Cell segmentation for high-resolution spatial transcriptomics

Hao Chen [1], Dongshunyi Li [1], and Ziv Bar-Joseph [*1,2]

Abstract

Spatial transcriptomics promises to greatly improve our ability to understand tissue organization and cell-cell interactions. While most current platforms for spatial transcriptomics only provide multi-cellular resolution (10-15 cells per spot), recent technologies provide a much denser spot placement leading to sub-cellular resolution. A key challenge for these newer methods is cell segmentation and the assignment of spots to cells. Traditional, image based, segmentation methods face several drawbacks and do not take full advantage of the information profiled by spatial transcriptomics. Here, we present SCS, which integrates imaging data with sequencing data to improve cell segmentation accuracy. SCS combines information from neighboring spots and employs a transformer model to adaptively learn the relevance of different spots and the relative position of each spot to the center of its cell. We tested SCS on two new sub-cellular spatial transcriptomics technologies and compared its performance to traditional image based segmentation methods. As we show, SCS achieves better accuracy, identifies more cells and leads to more realistic cell size estimation. Analysis of RNAs enriched in different sub-cellular regions based on SCS spot assignments provides information on RNA localization and further supports the segmentation results.

Availability: Code and example data are available at: https://github.com/chenhcs/SCS

[1] Computational Biology Department, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA

[2] Machine Learning Department, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA

[*] To whom correspondence should be addressed. Email: zivbj@cs.cmu.edu

1 Introduction

Spatial transcriptomics offers the unique ability to study both the internal and the external networks and systems that shape cell and tissue fate[1–3]. Unlike traditional multiplexing approaches, spatial transcriptomics profiles the expression of all genes, across tens of thousands of cells [4–8] . Unlike single cell sequencing approaches (scRNA-Seq) it provides spatial information about the location of the cells being profiled enabling the analysis of cell-cell signaling and cell type organization [9–12]. Several recent studies have demonstrated the usefulness of these profiling methods including for studying development [13], disease progression [14], immune infiltration [15] and several other biological systems and processes.

Spatial transcriptomics uses a set of barcoded spots that are placed at regular intervals on the sample to profile the expression of genes [6, 16]. To date, most platforms (including the commercial popular ones such as Visium [16]) used spots that were placed up to 100 μm apart. Thus, each spot profiled the expression of 10-20 cells depending on the tissue, making it hard to achieve some of the major goals on this technology, including the analysis of cell type distribution and the modeling of cell-cell interactions. Very recently, new spatial technologies enable a much more dense spot placement. For example, both Stereo-seq [6] and Seq-scope [5] achieve a spot-to-spot distance of 0.5 μm on average, resulting in more than 1000 spots per cell (Fig. S1 S2). In addition to enabling cell level analysis, such spot placement provides information on sub-cellular RNA distribution as well [5].

While promising, the new spatial technologies also raise new computational problems. To enable single cell based studies, a key step is cell segmentation and the assignment of an expression profile to each cell. Recently, new segmentation methods were developed for spatial proteomics [17] or FISH based spatial transcriptomics data [18]. These utilize the spatial distribution of RNA or protein molecules to improve cell segmentation. However, these methods cannot be extended to spatial transcriptomics due to the large number of genes profiled and sparseness of the expression captured by each spot. Most standard cell segmentation methods developed to date rely on nucleus or membrane straining to identify cell boundaries (*e.g.*, nucleic acid, Hematoxylin and eosin). Popular methods that perform such segmentation include Watershed [19], Cellpose [20], DeepCell [21], and StarDist [22]. While successful, these methods share a few problems when it comes to segmenting sub-cellular spatial transcriptomics data. First, in nucleus staining experiments, identification of cell boundaries is a challenge [23]. In addition, even when multiple channels are stained, it is frequently hard to clearly visualize boundaries for all cell types in a tissue [24]. Finally, the deep learning based methods in this category, such as DeepCell, Cellpose, and StarDist, require manual annotations for model training, which are not easy to obtain in sufficient quantities when profiling new tissues or sections.

To address these challenges and obtain accurate segmentation in high-resolution spatial transcriptomics, we developed a new method, SCS (Sub-cellular spatial transcriptomics Cell Segmentation), which combines sequencing and staining data to identify cell boundaries. To utilize the high-dimensional but sparse gene expression data of barcoded spots, we developed a transformer model with attention mechanism. The attention mechanism is used to aggregate information by adaptively learning the relevance of different neighboring spots, which has been shown in different tasks to be more powerful than weighting information with fixed learned kernels in convolutional neural networks or fully connected neural networks [25, 26]. In addition, the model maps spots to low-dimensional latent representations which are used to infer their relative position w.r.t. the centers of their cells. Rather than relying on manual annotations, we train the model by using the stained nucleus spots as ground truth inner cell spots. Model training is then performed on spots within the nucleus regions and applied to spots outside that region to determine if they are within, or outside, the cell.

We tested SCS on public subcellular-resolution *in situ* datasets generated using two different platforms: a mouse brain dataset from Stereo-seq [6] and a mouse liver dataset from Seq-scope [5]. We evaluated the method by comparing it with Watershed, which is the method used in the original publications of these platforms, as well as other state-of-the-art image based methods, including Cellpose, Deepcell, and StarDist. As we show, our method obtained more accurate segmentation, more cells and larger cell sizes when compared to these methods. In addition, we further used SCS to analyze sub-cellular localization of different RNAs and show that our results agree with prior knowledge further validating the accuracy of our segmentation.

