Cell Systems, Volume *16*

Supplemental information

Scalable image-based visualization and alignment of spatial transcriptomics datasets

Stephan Preibisch, Michael Innerberger, Daniel León-Periñán, Nikos Karaiskos, and Nikolaus Rajewsky

Supplementary Notes

Scalable image-based visualization & alignment of spatial transcriptomics datasets

Stephan Preibisch, Michael Innerberger, Daniel León-Periñán, Nikos Karaiskos & Nikolaus Rajewsky

Supplementary Figures:

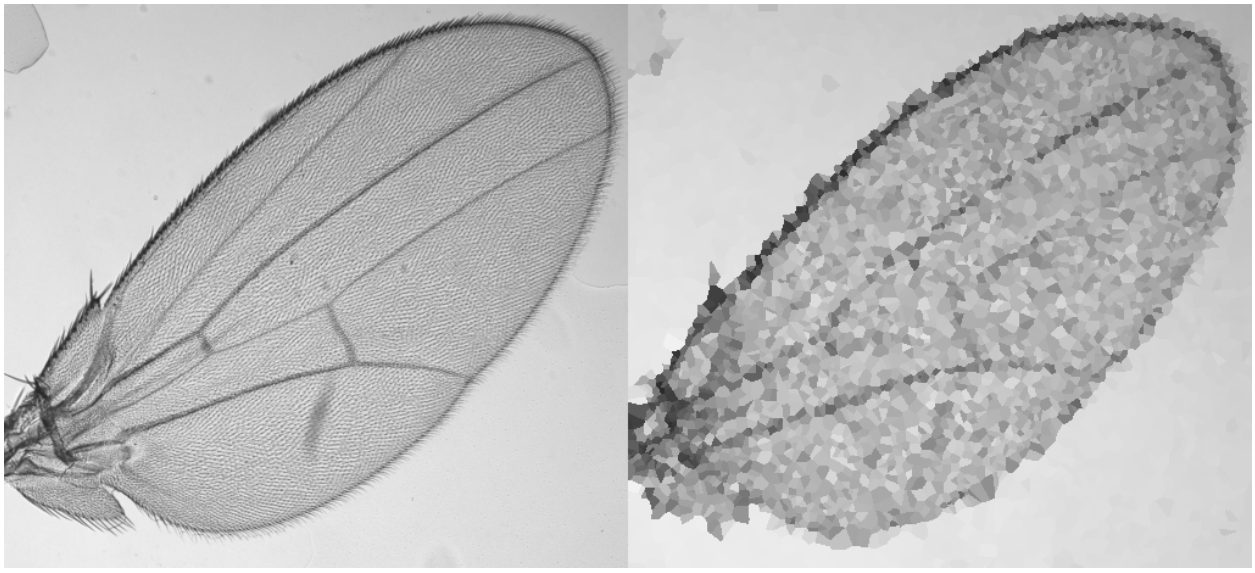
1. Comparison of regularly and irregularly spaced datasets
2. Interactive overlay of cell type onto a SlideSeq dataset
3. Alignment of a 10x Visum dataset
4. Comparison of non-rigid and regularized affine alignment
5. Applying existing machine learning segmentation software to spatial transcriptomics
6. Comparison of different point cloud rendering methods supported by STIM
7. Interactive alignment using SIFT in the STIM BigDataViewer-based GUI
8. Quantitative assessment and benchmark of pairwise alignment

Supplementary Tables:

