0.3)$ together with the initial values $(c_1(0), c_2(0)) = (5000, 100)$. The eight time points from which we extracted expression values were equally spaced in in the interval $[0, 500]$.

In figures, tables and text we refer to this synthetic data set as "Synthetic 1".

3.7.2 Distorted data

First, a $p \times p$ grid where each node represented a spatial capture location (e.g., spot) was generated, to figure as the domain in which signals will be collected. Next, to produce a spatial expression pattern, an i iterations long random walk was performed (the initial position also being randomly sampled from the domain). The number of times a node was visited in the walk was let to represent its observed expression level; this data represent the "ground truth". From the ground truth, a "distorted" representation of the same sample was produced by first applying a distortion field ($F(x, y)$) to the node positions while keeping their values constant. Then, we placed a new $p \times p$ grid identical to the first over the distorted data, and interpolated its node values by a nearest neighbor approach. For a depiction of the process see Supplementary Figure 2. For our data we let $p = 32$, $i = 1 \times 10^4$, and $F(x, y) = \frac{2}{\sqrt{x^2+y^2}} \cdot (-y + x, x + y)$.

In figures, tables and text we refer to this synthetic data set as "Synthetic 2".

3.8 Choosing the number of landmarks

While including more landmarks generally will render a better result, this gain in performance tends to be marginal after a certain number of landmarks have been included in the analysis. Hence, we aim to provide means to estimate a *lower bound* of the number of landmarks that should be used when transferring information to a reference. Below, we describe the steps to derive this lower bound.

First, we select one representative sample from our data set and position L landmarks in the (spatial) domain which

the sample inhabits. Then, landmarks are randomly placed in the domain using Poisson Disk Sampling, where the first landmark always is located at the domain center.[13] We denote the set of all landmarks as $\mathcal{L}$, this set is considered as ordered. Next we specify a sequence ($N_L = \{l_1, ..l_P\}, l_1 \geq 1, l_p \leq L$) of the numbers of landmarks that should be evaluated. Then, for each entry l_i we randomly choose l_i of the L landmarks, and learn the transfer function using the representative sample. In this analysis, the normalized total-UMI count figures as the feature of interest. For each number of landmarks l_i, we compute the mean negative marginal log likelihood (nMLL) for the last T iterations when fitting the model, and compare the mean values between all numbers in N_L. This process is repeated for n_{rep} times, which allows us to compute an average for each element l_i. By inspecting the graph obtained by plotting the average nMLL values as a function of the number of landmarks and applying a *Savitzky–Golay* filter for smoothing, we let the lower bound be defined as the number of landmarks where the average nMLL starts to plateau.

Table 1 shows the estimated lower bounds together with the actual number of used landmarks in the analysis, the graphs from which the lower bounds were determined are shown in Supplementary Figure 16. In all of our analyses we aimed to use as many landmarks as we could confidently identify, with the requirement that this number should be higher than the – to each sample – associated lower bound.

The following parameter values were used: $N_L = \{1, 3, 5, 7, 9, 11, 13, 15, 17, 20\}$, $T = 200$, $n_{rep} = 5$, for the Savitzky–Golay filter we used the function *savgol_filter* from the *scipy.signal* module (v. 1.7.1) with parameters `window_length=5` and `polyorder=4`.

3.9 ODE parameter estimation

To estimate the parameters of the ODE system representing the dynamical model, after the synthetic data had been transferred to the reference, we used the BFGS algorithm with a cost function dependent on the system model (Equation 11). First we aggregated the data in each compartment to get an expression tuple for every time point, that is:

$$\boldsymbol{c}(t) = (c_1(t), c_2(t)) = (\sum_{s \in C_1} y'_s(t), \sum_{s \in C_2} y'_s(t)) \quad (12)$$

Where C_i is the set of spots in compartment i, y'_s is the transferred expression value at array point s, and t represents time point t. Next, let $p(., \boldsymbol{r}; T)$ represent a function that propagates the first argument according to the dynamics given in Equation 11 T steps forward in time with parameter values $\boldsymbol{r}$. From this, the cost (C) for a given set of parameters $\boldsymbol{r}$ takes the form:

$$C = \frac{1}{2|\mathcal{T}|} \sum_{t \in \mathcal{T}} \sum_{i=1}^{2} [c_i(t) - p_i(\boldsymbol{c}(0), \boldsymbol{r}, t)]^2 \quad (13)$$

Where $\mathcal{T}$ is the set of observed time points. We used the *minimize* function from *scipy*'s *optimization* module for the optimization, and *odeint* function from the *integrate* module to solve the ODE system; with *scipy* (v. 1.7.1)

3.10 Spatial Arithmetics

Conducting any form of spatial arithmetics is straightforward once observed data has been transferred to the same reference. If we let $\lambda(.,.)$ represent an arbitrary arithmetic operation, and $y_s^{(i)}$ denotes transferred data from sample i at location s in the reference, then:

$$\xi_s^{(i,j)} = \lambda(y_s^i, y_s^j) \quad (14)$$

gives the expression for the spatial arithmetic calculation, where $\xi_s^{(i,j)}$ is the output associated with location s.