2 Methods

2.1 SCS framework

Unlike prior segmentation methods that only utilized image information, SCS integrates staining image data and sequencing data from spatially barcoded spots to improve segmentation accuracy. SCS adopts a bottom-up scheme, which performs segmentation in three steps (Fig. 1). It first identifies cell nuclei from the staining images using the Watershed algorithm [19]. Spots covered by each cell nucleus are considered as belonging to the corresponding cell. Second, a transformer model which uses the gene counts information from spots, is learned to predict the gradient direction from a spot to the nucleus center of its cell. Also, for each spot, SCS predicts whether it is part of a cell or part of the extracellular matrix (background). To train the model, we used as positive examples spots within the nuclei and as negative samples spots sampled from highly confident background regions. The learned model is then applied to all spots. Finally, spots that are determined to be part of the cell are grouped by tracking the gradient flow from spots to nucleus centers.

2.2 Data preprocessing

Gene counts in each barcoded spot were collected from the original paper of Stereo-seq [6] and Seq-scope [5], and used to generate a gene expression profile vector for each spot. Each element in the profile represents the number of transcripts observed in the spot for that gene.

To identify nuclei, the paired staining image and sequencing section are first aligned to match image pixels and spot coordinates. For this, a count heat map was created for the sequencing section, where each element in the heat map contains the total number of detected transcripts in a spot. The staining image was then aligned to the heat map using transformations implemented in Spateo (https://spateo-release.readthedocs.io) [6]. This step was omitted for the Seq-scope dataset as the images have been prealigned. Watershed algorithm implemented in Spateo was next used to segment nuclei from the aligned staining image, and the mask of each individual nucleus was obtained (Supplementary Notes).

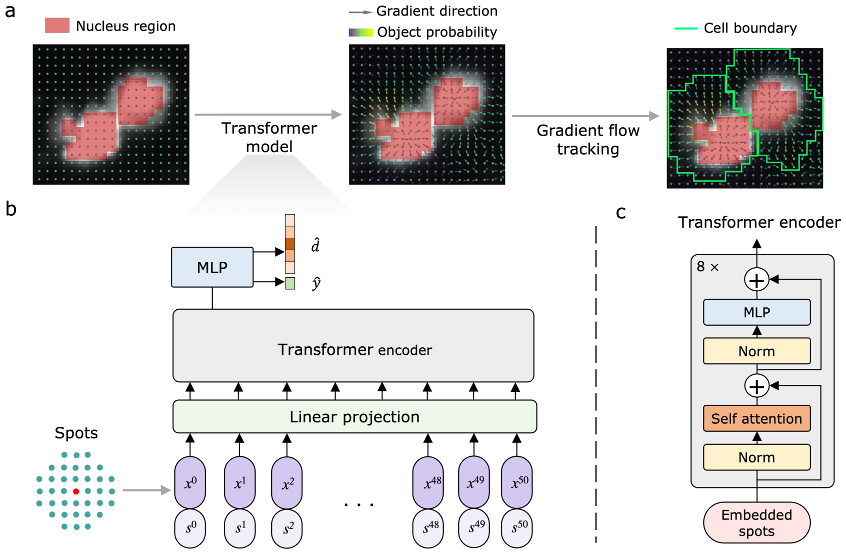

Figure 1: SCS overview. **a**, Barcoded spots (cyan dots) that reside inside a cell nuclei (red masks) are first identified by segmenting the stained image. A transformer model is next trained on these spots and some background spots to predict the gradient direction (arrow) from each spot to the center of the cell to which it belongs and the probability that it is part of a cell (yellow arrow) or part of the extracellular matrix (purple arrow). The transformer model is then applied to all other spots. A gradient flow tracking algorithm is used to segment cells by grouping spots based on their gradient prediction. **b**, The transformer model predicts for each input spot the probabilities from this spot to its cell center for 16 predefined directions ($\hat{d}$) and the probability that the spot is part of a cell ($\hat{y}$). For each spot (red dot), the transformer model aggregates information from its 50 nearest neighboring spots (cyan dots) by adaptively learning a weighting based on the spot expression (x) and relative positions (s). **c**, The structure of one transformer encoder layer, see "Methods" for details.

2.3 Transformer model for spot-level predictions

An overview of the model is depicted in Figure 1b. The model contains two components: an encoder and a classifier. The encoder maps each spot to a hidden representation z. Given the very sparse set of genes that are usually detected for each spot, we combine neighborhood gene expression information when generating the hidden representation for a spot as follows. For each spot, we use its expression profile and combine it with the profiles of the 50 nearest neighbor spots, represented as $x = (x^0, x^1, ..., x^{50})$. Each vector x^i has N dimensions which represent the number of detected transcripts in that spot for each of the N genes used in the study. The nearest neighbors are defined by using the euclidean distance between spot coordinates. To enable the model to use the relative locations of different neighbors, we include in the input to the encoder the distance from the center spot to each of the neighbor spots denoted as $s = (s^0, s^1, ..., s^{50})$, where s^i is a two dimensional vector containing distances on two axes. Given the resulting hidden vector z for a spot, a classifier then predicts the direction from the spot to the center of its cell denoted by $\hat{d}$, and the probability it belongs to part of a cell, $\hat{y}$.