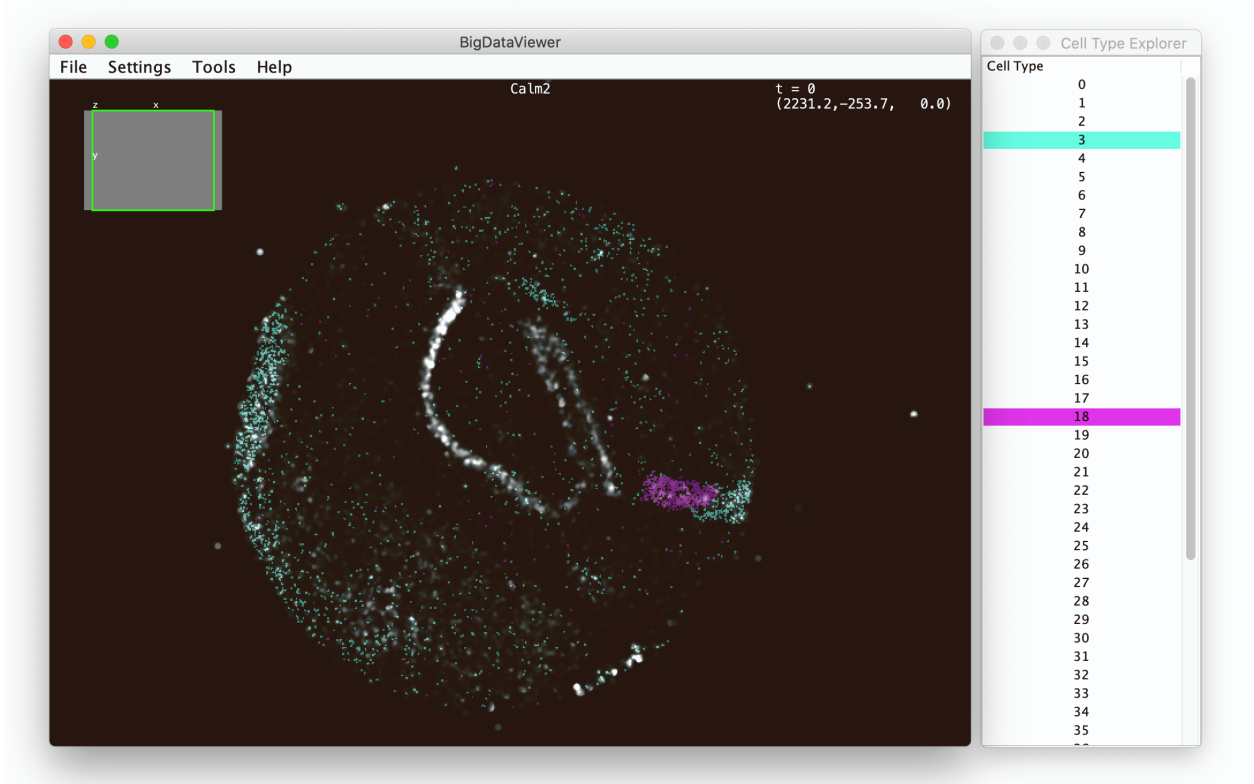
1. Comparison of 3D registration methods for Spatial Transcriptomics data

Supplementary Notes/Code Snippets:

1. Workflow for the alignment of the 10x Visium dataset
2. 3D Rendering of the SlideSeq dataset

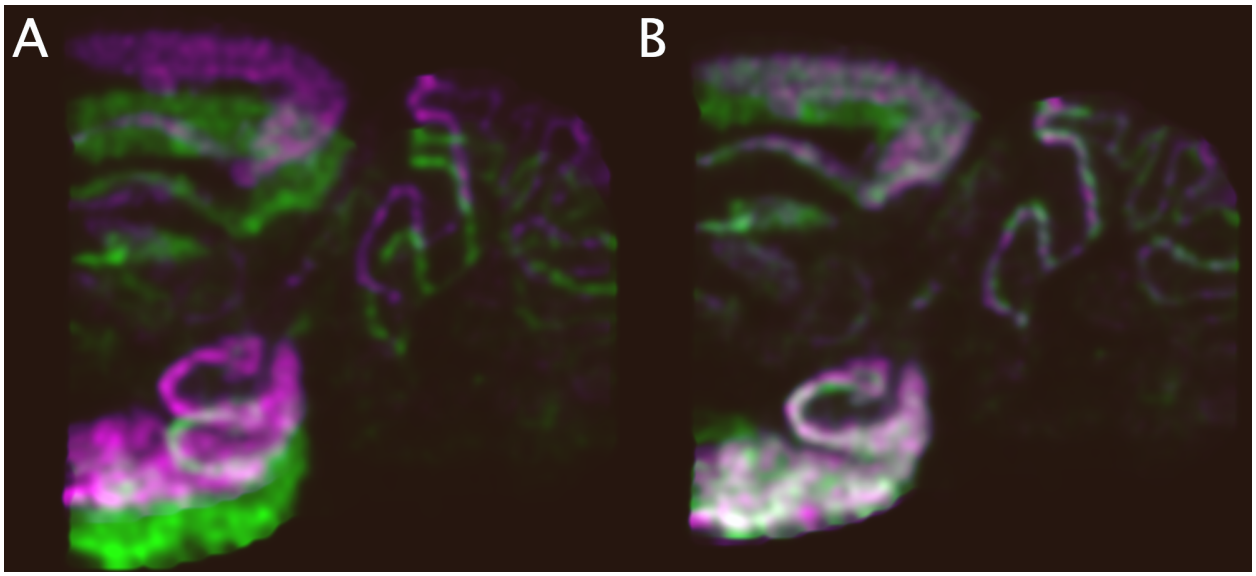


Supplementary Figure 1: *Comparison of regularly and irregularly spaced datasets.* The left panel shows a regular image of a Drosophila wing captured by a standard widefield microscope (image courtesy of Prof. Nicolas Gompel). The right panel shows was created by extracting 5000 points at random subpixel locations from the original image and rendering it using the built-in nearest neighbor interpolation for sparse datasets in ImgLib2.