3.11 Spatial Differential Expression Analysis

From the Gaussian Process Regression, we obtain both mean and variance estimates of the feature values at each location, together these can be used to perform *spatial differential expression analysis* (sDEA) between groups of samples (e.g., disease vs. control). First, we compute the local group mean (μ) and variance (σ^2) values for a FOI, which for any group G and location s are defined as:

$$\mu_s^{(G)} = \sum_{g \in G} w^{(g)} \mu_s^{(g)} \tag{15}$$

And

$$\begin{aligned}(\sigma_s^2)^{(G)} = &\sum_{g \in G} w^{(g)} (\sigma_s^2)^{(g)} + \\ &\sum_{g \in G} w^{(g)} (\mu_s^{(g)})^2 - \\ &(\sum_{g \in G} w^{(g)} \mu_s^{(g)})^2\end{aligned} \tag{16}$$

Where $w^{(g)}$ denotes the weight that should be given to sample g when computing the mean, and $\sum_{g \in G} w^{(g)} = 1$ if nothing else is stated we assign equal weights to all samples within the same group. Next, for each group G and location s we construct an interval ($I_s^{(G)}$) according to:

$$I_s^{(G)} = [\mu_s^{(G)} - z * \sigma^{(G)}, \mu_s^{(G)} + z * \sigma)^{(G)}] \tag{17}$$

Where z relates to the number of samples that would fall into the interval if we were to sample new values from the mixed distribution, if nothing else is stated we use $z = 2$. Finally, we consider the FOI to be to be spatially differentially expressed at location s between the two groups G_i and G_j if the two intervals $I_s^{(G_i)}$ and $I_s^{(G_j)}$ do not overlap. Evidently, a larger value of z will require that the two groups are more distinct in their expression of the FOI to be considered spatially differentially expressed at a given location.

3.12 Analysis

3.12.1 Transfer to reference with *eggplant*

In all analysis steps we used 1000 epochs, an *RQKernel*, and the Adam optimizer with a learning rate of 0.01. The references were all represented by approximately 1000 array points, except for the human developmental heart data where 10000 array points were used. The number of landmarks used in each analysis are listed in 1. The landmarks did not correspond to any "established" anatomical features but were rather selected based on their ease of identification from the morphology or gene expression pattern across the examined samples.

All references used in our analyses are found in Supplementary Figure 8 together with their respective landmark annotation. The charted observed data is displayed in Supplementary Figure 9-12. All this information is also available in the – to this manuscript – associated GitHub repository.

Data set	Lower Bound	Used Landmarks
Synthetic 1	5	9
MOB	9	14
Mouse Hippocampus	5	6
Human Breast Cancer	5	10
Human Developmental Heart	5	16

Table 1: *Landmark lower bounds and number of used landmarks. The column "Lower bound" gives the estimated lower bound for the number of landmarks to be used in each data set. The column "Used Landmarks" lists the number of landmarks actually used in the analysis. The representative sample (S) from each data set (D) are given as (D,S): (Synthetic 1, t_7), (MOB, Rep1), (Mouse Hippocampus, Visium), (Human Breast Cancer, bcA), (Human Developmental Heart, dhA).*

3.12.2 Benchmarking and Landmark Influence

We compared the transfer made by *eggplant* with three alternative strategies: "no correction", "constant mean", and thin plate spline

interpolation (TPS). The task designed to measure performance consisted of trying to transfer distorted data back to its original (ground truth) distribution in a data set generated according to the procedure described in Methods Section 3.7.2. The Root Mean Squared Value (RMSE) value between the ground truth and the corrected values was used as a metric to assess performance. In the "no correction" strategy, the grid values in the distorted data is immediately compared to the ground truth values. This strategy emulates a scenario where tissue sections would be aligned, but non-linear distortions not accounted for. In the "constant mean" approach, we assign all grid points the same value, being the mean value. Notably, the expected RMSE value for this approach is 1 since we applied standard scaling to the data. Finally, with the TPS method, the same landmarks as provided to *eggplant* were used to correct for the distortion; then every grid point in the reference domain was assigned the value of its nearest neighbor among the shifted data points. We compared the strategies with different number of landmarks ($L \in \{3, 7, 11, 15, 19, 23, 27, 31, 35, 39\}$) and repeated each comparison 3 times to asses variance of the outcome. In each iteration, the landmarks were selected from a set of 40 random positions – sampled by the same Poisson Disc Sampling strategy as referenced above – in the spatial domain and then distorted by the same field F as the grid points during generation of the distorted set. For the TPS strategy we use the *Morphops* (v. 0.1.12) package , for the 2d interpolation we used *scipy.interpolate*'s *griddata* function (v. 1.7.1).