***Encoder*:** The expression profiles, x, and the distance vectors, s, for the center spot and neighbor spots are first projected to a set of $D = 64$ dimensional vectors r through two separate fully connected layers followed by summation:

$$r = xW_1 + sW_2, \tag{1}$$

where $W_1 \in \mathbb{R}^{N\times D}$ and $W_2 \in \mathbb{R}^{2\times D}$ are weights matrices of the two dense layers.

We used a similar network architecture as the one proposed in the original Transformer paper [25]. In this architecture, the encoder is composed of a stack of $L = 8$ identical layers. Each layer has two blocks. The first is a self-attention block (SA):

$$\begin{aligned} [q, k, v] &= rU_{qkv}, \\ A &= \text{softmax}(\frac{qk^\top}{\sqrt{D}}), \\ \text{SA}(r) &= Av, \end{aligned} \tag{2}$$

where the tensor $U_{qkv} \in \mathbb{R}^{D\times 3D}$ projects each spot representation r^i to three D dimensional vectors, q^i (query), k^i (key), and v^i (value). The dot product between every query and all the keys, scaled by $\sqrt{D}$, passes through a softmax function to obtain attention scores, A. The attention scores are the weighting of neighbor spots, which are then multiplied by their values, v, and each spot obtains a weighed representation with D dimensions. The second block further transforms the representation using a two-layer fully connected network (MLP) with 128 and 64 nodes. GELU [27] activation function is used to introduce non-linearity. Dropout [28] with rate 0.1 is applied after each layer.

Layer normalization (LN) [29] is applied before every block, and residual connection [30] after every block. Taking together, one encoder layer can be described as:

$$\begin{aligned} r'_{l-1} &= \text{SA}(\text{LN}(r_{l-1}) + r_{l-1}, \\ r_l &= \text{MLP}(\text{LN}(r'_{l-1})) + r'_{l-1}, \end{aligned} \tag{3}$$

each encoder layer uses the output from the previous layer, r_{l-1}, and generates the next, r_l. The representation of the center spot from the last encoder layer, r_L^0, is taken as the input of the classifier.

***Classifier*:** We use the classifier to predict for a spot the gradient direction from it to its cell center and whether it is within a cell or outside a cell. The classifier has three components: (*i*) The spot representation r_L^0 is first transformed using a two-layer MLP with 1024 and 256 nodes. LN is applied before the transformation. The output from the MLP is then used as input to two fully connected layers. (*ii*) One layer connects to a softmax function that outputs the probabilities, $\hat{d}$, for each of the 16 possible directions to the cell center. (*iii*) Another layer generates a scalar output, $\hat{y}$, which is the object probability of the spot. Multi-class cross entropy is used as the loss function for the direction output:

$$L_d(d, \hat{d}) = -\sum_{k=0}^{15} d^{(k)} \log(\hat{d}^{(k)}), \tag{4}$$

where d is the one-hot encoded direction label, where the bit with 1 indicates the correct direction. The binary cross entropy loss is used for the object probability:

$$L_y(y, \hat{y}) = -y\log(\hat{y}) - (1-y)\log(1-\hat{y}), \tag{5}$$

where y is a binary label indicating whether the spot is part of a cell. Considering all the M spots in the training data, the overall loss function is:

$$\sum_i^M y^i L_d(d^i, \hat{d}^i) + L_y(y^i, \hat{y}^i). \tag{6}$$

The direction loss is masked by the object label so the background spots will not contribute to this loss.

2.4 Training data preparation for the deep model

Spots within cell nucleus regions and spots sampled from highly confident background regions are used for model training. All spots within nucleus masks are labeled as 1 for the object label. For the direction label, the center of each nucleus is first computed by averaging the X and Y coordinates of spots assigned to this nucleus from the staining image. The direction from each spot within the nucleus to the nucleus center, d, is then computed as follows:

$$d = \begin{cases} \mathrm{floor}(\frac{\arctan(\frac{Y_c-Y}{X_c-X})}{2\pi} * 16) & \text{if } Y_c - Y >= 0 \text{ and } X_c - X >= 0 \\ \mathrm{floor}(\frac{\arctan(\frac{Y_c-Y}{X_c-X})+\pi}{2\pi} * 16) & \text{if } X_c - X < 0 \\ \mathrm{floor}(\frac{\arctan(\frac{Y_c-Y}{X_c-X})+2\pi}{2\pi} * 16) & \text{if } Y_c - Y < 0 \text{ and } X_c - X >= 0 \end{cases}, \tag{7}$$

where X and Y are the coordinates of the spot for two axes, and X_c and Y_c are the coordinates of the nucleus center. If both $X_c - X$ and $Y_c - Y$ are 0s, the spot will be ignored since there is no gradient direction. Figure S3 shows how the 16 direction classes evenly divide a full circle. On the other hand, the same number of spots as nucleus spots are sampled from background and labeled as 0 for the object label. The background spots have to meet the following two criteria: (i) The staining signal intensity of the corresponding pixel is smaller or equal to 10. (ii) The euclidian distance from the spot to any of the nucleus centers is greater than 15 μm.