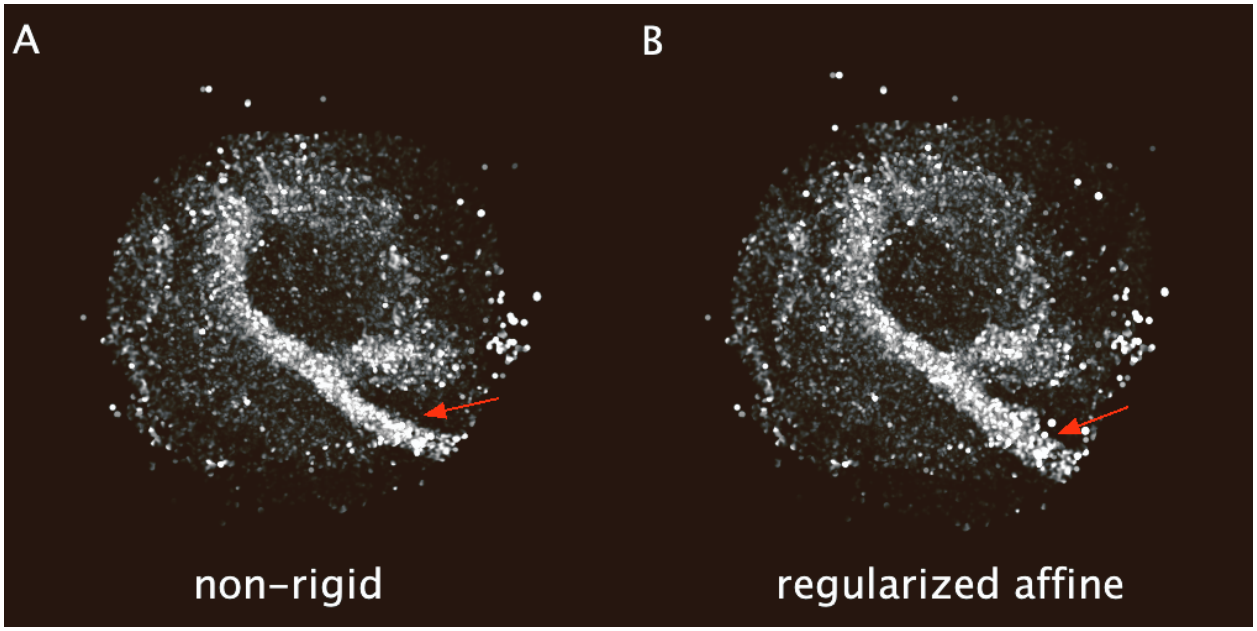


Supplementary Figure 2: *Interactive overlay of cell type onto a SlideSeq dataset.* Depicted is a screenshot using `st-bdv-view` that visualizes the gene *Calm2* (more than one gene can be displayed in parallel), together with all predicted cell types. Two of them are interactively highlighted. The screenshot was created calling:

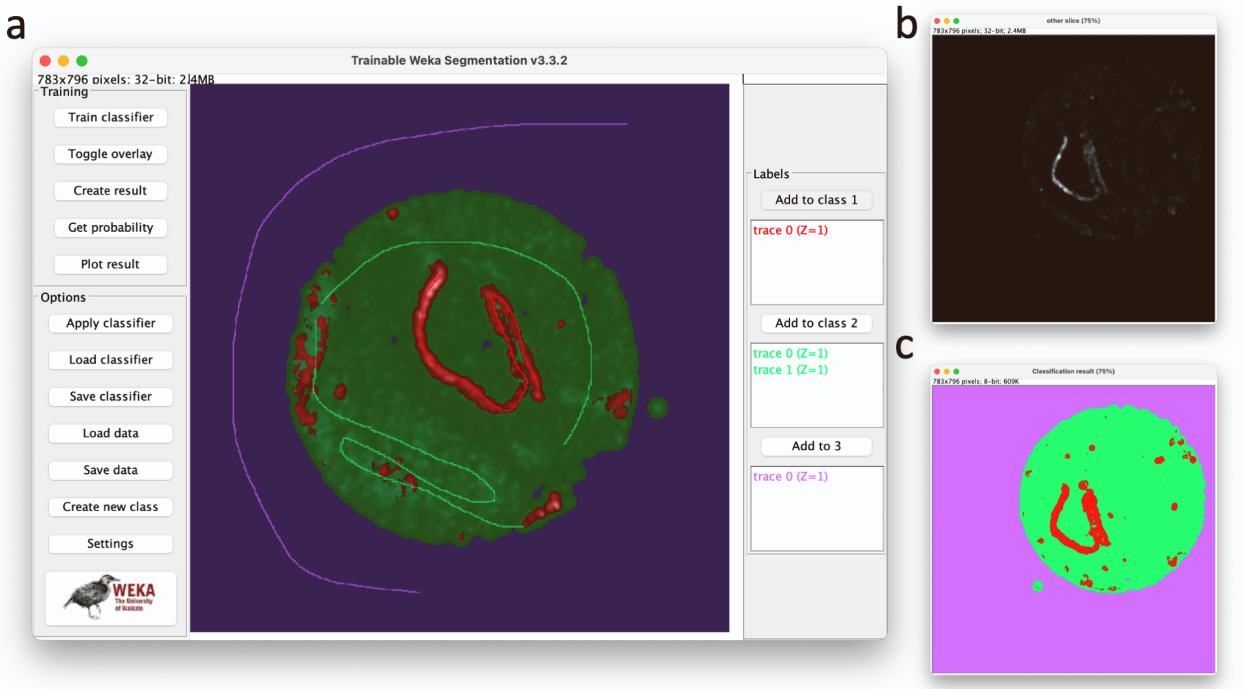
```
./st-bdv-view -i /home/slide-seq.n5/ -d Puck_180531_22 -g Calm2 -rf 1.0 -a celltype
```



