

SPAC: A Python Package for Spatial Single-Cell Analysis of Multiplexed Imaging

Fang Liu  ¹, Rui He  ², Andrei Bombin  ³, Ahmad B. Abdallah ⁴, Omar Eldaghar ⁴, Tommy R. Sheeley ⁴, Sam E. Ying ⁴, and George Zaki  ^{1¶}

¹ Frederick National Laboratory for Cancer Research, United States of America ² Essential Software Inc., United States of America ³ Axle Informatics, United States of America ⁴ Purdue University, United States of America ¶ Corresponding author

DOI: [10.21105/joss.08787](https://doi.org/10.21105/joss.08787)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: Adam Tyson  

Reviewers:

- [@jluethi](#)
- [@kevinyamauchi](#)

Submitted: 15 April 2025

Published: 14 January 2026

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).

Summary

Multiplexed immunofluorescence microscopy captures detailed, spatially resolved measurements of multiple biomarkers simultaneously. These measurements reveal tissue composition and cellular interactions *in situ* at the single-cell level. The growing scale and dimensional complexity of these datasets demand reproducible, comprehensive, and user-friendly computational tools. To address this need, we developed SPAC (SPAtial single-Cell analysis), a Python-based package and a corresponding Shiny application within an integrated, modular SPAC ecosystem designed specifically for biologists without extensive coding expertise. Following image segmentation and extraction of spatially resolved single-cell data, SPAC streamlines downstream phenotyping and spatial analysis, facilitating the characterization of cellular heterogeneity and spatial organization within tissues. Through scalable performance, specialized spatial statistics, highly customizable visualizations, and seamless workflows from dataset to insights, SPAC significantly lowers the barriers to sophisticated spatial analyses.

Statement of Need

Multiplexed protein imaging (e.g., CODEX ([Goltsev et al., 2018](#)), CyCIF ([Lin et al., 2018](#)), MxIF ([Gerdes et al., 2013](#))) yields gigabyte-scale, single-cell datasets with millions of cells per study. Extracting biological signal from these datasets requires reproducible workflows that span preprocessing, phenotyping, spatial statistics, and interactive visualization.

The spatial omics ecosystem includes general frameworks for single-cell data and spatial analysis ([AnnData](#) ([Virshup et al., 2024](#)), [Scanpy](#) ([Wolf et al., 2018](#)), [Squidpy](#) ([Palla et al., 2022](#))) and toolkits focused on multiplex imaging and spatial proteomics ([SCIMAP](#) ([Nirmal & Sorger, 2024](#))), alongside methods aimed primarily at spatial transcriptomics ([Seurat](#) ([Hao et al., 2021](#)), [Giotto Suite](#) ([Chen et al., 2025](#)), [GraphST](#) ([Long et al., 2023](#)), [Bento](#) ([Mah et al., 2024](#))). These packages collectively provide data structures, clustering, dimensionality reduction, neighborhood graphs, and spatial interaction metrics, and they are widely adopted for programmatic analyses.

Despite this progress, biologists without coding expertise face steep barriers when conducting end-to-end analyses independently and iteratively. Projects often require stitching together multiple libraries and ad hoc scripts, which slows hypothesis testing, complicates reproducibility, and makes figure generation brittle and hard to reuse at scale.

SPAC is a Python package for downstream analysis of single-cell multiplexed imaging data following segmentation. It ingests per-cell tables (CSV/H5AD/tabular) with spatial coordinates and marker intensities and exposes a reproducible, AnnData-compliant Python API; all analysis

artifacts and figures are preserved to support sharing, reruns, and downstream reuse. As the core analytical layer of the broader SPAC ecosystem (Liu et al., 2025), the package connects to interactive pipelines and real-time dashboards on enterprise platforms through a modular, layered architecture. This design enables large-scale analyses without requiring users to write code. Its purpose is to lower the barrier for non-programmers while preserving rigorous, reusable workflows. For example, SPAC uses biologist-friendly terminology (e.g., “cells,” “features,” “tables,” “associated tables,” and “annotations”) and clear messaging so users can interpret results and diagnostics without a bioinformatics analyst’s assistance.