2.5 Cell boundary generation

Spot-level predictions are adjusted based on the locations of identified nuclei and then smoothed (Supplementary Notes), which are next used to group spots to cells. Spots with object probabilities smaller than 0.1 are determined to be background, which in practice can be determined by users according to the distribution of object probabilities of spots (Fig. S4). The direction vectors of the rest of spots are treated as gradients and the gradient flow tracking algorithm [31] is performed to segment cells. In the algorithm, the vectors flow toward a sink, which corresponds to the center of the nuclei for each cell. Starting from a spot $b = (X, Y)$, the next spot b' to which the flow is directed is selected using:

$$b' = b + \mathrm{round}(\frac{v(b)}{||v(b)||}), \tag{8}$$

where $v(b)$ is the gradient vector at b. When the angle between two consecutive steps is equal to or smaller than $\frac{\pi}{2}$, the gradient flow tracking procedure stops, and a sink is reached. The angle is computed as:

$$\arccos(\frac{v(b)}{||v(b)||}, \frac{v(b')}{||v(b')||}), \tag{9}$$

The set of spots that flow to the same sink produces an attraction basin of the sink. If the euclidean distance between two sinks is less than 3.5 spots, the attraction basins of the two sinks are combined together to obtain a larger attraction basin. An attraction basin with at least a certain number of spots is segmented as a cell (Supplementary Notes).

3 Results

3.1 Application of SCS to high-resolution spatial transcriptomics data

We applied SCS to two different high-resolution spatial transcriptomics protocols: a mouse brain dataset profiling using Stereo-seq [6] and a mouse liver dataset that utilized Seq-scope [5].

The Stereo-seq dataset captures a whole adult mouse brain slice in a single section. The barcoded spots are arranged in a grid with a distance of 0.5 μm between spots (which, given an average cell size of 20 μm in diameter means that there are roughly 1200 spots per cell). In total, this dataset profiled 26,177 genes in more than 42,000,000 spots with an average of 3.3 unique molecular identifier (UMI) counts per spot (Fig. S5). The brain slice was imaged with nucleic acid staining, allowing for segmentation of the nucleus using image based methods. Due to the large size of the assay, we cut it into partially overlapping patches with an overlap width of 60 spots for model training, each with an area of 600 μm×600 μm (1200 spots×1200 spots) resulting in 87 patches, and processed one patch at a time. To obtain stable gene compositions for spots, we merged 3×3 spots into one spot. Therefore, each merged spot aggregates RNAs detected in a 1.5 μm×1.5 μm region. We computed the most 2,000 variable genes across all the merged spots in each patch using Scanpy [32], which decides the scope of genes in the expression profiles of spots.

The Seq-scope dataset contains four tissue sections from mouse liver. The center-to-center distance of barcoded spots is similar to the distance for the Stereo-seq data, 0.5 μm on average. In total, 24,171 genes were profiled in four sections and each section contains over 570,000 spots, with 5.7 UMI counts per spot on average (Fig. S6). Instead of using nuclei staining, the Seq-scope protocol images tissues with the hematoxylin and eosin (H&E) staining. Therefore, the entire cell bodies can be segmented using the imaging data. Similar to Stereo-seq, we merged spots in each 1.5 μm×1.5 μm region into one spot. We again computed the 2,000 most variable genes across the merged spots for each section.

3.2 SCS accurately segments high-resolution spatial transcriptomics data

We first applied SCS to the Stereo-seq data. To evaluate its performance we compared SCS with Watershed cell segmentation (Supplementary Notes), and other popular segmentation methods that are based on deep learning including Cellpose, DeepCell, and StarDist. Appropriate pretrained models of the deep learning methods are used for evaluation (Supplementary Notes). Since ground truth for cell segmentation does not exist, we used a popular method for evaluating cell segmentation methods [17, 18]. In this evaluation, we compare the expression of regions where two methods (SCS and another method) agree to regions where they disagree (Fig. 2a). Specifically, for each nucleus, we found a cell mask from the segmentation of each method, which has the largest overlap with the nucleus among all the cells in the segmentation. The intersection and difference regions between the two cell masks for this nucleus were then computed. We next estimated the correlation of expression profiles between the intersection region and each of the difference regions. Since the intersection region is often dominated by the nucleus, which is easily detected by all methods (much easier to stain), we treat it as ground truth and compare the non-intersecting regions to the intersection. We expect that the more correlated the difference region is with the intersection region the more accurate the segmentation of the method (Supplementary Notes).