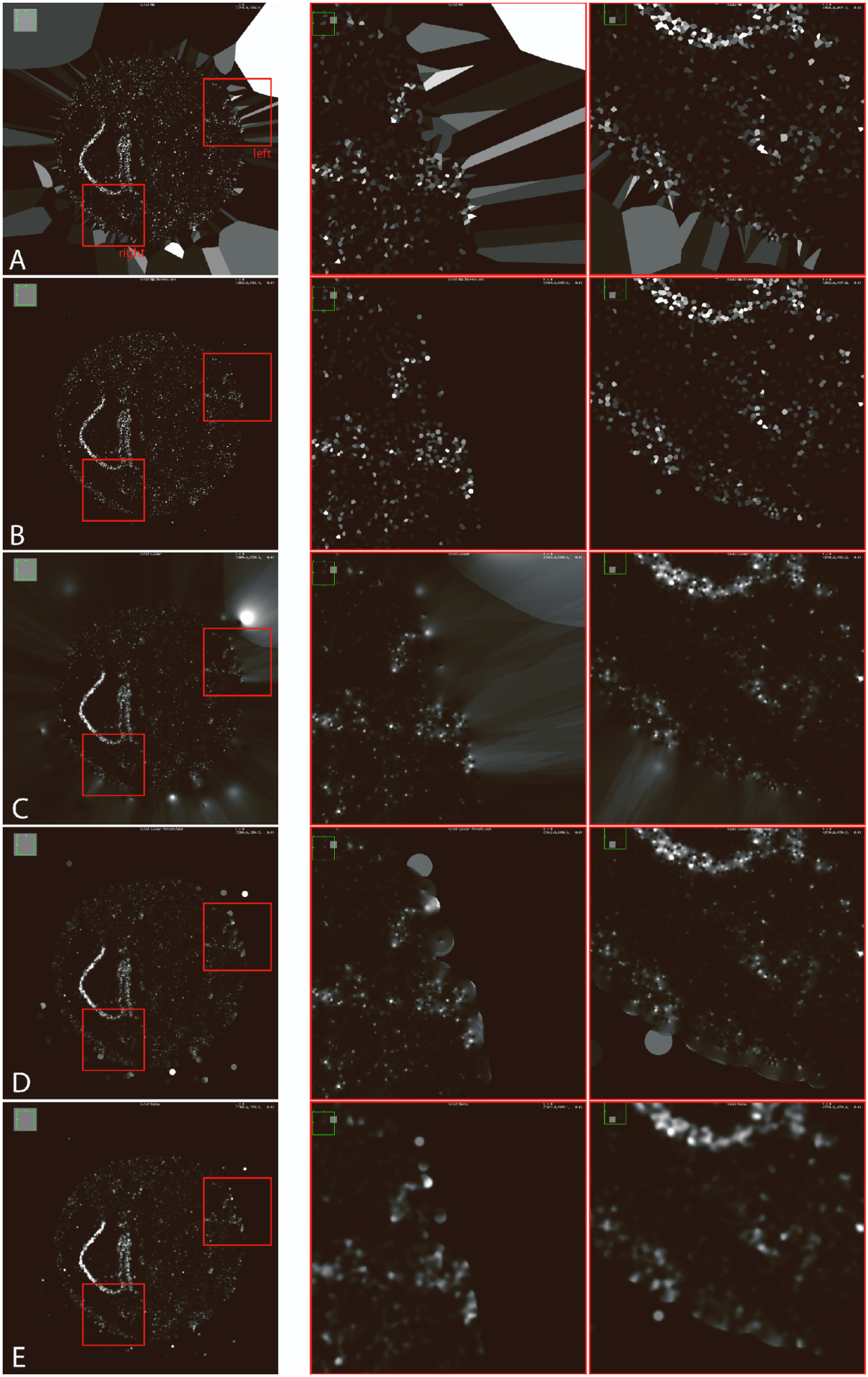
Supplementary Figure 3: *Alignment of a 10x Visium dataset.* (**A**) shows the Gaussian point cloud rendering of the gene *Calm2* before alignment, (**B**) shows the same gene after applying an affine transformation. The alignment and visualization were performed on the command line as outlined below. In this case ICP refinement did not improve the alignment quality further and was therefore omitted.