To address real-time scalability for datasets exceeding 10 million cells, SPAC integrates optimized numerical routines from NumPy’s compiled, C-based backend (Harris et al., 2020). These optimizations make common plotting and summarization tasks (e.g., histograms, box plots) substantially faster—more than 5 \times faster than comparable Seaborn-based workflows (Waskom, 2021) in our tests—reducing typical runtimes from tens of seconds to a few seconds. This responsiveness supports iterative exploratory data analysis and figure generation on very large cohorts without sacrificing reproducibility.

SPAC provides customizable visualization methods. Pinning colors to annotations maintains consistent color mapping across figures and sessions. Interactive spatial plots (Plotly) (Plotly Technologies Inc., 2015) allow users to toggle features (e.g., biomarkers) on and off and to switch among multiple annotations simultaneously, enabling intuitive exploration of spatial relationships and patterns. Stratified and gridded plotting with optimized statistical summaries makes subgroup comparisons (e.g., across conditions or phenotypes) clear within a single view.

SPAC builds directly on community standards and methods but contributes a coherent foundation that enhances key functions for spatial analyses and visualization. To highlight a few examples, SPAC enhances core analyses (e.g., nearest-neighbor metrics leveraging SCIMAP’s spatial-distance utilities) with flexible, built-in plotting and native support for stratification, subsetting, and faceting. Users can fine-tune plot aesthetics and export layouts, making it straightforward to produce consistent, publication-ready figures. SPAC strengthens the functionality of existing packages: SPAC implements a phenotype-pair-specific Ripley’s L (Kiskowski et al., 2009) that treats one phenotype as “centers” and another as “neighbors,” and applies radius-dependent guard-zone edge correction by excluding center cells whose r-neighborhood intersects the ROI boundary. Unlike generalized implementations (e.g., Squidpy), this design reduces edge-induced inflation at larger radii and yields more reliable, interpretable L(r) curves for concrete cell-cell interactions. SPAC also adds new features; for instance, it implements a neighborhood profile via a KDTree-based approach, quantifying the distribution of neighboring cell phenotypes within user-defined distance bins. The resulting three-dimensional array encodes, for every cell and every distance bin, the count of neighboring cells of each phenotype. This array captures the local cellular microenvironment within the AnnData object and supports downstream dimensionality-reduction methods such as spatial UMAP (Giraldo et al., 2021).

Structure and Implementation

The SPAC package is available on [GitHub](#) and can be installed locally via Conda or Docker images. It comprises five modules that streamline data processing, transformation, spatial analysis, visualization, and common utilities. All modules are interoperable, forming a cohesive workflow (Figure 1).

At the architectural level, the SPAC package serves as a foundation layer that builds on community libraries while underpinning interactive, no-code surfaces. By decoupling the analysis layer from any specific UI, a single, tested codebase can power both web-hosted pipelines and interactive dashboards, eliminating duplicated analysis logic. This separation keeps the package focused on analytical semantics and performance while allowing heterogeneous front ends to reuse the same analysis primitives. For instance, an interactive Shiny for Python dashboard ([GitHub](#)) is hosted on Posit Connect and is accessible via a web browser at [SPAC](#).

Interactive Visualization (Liu et al., 2025). This dashboard provides a no-code interface to SPAC's real-time visualization capabilities, featuring multiple tool modules and "Getting Started" tutorials that are publicly available to researchers without programming expertise.

SPAC adheres to enterprise-level software engineering standards, featuring extensive unit testing, rigorous edge-case evaluation, comprehensive logging, and clear, context-rich error handling. These practices ensure reliability, adaptability, and ease of use across various deployment environments, including interactive Jupyter notebooks, analytic platforms (e.g., Code Ocean (Code Ocean, 2019), Palantir Foundry (Palantir Technologies, 2003)), Galaxy (Jalili et al., 2020), and real-time dashboards. Emphasizing readability and maintainability, SPAC provides a versatile and enhanced analytical solution for spatial single-cell analyses. To date, SPAC has been used in the analysis of more than eight datasets comprising over 30 million cells across diverse studies (Keretsu et al., 2025).