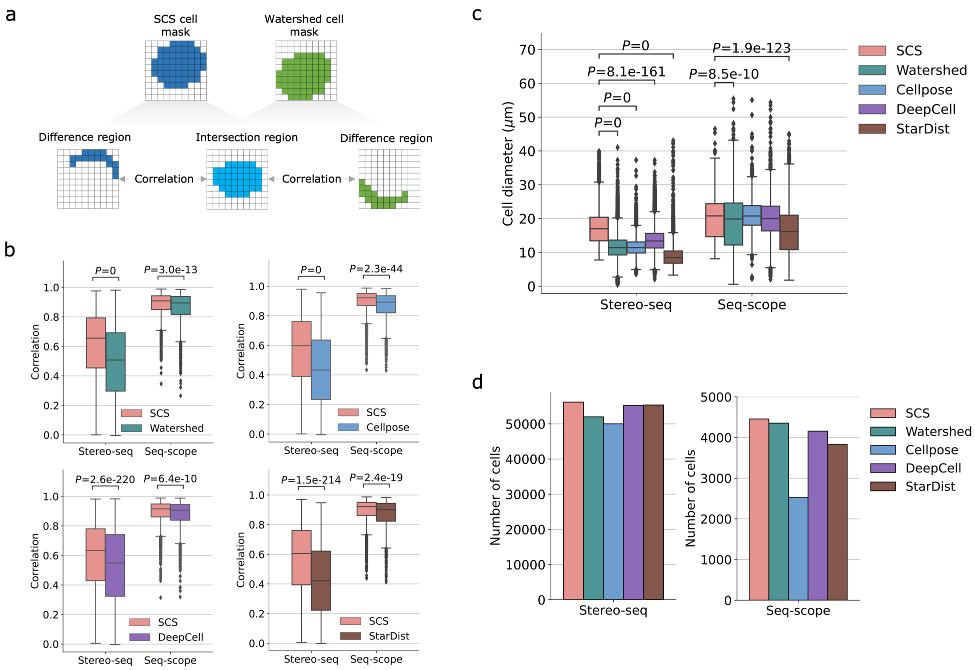

Figure 2: Evaluation and comparison of SCS. **a**, The benchmark used for evaluating the performance and for comparison of different segmentation methods. The intersection region and respective difference (unique) regions between two segmentation are calculated for each cell. The segmentation is said to have higher accuracy if its unique cell region is better correlated with the intersection region. **b**, Comparison between SCS and the four other image segmentation methods using the correlation benchmark. Each cell used for evaluation contributes one point of correlation to a boxplot. SCS achieved significantly higher segmentation accuracy than other methods on both datasets (one-sided Wilcoxon test). **c**, Comparison of the sizes of cells segmented by SCS and other methods. SCS obtained segmented cells with larger cell diameters for Stereo-seq (significant differences measured by Kruskal-Wallis tests are noted). **d**, The number of cells identified by the segmentation methods for the two datasets.

On the Stereo-seq dataset, SCS segmentation achieved an average correlation 24% higher than that of Watershed (0.61 vs. 0.49) (Fig. 2b), and at least 13% higher than those of all other deep learning segmentation methods (0.60 vs. 0.53 of DeepCell). We also tested the use of the intersection of all five methods as the ground truth and obtained similar results, Fig. S7). An ablation study shows the attention layers in transformer contributed to this high accuracy (Supplementary Notes, Fig. S8). While image based methods on nucleus staining images tend to underestimate cell sizes, SCS is able to accurately capture cytoplasm regions of cells (Fig. 3a). As a result, the segmented cells of SCS show more realistic cell diameter [33] compared to other segmentation methods (Fig. 2c), while further analysis ruled out the possibility that the high correlation is due to the larger cell sizes (Fig. S9). In addition, we observed that some cells were completely missed by image based method due to their low staining signal intensity. However, SCS can identify such cells based solely on transcriptomics data (Fig. 3b), which enabled SCS to identify at least 1.5% more cells than all the other methods (56,187 vs. 55,364 of StarDist which is the 2nd highest, Fig. 2d). Regions covered by SCS newly identified cells show weak signals in staining images, however, they are still significantly higher than the background (Fig. S10), which indicates there are indeed cells in these regions.

For the Seq-scope data, the differences were less dramatic due to the use of H&E images. Still, SCS had a higher correlation of 0.88 vs. 0.86 for Watershed and also higher correlations than all the other deep learning based methods (Fig. 2b, the comparison of five methods together is shown in Fig.

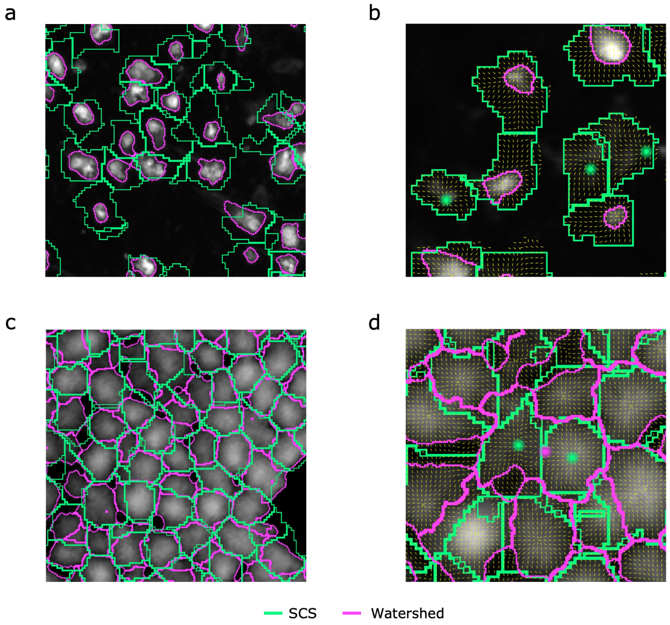

Figure 3: **a**, Comparison of segmentation results on the Stereo-seq dataset between SCS and Watershed. SCS captured cytoplasm regions of cells and thus segmented cells with larger sizes. **b**, Example of segmentation results on the Stereo-seq dataset where Watershed missed three cells due to their low staining signal intensity while SCS identified them (green dots). **c**, Comparison of segmentation results on the Seq-scope dataset between SCS and Watershed. Two segmentations show similar cell sizes but with disagreement on cell boundaries. **d**, Example of segmentation results on the Seq-scope dataset where Watershed merged two cells as one cell (pink dot) due to their unclear boundary in the image while SCS successfully segmented them (green dots).