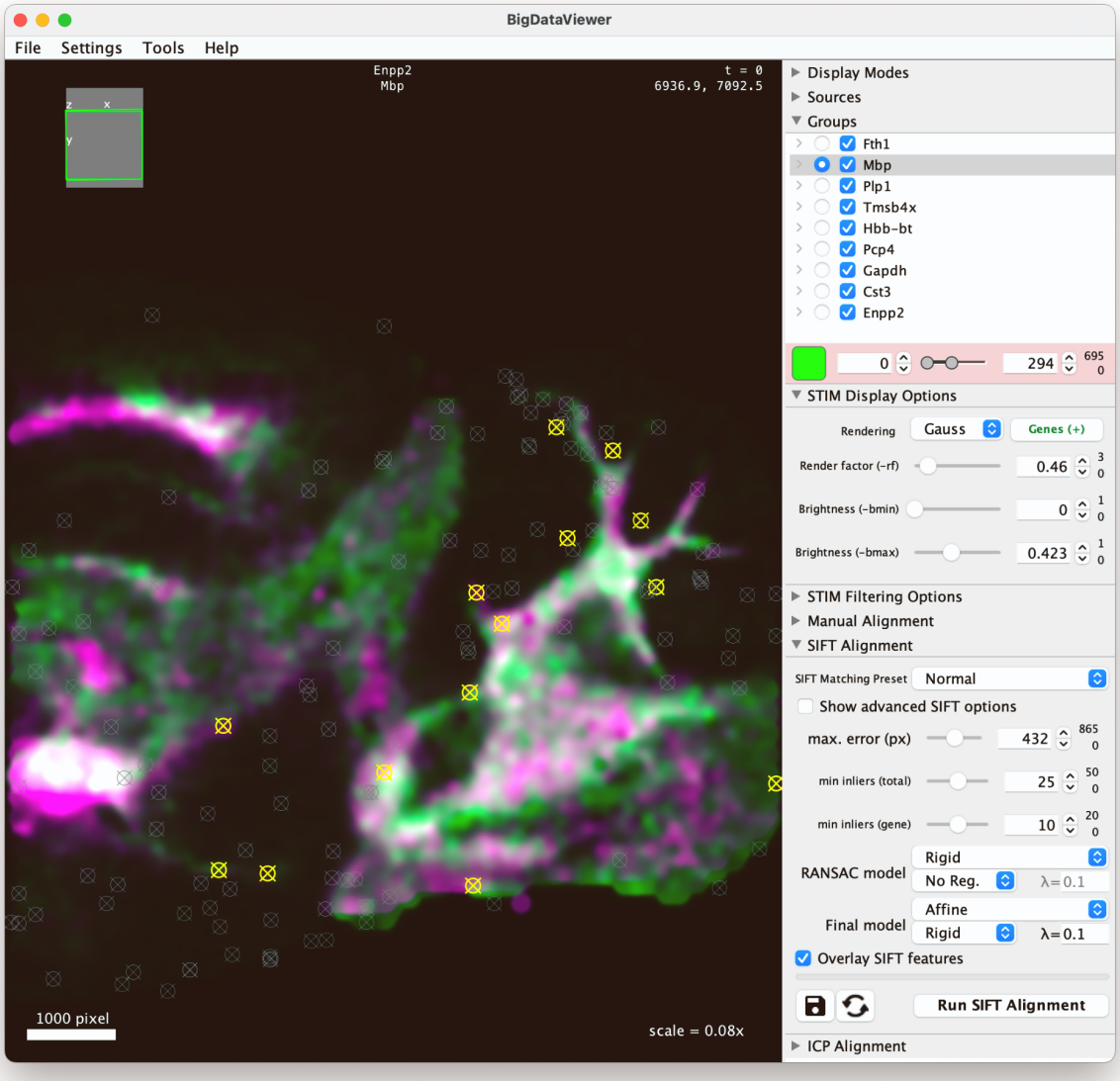
Supplementary Figure 4: *Comparison of non-rigid and regularized affine alignment.* (**A,B**) show maximum intensity projections of 8 selected slices of the SlideSeq dataset for the gene *Mbp*. The other slices were omitted due to insufficient data quality that did not allow for an automatic non-rigid alignment. (A)