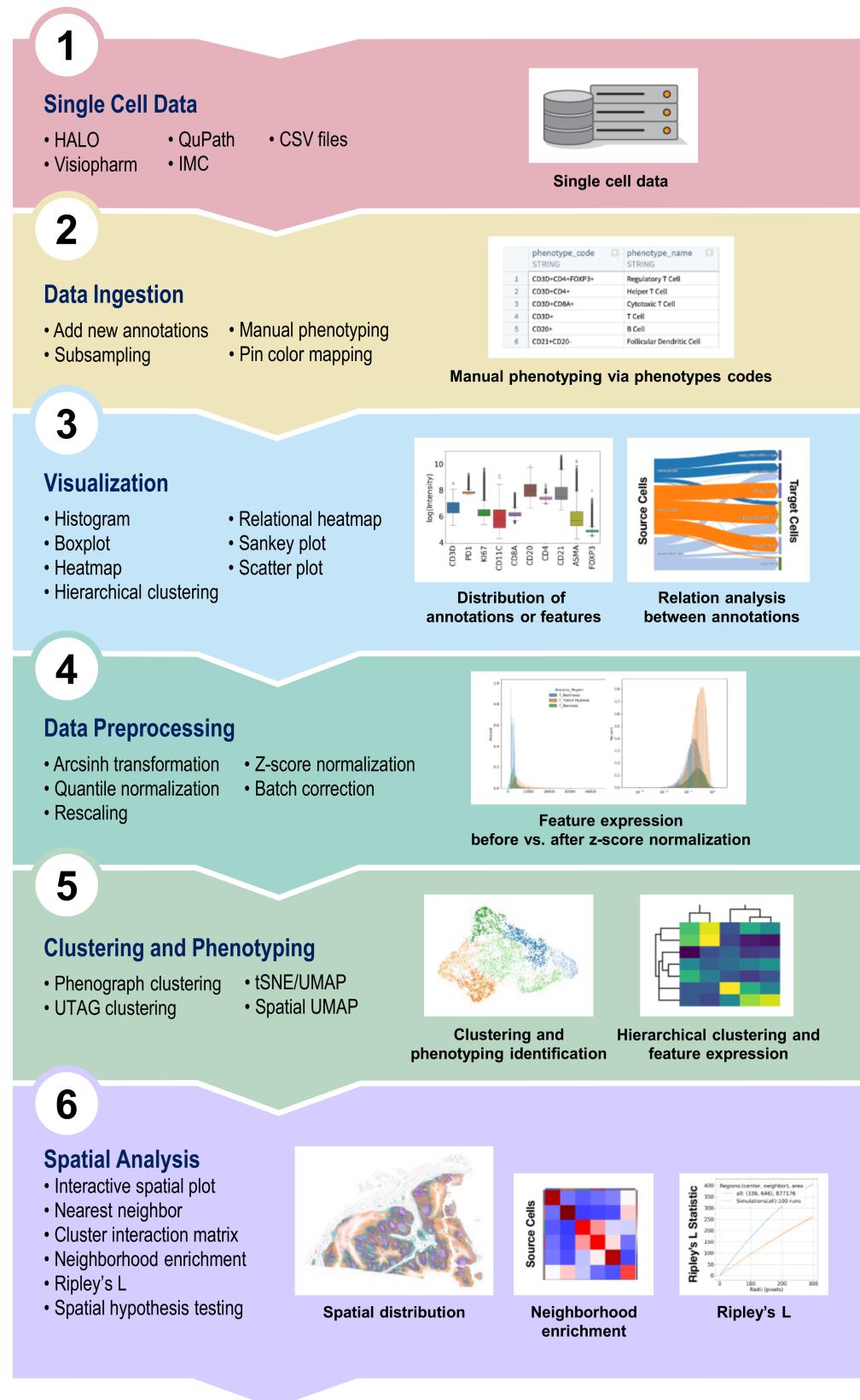


Figure 1: Overview of the SPAC workflow. The schematic presents an integrated pipeline for spatial single-cell analysis. Segmented cell data with spatial coordinates from various imaging platforms are ingested, normalized, clustered, phenotyped, and analyzed spatially to assess cell distributions and interactions while maintaining consistent data lineage.