S7). All the methods achieved higher correlations when using cell stained images instead of nucleus staining, as image based methods have an easier time determining the cell boundaries (although, as we show, using the expression observation can still improve in this case as well), which also validates our use of expression correlation for evaluating cell segmentation. Segmentation examples on this dataset are shown in Figure 3c. As the cell stained images were used, different methods show similar cell sizes. SCS segmentation show cell diameter (19.9 ± 6.4 μm) slightly larger when compared to Watershed (18.3 ± 9.4 μm) and StarDist (16.2 ± 7.6 μm) but slightly smaller when compared to Cellpose (21.1 ± 4.8 μm) and DeepCell (20.1 ± 6.0 μm) (Fig. 2c). Again, SCS determined cell size is consistent with previous findings [34]. In addition, we observed that when the boundaries of two cells are unclear in the staining images, image based methods tend to merge them, while SCS can segment them with the help of transcriptomics data (Fig. 3d) leading to at least 2.3% more cells when using SCS segmentation (4,456 vs. 4,354 of Watershed) (Fig. 2d). To further validate these findings, we identified 258 adjacent cell pairs segmented by SCS that were merged as one cell in Watershed. Among them, 189 (73.3%) pairs contain cells from two different cell types based on annotations using expression data (Supplementary Notes). This is a strong indication that these are indeed two different cells. Figure S11 shows examples of such cell pairs, where the two cells annotated with different cell types have different marker genes upregulated.

3.3 SCS enables sub-cellular analysis of spatial transcriptomics

While traditional spatial transcriptomics is a powerful method for studying cell type expression and interactions [9, 35], the use of high-resolution methods opens the door to characterizing molecular

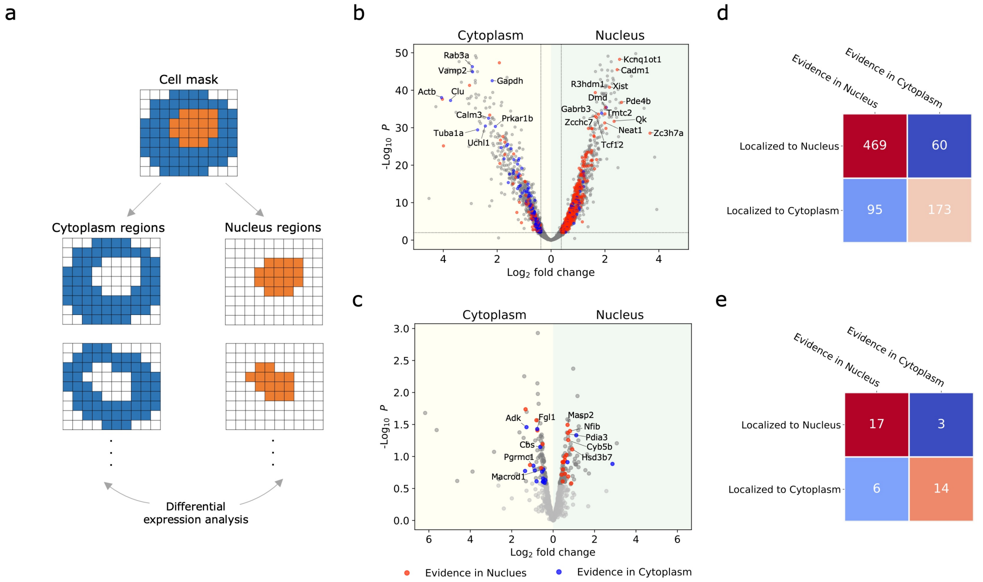

Figure 4: Sub-cellular analysis using SCS a, Identification of genes whose RNAs are differentially localized. **b** Volcano plot that shows quantitative changes in expression levels for genes between the nucleus and cytoplasm for the Stereo-seq dataset. Genes with P-values < 0.01 and fold changes greater than 1.3 were identified from each group. Genes whose RNAs have been experimentally shown to reside in the nucleus or cytoplasm are colored accordingly. **c** Volcano plot for the Seq-scope dataset. The top 100 genes with the smallest P-values were identified from each group. **d** Agreement between the experimental data and SCS assignments for Stereo-seq. **e** Agreement for the Seq-scope dataset.

heterogeneity within individual cells. This can be important to study RNA dynamics and to fully understand cellular variability in tissues [36]. We therefore used SCS to investigate how RNAs are distributed within cells. Specifically, we divided each cell into two regions, the nucleus region (identified using the staining image data) and the cytoplasm region (the rest of the cell mask identified by SCS, Fig. 4a). RNAs detected in all spots within each region were aggregated to generate a gene expression profile, which was then summarized in a region by gene matrix. Genes whose RNAs localize deferentially between two groups of regions were identified using t-test (Fig. 4b-c). Fewer genes were identified when using the Seq-scope dataset likely due to smaller number of cells and challenges involved in identification of nucleus regions from H&E images. Interestingly, in both datasets, RNAs that have been experimentally shown to reside in the nucleus or cytoplasm [37] are significantly enriched in our identified RNAs in the corresponding regions ($P = 4.0 \times 10^{-50}$ for Stereo-seq, $P = 6.4 \times 10^{-4}$ for Seq-scope, Fisher's exact test, Fig. 4d-e). For example, long non-coding RNA (lncRNA) Kcnq1ot1 is a nuclear transcript that interacts with chromatin and regulates transcription of multiple genes [38]. lncRNA Neat1 is a well known nuclear transcript that forms the core component of organelles in nucleus [39]. Both RNAs are identified as being differentially localized to nucleus in SCS segmentation (Fig. 4b). In contrast, gene Rab3a and gene Vamp2 both encode proteins that are involved in neurotransmitter release and associated with cytoplasmic vesicles [40, 41]. They are both found with high expression levels in the cytoplasm regions in SCS segmentation (Fig. 4b). These results provide further support for the ability of SCS to accurately segment cells.