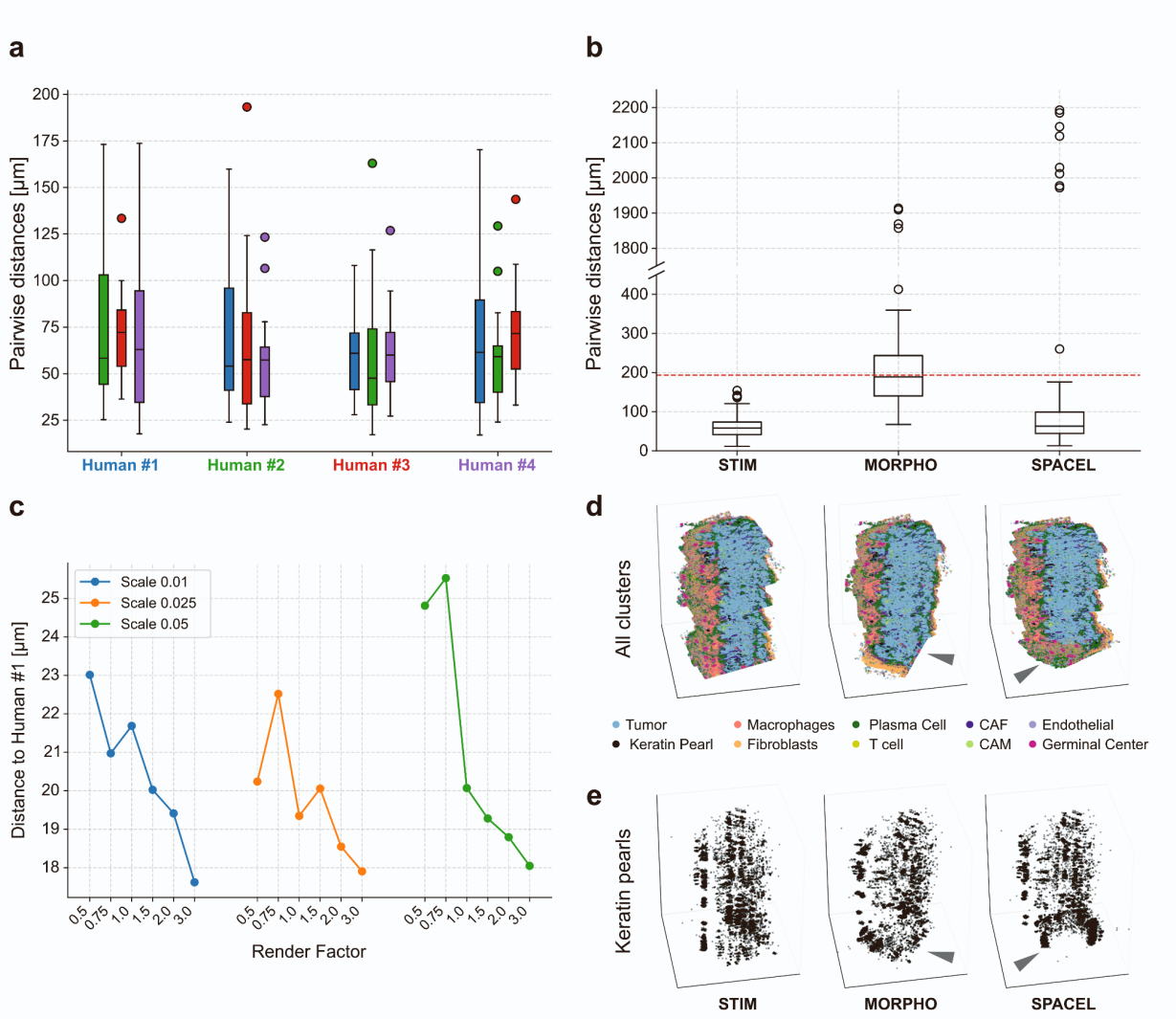
Supplementary Figure 5: *Applying existing machine learning segmentation software to spatial transcriptomics.* (**a**) illustrates how the Trainable Weka Segmentation (Random Forest-based method) is used to manually annotate a prominent visible structure in a Slide-Seq dataset using the *Calm2* gene. Note that the annotation effort is limited to the clearly visible lines belonging to each class. (**b**) shows another slice of the dataset to whom the trained classifier was applied (**c**).