3.12.3 Statistical Tests

In our study we perform a permutation test to asses whether there's an asymmetry between the two different atria (Region 2 and 3) w.r.t. *COL2A1* expression in the human developmental heart data set. We favored a permutation test since our observations violate the i.i.d. assumption that most statistical tests rely on. We outline how this test is constructed below.

For two arbitrary regions A and B, we let R_A and R_B denote the sets of feature values associated with the locations contained within respective region. Without loss of generality, we here assume that our objective is to determine whether the expression of a feature of interest differs between region A and region B. We define the *mean region difference* ($\Delta_{A,B}$) as the mean of the difference in feature value across all combinations of observations from each set. That is:

$$\Delta_{A,B} = \frac{1}{|R_A| * |R_B|} \sum_{x \in R_A} \sum_{y \in R_B} x - y \qquad (18)$$

Our objective is then equivalent to testing whether the observed mean region difference is more extreme than what is expected by chance. To perform this test we shuffle the observations' region labels and compute the $\Delta_{A,B}$ value for each permutation. We then compute the p-value as:

$$\begin{aligned} p_1 &= \tfrac{1}{n_{\text{perm}}} \textstyle\sum_i^{n_{\text{perm}}} \mathbb{I}[\Delta_{A,B}^{\text{perm},i} \leq \Delta_{A,B}^{\text{obs}}] \\ p_2 &= \tfrac{1}{n_{\text{perm}}} \textstyle\sum_i^{n_{\text{perm}}} \mathbb{I}[\Delta_{A,B}^{\text{perm},i} \geq \Delta_{A,B}^{\text{obs}}] \\ p_{\text{val}} &= 2 \times \min(p_1, p_2) \end{aligned} \qquad (19)$$

Where $\mathbb{I}$ is the indicator function. If $p_{\text{val}} \leq \alpha$, the difference in expression between the two regions is considered statistically significant. Here α is the significance level, and the test is two-sided in its character. In our analysis of the *COL2A1* right-left asymmetry, we let $\alpha = 0.05$ and ran 1000 permutations.[14] The test was applied to the *composite* representation of the *COL2A1* expression.

3.12.4 Single cell mapping with *stereoscope*

For the *stereoscope* (v. 0.3.1) analysis we used the *major* cell type tier found in the single cell data, only including cells from HER2-positive patients. Cell types with less than 25 members were excluded, for cell types with more than 500 members, a subset consisting of 500 cells were randomly sampled from these. We also used a curated list of genes in the analysis consisting of 5540 members, representing

a union of the 5000 highest expressed genes and cell type specific marker genes, see Supplementary Data 13 in [15]. We used 50000 epochs and a batch size of 2048 for the single cell parameter estimation as well as the proportion inference.

3.13 Contributions

A.A. conceived the method, implemented it in code, analyzed the data, generated the results, and wrote the paper under supervision of J.L. P.C. was involved in the discussion of using a landmark-based approach, and also beta-tested the code. Together, Ž.A., X.L., and E.S. produced the human developmental heart data, Ž.A. also helped in the writing of the experimental methods part. All authors read and gave feedback on the paper.

3.14 Acknowledgments

The authors thank Marco Vicari for helpful discussions regarding the developmental heart's anatomy and structure. We also thank both Franziska Hildebrandt and Ludvig Larsson whom read the paper and provided constructive comments. Finally, a special thanks is directed to Ludvig Bergenstråhle for giving invaluable feedback. This work was made possible by generous support from the Knut and Alice Wallenberg foundation, the Erling-Persson family foundation, the Swedish Cancer Society, the Swedish Foundation for Strategic Research, Karolinska Intsitutet Research Funds, and the Swedish Research Council. Human developmental heart tissue was acquired through the Karolinska Institutet Developmental Tissue Bank.