The same experiments were also performed for the other segmentation methods we compared to. We found that for all other methods we obtain fewer RNAs that are experimentally verified when compared to those identified by SCS (Fig. S12). These results indicate that SCS segmentation can better help sub-cellular analysis in high-resolution spatial transcriptomics data and partially explains why our model can obtain better cell segmentation with transcriptomics data.

4 Discussion

A key step in the analysis of spatial transcriptomics data is cell segmentation. This is especially true when using the very recent sub-cellular profiling platforms. For such data, accurate cell segmentation is essential as errors in identifying cell boundaries can directly impact gene expression level quantification in cells, and further influence downstream analysis. Existing cell segmentation methods for this data only rely on the stained image, which do not fully utilize the information provided by the experiment leading to less accurate results.

In this study, we developed a cell segmentation method that combines both the staining information and the expression data to refine cell segmentation. Unlike prior methods, our method focuses on the spots but not the staining. For each spot, the method attempts to determine whether it is within a cell and if so, which cell. Once such assignments are determined, cells are naturally segmented by grouping all the spots that belong to the same cell. To enable spot assignment, SCS first aggregates information from neighboring spots and then maps spots to low-dimensional latent representations to determine their relative positions to the centers of cells. To train the supervised model, we first identify the nucleus regions which serve as ground truth and then use as positive examples spots in these regions and as negative samples spots sampled from highly confident background regions to train the transformer model, which is then applied to the whole section.

Applications of SCS to two datasets generated using very recent state-of-the-art *in situ* capturing platforms demonstrates the advantage of our method. SCS segmentation achieves higher segmentation accuracy, detects more cells, and yields more realistic cell sizes when compared to several widely used image based segmentation methods. Analysis of the spatial distribution of RNAs identified many RNAs enriched in different sub-cellular regions and these agree with experimentally confirmed results. These findings further validate SCS segmentation and suggest the ability of SCS in facilitating sub-cellular analysis on high-resolution spatial transcriptomics.

Although SCS works well for the platforms we tested, there are a number of ways to further improve it. Cell shape information can be further utilized to better obtain cell masks given the high-density spot placement. In addition, feature selection can be improved. In this study, we used the 2,000 variable genes as the input of the transformer model though better results may be obtained by a more sophisticated feature selection method. Finally, if RNA spatial distribution patterns are found to vary by cell types in the datasets [42], cell-type specific attention layers can be introduced (based on the expression profile) to allow the model to learn different patterns and better segment cells.

SCS was implemented in Python and is available for download at: https://github.com/chenhcs/SCS. While sub-cellular spatial transcriptomics is still very new, we believe that its advantages and ability to provide spatially resolved single cell information would make it very popular going forward. We hope that SCS would be a useful pre-processing method to enable all downstream analysis of such data.

References

[1] Luz Garcia-Alonso et al. “Single-cell roadmap of human gonadal development”. In: *Nature* 607.7919 (2022), pp. 540–547.

[2] Gökcen Eraslan et al. “Single-nucleus cross-tissue molecular reference maps toward understanding disease gene function”. In: *Science* 376.6594 (2022), eabl4290.

[3] Devika Agarwal et al. “A single-cell atlas of the human substantia nigra reveals cell-specific pathways associated with neurological disorders”. In: *Nature communications* 11.1 (2020), pp. 1–11.

[4] Samuel G Rodriques et al. “Slide-seq: A scalable technology for measuring genome-wide expression at high spatial resolution”. In: *Science* 363.6434 (2019), pp. 1463–1467.

[5] Chun-Seok Cho et al. “Microscopic examination of spatial transcriptome using Seq-Scope”. In: *Cell* 184.13 (2021), pp. 3559–3572.

[6] Ao Chen et al. “Spatiotemporal transcriptomic atlas of mouse organogenesis using DNA nanoball-patterned arrays”. In: *Cell* 185.10 (2022), pp. 1777–1792.

[7] Eric Lubeck et al. “Single-cell in situ RNA profiling by sequential hybridization”. In: *Nature methods* 11.4 (2014), pp. 360–361.

[8] Kok Hao Chen et al. “Spatially resolved, highly multiplexed RNA profiling in single cells”. In: *Science* 348.6233 (2015), aaa6090.

[9] Ye Yuan and Ziv Bar-Joseph. “GCNG: graph convolutional networks for inferring gene interaction from spatial transcriptomics data”. In: *Genome biology* 21.1 (2020), pp. 1–16.

[10] Dongshunyi Li, Jun Ding, and Ziv Bar-Joseph. “Identifying signaling genes in spatial single-cell expression data”. In: *Bioinformatics* 37.7 (2021), pp. 968–975.