left
right
A
B
C
D
E

Supplementary Figure 6: *Comparison of different point cloud rendering methods supported by STIM.* Each row shows a different rendering method for the same field of view of the gene *Calm2* on the left side, and two zoom-ins on the right side. (**A**) illustrates nearest-neighbor rendering. (**B**) show nearest neighbor rendering with a cut-off after the median distance between all sequenced locations (~16 distance units). (**C**) shows distance-weighted rendering using a maximum number of 20 neighbors and quadratic distance weight. (**D**) shows the same distance-weighted rendering, but with an additional cut-off at 5x the median distance between all sequenced locations. (E) shows the Gaussian-distance weighted rendering with a sigma equivalent to the median distance between all sequenced locations. This rendering method was used for all experiments, figures and videos in this publication. The code for creating these representations can be found in the `net.imglib2` package.



Supplementary Figure 7: *Interactive alignment using SIFT in the STIM BigDataViewer-based GUI.* Corresponding SIFT features were identified using a rigid model and the final transformation computed from the features was a rigidly regularized affine model. Yellow crosshairs indicate automatically identified corresponding features in the visible gene (*Mbp*), while gray crosshairs show corresponding features in the other genes that were also used for alignment (*Fth1*, *Plp1-Enpp2*). The GUI allows to interactively modify visualization parameters, filtering, manual alignment, SIFT alignment and ICP alignment (Note: ICP iterations are interactively updated).



Supplementary Figure 8: *Quantitative assessment and benchmark of pairwise alignment.* **(a)** Pairwise error between aligned sections of the metastatic lymph node comparing manual alignments by four different users. **(b)** Pairwise alignment error for STIM, Morpho, and SPACEL, with the maximum inter-human variability shown as a red line. **(c)** Effect of STIM's scale and render factor parameters on alignment quality, shown as pairwise error between STIM and Human #1. **(d)** Three-dimensional visualization of the lymph node dataset showing cell type annotations after alignment by STIM, Morpho, and SPACEL. **(e)** Three-dimensional visualization of keratin pearl structures after alignment by each method. In (d) and (e), gray arrowheads indicate groups of sections that are misaligned.

Method	Visualization	Global	Non-Rigid	CPU time	Memory	Automated
STIM	✓✓	✓	✗	**~181 min**	**~6.8 GB**	✓
Morpho [1]	✓	✗	✓	>1 day (DNF)	≥416 GB	✓
GPSA [2]	✗	✗	✓	>1 day (DNF)	≥162 GB	✓
PASTE2 [3]	✗	✗	✓	>1 day (DNF)	≥231 GB	✓
SPACEL [4]	✗	✗	✓	~433 min	~18 GB	✗ (clusters)
ST-GEARS [5]	✓	✗	✓	>1 day (DNF)	≥358 GB	✗ (clusters)
Sc3d [6]	✓ (napari)	✗	✗	~191 min	~24 GB	✗ (clusters)
Semla [7]	✓	✗	✗	N/A	N/A	✗ (landmarks)
STAlign [8]	✗	✗	✓	N/A	N/A	✗ (landmarks)
Eggplant [9]	✗	✗	✓	N/A	N/A	✗ (landmarks)

Supplementary Table 1: Comparison of 3D registration methods for Spatial Transcriptomics data. We selected properties that are important for aligning the Open-ST metastatic lymph node, a large dataset containing ~1 million cells (19 sections spanning ~3,000x4,000x350 μm^3, ~60,000 cells/section). For methods using specific genes for alignment, we selected the top 10 highly variable genes (scanpy, using flavor 'seurat' with default parameters). *Visualization*: seamless interactive exploration of the data in 3D, i.e., via custom visualization tools or functions connecting to existing ones. ✓✓: visualization and interactive alignment are available. ✓: only visualization tools are available. ✗: visualization tools not available. *Global*: the model's ability to perform global registration of stacks, to regularize the propagation of errors after pairwise alignment across the final z-stack. *Non-rigid:* whether the method provides non-rigid transformation models. *CPU Time*: excluding time required for converting the dataset, manual selection of points, or tasks not strictly related to alignment. *Memory:* peak RAM used during alignment, excluding preprocessing tasks. *Automated*: the method does not require manual selection of *landmarks,* or extensive preprocessing and annotation of cell types or *regions*. Morpho: from *spateo*; Sc3D: using the sc3d mode (instead of wrapper for PASTE). For the methods based on region selection, we used the transcriptomic cluster identities from the original publication. Under *CPU time* and *Memory*, the best method is **bold**, and the second best is underlined. The following software versions were used: STIM (0.3.0) PASTE2 (1.0.1), spateo-release (1.1.0), GPSA (0.8), SPACEL (1.1.7), ST-GEARS (1.0.0). The benchmark was run on a quad-socket server equipped with 4x Intel(R) Xeon(R) Platinum 8280 CPUs and 4,227 GB of RAM. All tools were run using a single thread, to discard additional sources of overhead. Jobs requiring more than 1 day to run were terminated, thus did not finish (DNF).