3.15 Ethics declaration

The use of human developmental heart in the study was approved by the Regional Ethical Review Board in Stockholm and the National Board of Health and Welfare. The procuration of the tissue and processing of the data were in concordance with the ethical stipulations of the Helsinki Convention (Dnr: 2:9/2015). The human developmental heart tissue used in this study was retrieved at the Department of Gynecology, Danderyd Hospital and Karolinska Huddinge Hospital, all patients provided a written statement of informed consent.

3.15.1 Competing Interests

A.A., Ž.A. and J.L. are scientific consultants for 10x Genomics Inc., providing spatially barcoded slides. The remaining authors declare no competing interests.

References

[1] Anjali Rao et al. "Exploring tissue architecture using spatial transcriptomics". In: *Nature* 596.7871 (Aug. 2021), pp. 211–220. DOI: 10.1038/s41586-021-03634-9. URL: https://doi.org/10.1038/s41586-021-03634-9.

[2] Vivien Marx. "Method of the Year: spatially resolved transcriptomics". In: *Nature Methods* 18.1 (Jan. 2021), pp. 9–14. DOI: 10.1038/s41592-020-01033-y. URL: https://doi.org/10.1038/s41592-020-01033-y.

[3] Aviv Regev et al. "The Human Cell Atlas". In: *eLife* 6 (Dec. 2017). DOI: 10.7554/elife.27041. URL: https://doi.org/10.7554/elife.27041.

[4] Jennifer E. Rood et al. "Toward a Common Coordinate Framework for the Human Body". In: *Cell* 179.7 (Dec. 2019), pp. 1455–1467. DOI: 10.1016/j.cell.2019.11.019. URL: https://doi.org/10.1016/j.cell.2019.11.019.

[5] Valentine Svensson, Sarah A Teichmann, and Oliver Stegle. "SpatialDE: identification of spatially variable genes". In: *Nature Methods* 15.5 (Mar. 2018), pp. 343–346. DOI: 10.1038/nmeth.4636. URL: https://doi.org/10.1038/nmeth.4636.

[6] Damien Arnol et al. "Modeling Cell-Cell Interactions from Spatial Molecular Data with Spatial Variance Component Analysis". In: *Cell Reports* 29.1 (Oct. 2019), 202–211.e6. DOI: 10.1016/j.celrep.2019.08.077. URL: https://doi.org/10.1016/j.celrep.2019.08.077.

[7] F. Alexander Wolf, Philipp Angerer, and Fabian J. Theis. "SCANPY: large-scale single-cell gene expression data analysis". In: *Genome Biology* 19.1 (Feb. 2018). DOI: 10.1186/s13059-017-1382-0. URL: https://doi.org/10.1186/s13059-017-1382-0.

[8] Patrik L. Ståhl et al. "Visualization and analysis of gene expression in tissue sections by spatial transcriptomics". In: *Science* 353.6294 (June 2016), pp. 78–82. DOI: 10.1126/science.aaf2403. URL: https://doi.org/10.1126/science.aaf2403.

[9] Michaela Asp et al. "A Spatiotemporal Organ-Wide Gene Expression and Cell Atlas of the Developing Human Heart". In: *Cell* 179.7 (Dec. 2019), 1647–1660.e19. DOI: 10.1016/j.cell.2019.11.025. URL: https://doi.org/10.1016/j.cell.2019.11.025.

[10] Alma Andersson et al. "Single-cell and spatial transcriptomics enables probabilistic inference of cell type topography". In: *Communications Biology* 3.1 (Oct. 2020). DOI: 10.1038/s42003-020-01247-y. URL: https://doi.org/10.1038/s42003-020-01247-y.

[11] Jacob R Gardner et al. "GPyTorch: Blackbox Matrix-Matrix Gaussian Process Inference with GPU Acceleration". In: *Advances in Neural Information Processing Systems*. 2018.

[12] Carl Rasmussen. *Gaussian processes for machine learning*. Cambridge, Mass: MIT Press, 2006. ISBN: 978-0262182539.

[13] Robert Bridson. "Fast Poisson disk sampling in arbitrary dimensions". In: ACM Press, 2007. DOI: 10.1145/1278780.1278807. URL: https://doi.org/10.1145/1278780.1278807.

[14] M. Marozzi. "Some remarks about the number of permutations one should consider to perform a permutation test". In: *Statistica; Vol 64* (2007), No 1 (2004), 193–201. DOI: 10.6092/ISSN.1973-2201/32. URL: http://rivista-statistica.unibo.it/article/view/32.

[15] Alma Andersson et al. "Spatial deconvolution of HER2-positive breast cancer delineates tumor-associated cell type interactions". In: *Nature Communications* 12.1 (Oct. 2021). DOI: 10.1038/s41467-021-26271-2. URL: https://doi.org/10.1038/s41467-021-26271-2.