Acknowledgements

We thank our collaborators at the National Cancer Institute, Frederick National Laboratory, the Purdue Data Mine program, and the single-cell and spatial imaging communities for their essential contributions and resources.

References

- Chen, J. G., Chávez-Fuentes, J. C., O'Brien, M., Xu, J., Ruiz, E., Wang, W., Amin, I., Sheridan, J., Shin, S. C., Hasyagar, S. V., Sarfraz, I., Guckhool, P., Sistig, A., Jarzabek, V., Yuan, G.-C., & Dries, R. (2025). Giotto suite: A multiscale and technology-agnostic spatial multiomics analysis ecosystem. *Nature Methods*, 22, 2052–2064. <https://doi.org/10.1038/s41592-025-02817-w>
- Code Ocean. (2019). *Code Ocean*. <https://codeocean.com/>
- Gerdes, M. J., Sevinsky, C. J., Sood, A., Adak, S., Bello, M. O., Bordwell, A., Can, A., Corwin, A., Dinn, S., Filkins, R. J., Hollman, D., Kamath, V., Kaanumalle, S., Kenny, K., Larsen, M., Lazare, M., Li, Q., Lowes, C., McCulloch, C. C., ... Ginty, F. (2013). Highly multiplexed single-cell analysis of formalin-fixed, paraffin-embedded cancer tissue. *Proceedings of the National Academy of Sciences of the United States of America*, 110(29), 11982–11987. <https://doi.org/10.1073/pnas.1300136110>
- Giraldo, N. A., Berry, S., Becht, E., Ates, D., Schenk, K. M., Engle, E. L., Green, B., Nguyen, P., Soni, A., Stein, J. E., Succaria, F., Ogurtsova, A., Xu, H., Gottardo, R., Anders, R. A., Lipson, E. J., Danilova, L., Baras, A. S., & Taube, J. M. (2021). Spatial UMAP and image cytometry for topographic immuno-oncology biomarker discovery. *Cancer Immunology Research*, 9(11), 1262–1269. <https://doi.org/10.1158/2326-6066.CIR-21-0015>
- Goltsev, Y., Samusik, N., Kennedy-Darling, J., Bhate, S., Hale, M., Vazquez, G., Black, S., & Nolan, G. P. (2018). Deep profiling of mouse splenic architecture with CODEX multiplexed imaging. *Cell*, 174(4), 968–981.e15. <https://doi.org/10.1016/j.cell.2018.07.010>
- Hao, Y., Hao, S., Andersen-Nissen, E., Mauck, W. M., Zheng, S., Butler, A., Lee, M. J., Wilk, A. J., Darby, C., Zager, M., Hoffman, P., Stoeckius, M., Papalexi, E., Mimitou, E. P., Jain, J., Srivastava, A., Stuart, T., Fleming, L. M., Yeung, B., ... Satija, R. (2021). Integrated analysis of multimodal single-cell data. *Cell*, 184(13), 3573–3587.e29. <https://doi.org/10.1016/j.cell.2021.04.048>
- Harris, C. R., Millman, K. J., Walt, S. J. van der, Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., Kerkwijk, M. H. van, Brett, M., Haldane, A., Río, J. F. del, Wiebe, M., Peterson, P., ... Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362. <https://doi.org/10.1038/s41586-020-2649-2>
- Jalili, V., Afgan, E., Gu, Q., Clements, D., Blankenberg, D., Goecks, J., Taylor, J., & Nekrutenko, A. (2020). The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2020 update. *Nucleic Acids Research*, 48(W1), W395–W402. <https://doi.org/10.1093/nar/gkaa554>
- Keretsu, S., Hana, T., Lee, A., Kedei, N., Malik, N., Kim, H., Spurgeon, J., Khayrullina, G., Ruf, B., Hara, A., Coombs, M., Watowich, M., Hari, A., Ford, M. K. B., Sahinalp, C., Watanabe, M., Zaki, G., Gilbert, M. R., Cimino, P. J., ... Terabe, M. (2025). Characterization of MAIT cells in glioblastoma reveals a potential immunosuppressive role through a MAIT-Neutrophil axis. *Neuro-Oncology Advances*, vdaf173. <https://doi.org/10.1093/noajnl/vdaf173>
- Kiskowski, M. A., Hancock, J. F., & Kenworthy, A. K. (2009). On the use of ripley's K-function and its derivatives to analyze domain size. *Biophysical Journal*, 97(4), 1095–1103.