Workflow for the alignment of the 10x Visium dataset

```
./st-resave
        -i /home/10x-Visium/section1_locations.csv,/home/10x-Visium/section1_reads.csv,sec1
        -i /home/10x-Visium/section2_locations.csv,/home/10x-Visium/section2_reads.csv,sec2
        -c /home/10x-Visium.n5
```

```
./st-explorer -i /home/10x-Visium.n5/
./st-align-pairs -c /home/10x-Visium.n5/ -n 100 -rf 0.5 --maxEpsilon 100 --minNumInliersGene 30
./st-align-global -c /home/10x-Visium.n5/ --absoluteThreshold 100 -rf 0.5 --lambda 0.0 --skipICP

./st-render -i /home/10x-Visium.n5/ -g Calm2 -rf 0.6 -s 0.1
./st-render -i /home/10x-Visium.n5/ -g Calm2 -rf 0.6 -s 0.1 -ignoreTransforms
```

3D Rendering of the Slide-Seq dataset

In order to create the 3D rendering shown in **Fig. 2** (which is also **Supplementary Video 1**), we exported three genes of the aligned SlideSeq dataset in low (1609 x 1771 x 256 px) and high resolution (3217 x 3540 x 511 px) and merged both into an RGB image (*Calm2* is shown in white, *Ptdgs* in green, and *Mbp* in red).

We then created a 3D projection using the Fiji command "`Image > Stacks > 3D project`" for both images, which creates a visually pleasing animation of both images.

In order to create the zoom-in effect we wrote a script that smoothly interpolates between the two datasets as they rotate, which can be found in the `examples.MakeMovie` class in the STIM repository (https://github.com/PreibischLab/STIM/blob/master/src/main/java/examples/MakeMovie.java).

Bibliography

1. Qiu, X., Zhu, D.Y., Yao, J., Jing, Z., Zuo, L., Wang, M., Min, K.H. (Joseph), Pan, H., Wang, S., Liao, S., et al. (2022). Spateo: multidimensional spatiotemporal modeling of single-cell spatial transcriptomics. BioRxiv. 10.1101/2022.12.07.519417.
2. Jones, A., Townes, F.W., Li, D., and Engelhardt, B.E. (2023). Alignment of spatial genomics data using deep Gaussian processes. Nat. Methods *20*, 1379–1387. 10.1038/s41592-023-01972-2.
3. Liu, X., Zeira, R., and Raphael, B.J. (2023). Partial alignment of multislice spatially resolved transcriptomics data. Genome Res. *33*, 1124–1132. 10.1101/gr.277670.123.
4. Xu, H., Wang, S., Fang, M., Luo, S., Chen, C., Wan, S., Wang, R., Tang, M., Xue, T., Li, B., et al. (2023). SPACEL: deep learning-based characterization of spatial transcriptome architectures. Nat. Commun. *14*, 7603. 10.1038/s41467-023-43220-3.
5. Xia, T., Hu, L., Zuo, L., Zhang, Y., Xu, M., Lu, Q., Zhang, L., Cao, L., Pan, T., Zhang, B., et al. (2023). ST-GEARS: Advancing 3D Downstream Research through Accurate Spatial Information Recovery. BioRxiv. 10.1101/2023.12.09.570320.
6. Sampath Kumar, A., Tian, L., Bolondi, A., Hernández, A.A., Stickels, R., Kretzmer, H.,

Murray, E., Wittler, L., Walther, M., Barakat, G., et al. (2023). Spatiotemporal transcriptomic maps of whole mouse embryos at the onset of organogenesis. Nat. Genet. *55*, 1176–1185. 10.1038/s41588-023-01435-6.

7. Larsson, L., Franzén, L., Ståhl, P.L., and Lundeberg, J. (2023). Semla: a versatile toolkit for spatially resolved transcriptomics analysis and visualization. Bioinformatics *39*. 10.1093/bioinformatics/btad626.
8. Clifton, K., Anant, M., Aihara, G., Atta, L., Aimiuwu, O.K., Kebschull, J.M., Miller, M.I., Tward, D., and Fan, J. (2023). STalign: Alignment of spatial transcriptomics data using diffeomorphic metric mapping. Nat. Commun. *14*, 8123. 10.1038/s41467-023-43915-7.
9. Andersson, A., Andrusivová, Ž., Czarnewski, P., Li, X., Sundström, E., and Lundeberg, J. (2021). A Landmark-based Common Coordinate Framework for Spatial Transcriptomics Data. BioRxiv. 10.1101/2021.11.11.468178.