[11] Haotian Teng, Ye Yuan, and Ziv Bar-Joseph. “Clustering spatial transcriptomics data”. In: *Bioinformatics* 38.4 (2022), pp. 997–1004.

[12] Hao Chen et al. “A unified analysis of atlas single cell data”. In: *bioRxiv* (2022).

[13] Madhav Mantri et al. “Spatiotemporal single-cell RNA sequencing of developing chicken hearts identifies interplay between cellular differentiation and morphogenesis”. In: *Nature communications* 12.1 (2021), pp. 1–13.

[14] Soumya Badrinath et al. “A vaccine targeting resistant tumours by dual T cell plus NK cell attack”. In: *Nature* (2022), pp. 1–7.

[15] Jarrod Shilts et al. “A physical wiring diagram for the human immune system”. In: *Nature* 608.7922 (2022), pp. 397–404.

[16] Patrik L Ståhl et al. “Visualization and analysis of gene expression in tissue sections by spatial transcriptomics”. In: *Science* 353.6294 (2016), pp. 78–82.

[17] Monica T Dayao et al. “Membrane marker selection for segmenting single cell spatial proteomics data”. In: *Nature communications* 13.1 (2022), pp. 1–10.

[18] Viktor Petukhov et al. “Bayesian segmentation of spatially resolved transcriptomics data”. In: *bioRxiv* (2020).

[19] Serge Beucher. “Use of watersheds in contour detection”. In: *Proceedings of the International Workshop on Image Processing*. CCETT. 1979.

[20] Carsen Stringer et al. “Cellpose: a generalist algorithm for cellular segmentation”. In: *Nature methods* 18.1 (2021), pp. 100–106.

[21] Dylan Bannon et al. “DeepCell Kiosk: scaling deep learning–enabled cellular image analysis with Kubernetes”. In: *Nature methods* 18.1 (2021), pp. 43–45.

[22] Uwe Schmidt et al. “Cell detection with star-convex polygons”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2018, pp. 265–273.

[23] Michael Y Lee et al. “CellSeg: a robust, pre-trained nucleus segmentation and pixel quantification software for highly multiplexed fluorescence images”. In: *BMC bioinformatics* 23.1 (2022), pp. 1–17.

[24] Reka Hollandi et al. “Nucleus segmentation: towards automated solutions”. In: *Trends in Cell Biology* (2022).

[25] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[26] Alexey Dosovitskiy et al. “An image is worth 16x16 words: Transformers for image recognition at scale”. In: *arXiv preprint arXiv:2010.11929* (2020).

[27] Dan Hendrycks and Kevin Gimpel. “Gaussian error linear units (gelus)”. In: *arXiv preprint arXiv:1606.08415* (2016).

[28] Nitish Srivastava et al. “Dropout: a simple way to prevent neural networks from overfitting”. In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[29] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. “Layer normalization”. In: *arXiv preprint arXiv:1607.06450* (2016).

[30] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[31] Gang Li et al. “3D cell nuclei segmentation based on gradient flow tracking”. In: *BMC cell biology* 8.1 (2007), pp. 1–10.

[32] F Alexander Wolf, Philipp Angerer, and Fabian J Theis. “SCANPY: large-scale single-cell gene expression data analysis”. In: *Genome biology* 19.1 (2018), pp. 1–5.

[33] Joshua P Gilman, Maria Medalla, and Jennifer I Luebke. “Area-specific features of pyramidal neurons—a comparative study in mouse and rhesus monkey”. In: *Cerebral Cortex* 27.3 (2017), pp. 2078–2094.

[34] Janie L Baratta et al. “Cellular organization of normal mouse liver: a histological, quantitative immunocytochemical, and fine structural analysis”. In: *Histochemistry and cell biology* 131.6 (2009), pp. 713–726.

[35] Dylan M Cable et al. "Cell type-specific inference of differential expression in spatial transcriptomics". In: *Nature methods* 19.9 (2022), pp. 1076–1087.

[36] Yue Qin et al. "A multi-scale map of cell structure fusing protein images and interactions". In: *Nature* 600.7889 (2021), pp. 536–542.

[37] Tianyu Cui et al. "RNALocate v2. 0: an updated resource for RNA subcellular localization with increased coverage and annotation". In: *Nucleic acids research* 50.D1 (2022), pp. D333–D339.

[38] Lisa Korostowski, Natalie Sedlak, and Nora Engel. "The Kcnq1ot1 long non-coding RNA affects chromatin conformation and expression of Kcnq1, but does not regulate its imprinting in the developing heart". In: (2012).

[39] Christine M Clemson et al. "An architectural role for a nuclear noncoding RNA: NEAT1 RNA is essential for the structure of paraspeckles". In: *Molecular cell* 33.6 (2009), pp. 717–726.

[40] Jan RT Van Weering, Ruud F Toonen, and Matthijs Verhage. "The role of Rab3a in secretory vesicle docking requires association/dissociation of guanidine phosphates and Munc18-1". In: *PLoS One* 2.7 (2007), e616.

[41] Natali L Chanaday and Ege T Kavalali. "Synaptobrevin-2 dependent regulation of single synaptic vesicle endocytosis". In: *Molecular biology of the cell* 32.19 (2021), pp. 1818–1823.

[42] Krysta L Engel et al. "Mechanisms and consequences of subcellular RNA localization across diverse cell types". In: *Traffic* 21.6 (2020), pp. 404–418.