<https://doi.org/10.1016/j.bpj.2009.05.039>

- Lin, J.-R., Izar, B., & Sorger, P. K. (2018). Highly multiplexed immunofluorescence imaging of human tissues and tumors using t-CyCIF and conventional optical microscopes. *eLife*, 7, e31657. <https://doi.org/10.7554/eLife.31657.002>
- Liu, F., He, R., Sheeley, T., Scheiblin, D., Lockett, S. J., Ridnour, L. A., Wink, D. A., Jensen, M., Cortner, J., & Zaki, G. (2025). SPAC: a scalable, integrated enterprise platform for end-to-end single cell spatial analysis of multiplexed tissue imaging. *bioRxiv*. <https://doi.org/10.1101/2025.04.02.646782>
- Long, Y., Ang, K. S., Li, M., Chong, K. L. K., Sethi, R., Zhong, C., Xu, H., Ong, Z., Sachaphibulkij, K., Chen, A., Zeng, L., Fu, H., Wu, M., Lim, L. H. K., Liu, L., & Chen, J. (2023). Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST. *Nature Communications*, 14(1), 1325. <https://doi.org/10.1038/s41467-023-36796-3>
- Mah, C. K., Ahmed, N., Lopez, N. A., Lam, D. C., Pong, A., Monell, A., Kern, C., Han, Y., Prasad, G., Cesnik, A. J., Lundberg, E., Zhu, Q., Carter, H., & Yeo, G. W. (2024). Bento: a toolkit for subcellular analysis of spatial transcriptomics data. *Genome Biology*, 25(1), 82. <https://doi.org/10.1186/s13059-024-03217-7>
- Nirmal, A. J., & Sorger, P. K. (2024). SCIMAP: A Python toolkit for integrated spatial analysis of multiplexed imaging data. *Journal of Open Source Software*, 9(97), 6604. <https://doi.org/10.21105/joss.06604>
- Palantir Technologies. (2003). *Palantir Foundry documentation*. <https://palantir.com/docs/foundry/>
- Palla, G., Spitzer, H., Klein, M., Fischer, D., Schaar, A. C., Kuemmerle, L. B., Rybakov, S., Ibarra, I. L., Holmberg, O., Virshup, I., Lotfollahi, M., Richter, S., & Theis, F. J. (2022). Squidpy: a scalable framework for spatial omics analysis. *Nature Methods*, 19(2), 171–178. <https://doi.org/10.1038/s41592-021-01358-2>
- Plotly Technologies Inc. (2015). *Collaborative data science*. Plotly Technologies Inc. <https://plot.ly>
- Virshup, I., Rybakov, S., Theis, F. J., Angerer, P., & Wolf, F. A. (2024). anndata: Access and store annotated data matrices. *Journal of Open Source Software*, 9, 4371. <https://doi.org/10.21105/joss.04371>
- Waskom, M. L. (2021). Seaborn: Statistical data visualization. *Journal of Open Source Software*, 6(60), 3021. <https://doi.org/10.21105/joss.03021>
- Wolf, F. A., Angerer, P., & Theis, F. J. (2018). SCANPY: Large-scale single-cell gene expression data analysis. *Genome Biology*, 19. <https://doi.org/10.1186/s13059-017-1382-0>