iMSminer: A Data Processing and Machine Learning Package for Imaging Mass Spectrometry

Yu Tin Lin, Haohui Bao, Troy R. Scoggins IV, Boone M. Prentice*

Department of Chemistry, University of Florida, Gainesville, FL 32611

*Address correspondence to:
Dr. Boone M. Prentice
214 Leigh Hall
PO Box 117200
Department of Chemistry
University of Florida
Gainesville, FL 32611, USA
Phone: (352) 392-0556
Fax: (352) 392-4651
Email: booneprentice@chem.ufl.edu

Key words: imaging mass spectrometry, data processing, machine learning

ABSTRACT

Imaging mass spectrometry enables untargeted spatial profiling of compounds in animal tissues. Data preprocessing and mining are central to comprehensively unravel the complexity of hyperspectral imaging mass spectrometry experiments. Herein, we describe a user-friendly, partially GPU- or compiler-accelerated software pipeline that enables multi-ROI, multi-condition, and multi-replicate preprocessing and mining of larger-than-memory imaging mass spectrometry datasets in Python. The package, termed iMSminer, streamlines computational imaging mass spectrometry workflows, from spectral preprocessing to unsupervised exploratory analysis to univariate fold-change statistical analysis. These capabilities enable mining of ions for molecular co-localization, characteristic molecular profiles, and differential expression. Functions include raw imzML import, peak picking, baseline subtraction, mass alignment, peak integration, normalization, ROI selection, calibration, chemical database search, analyte filtering, image processing, box plot visualization, volcano plot and heatmap visualizations, dimensionality reduction, image clustering, and *in situ* segmentation. Furthermore, data processed by iMSminer can be easily interfaced to standard deep learning packages and other special-purpose modelling tasks in Python for more advanced use cases.

INTRODUCTION

Imaging mass spectrometry enables spatially-resolved, label-free measurements of animal tissue samples.[1, 2] Each imaging mass spectrometry experiment generates a dataset of molecular images hyperdimensional over the number of mass spectral peaks and x, y coordinate values. Computational preprocessing and data mining are thus crucial to unravel useful biochemical patterns and facilitate interpretation of global molecular interactions.[3, 4] Despite the capacity for hyperspectral data acquisition, the majority of published imaging mass spectrometry studies manually analyze a small targeted set of analytes of interest.[5] Several vendor-based software (*e.g.*, SCiLS Lab by Bruker Daltonics) and proprietary packages (*e.g.*, MSiReader[6]) provide access to a limited set of machine learning algorithms, though these typically require license purchases to access a larger set of machine learning tools. Open-source freeware is also available for computational workflows spanning data preprocessing and mining. For instance, software such as rMSI,[7] MSIQuant,[8] Ion-to-Image (i2i),[9] mzMine,[10] and BASIS[11] provides spectral and image processing capabilities, but often have limited downstream data analysis functionalities. Cardinal,[12-14] built in the programming language *R*,[15] is the most comprehensive open-source freeware that streamlines robust capabilities from data import to machine learning. However, some experience in programming and statistical modeling is advisable, introducing a barrier for widespread adoption by the imaging mass spectrometry community. Additionally, *R* traditionally limits accelerated computing to central processing unit (CPU)-based multi-processing, which is more inefficient than graphics processing (GPU)-based processing, and hinders access to standard machine learning packages (*e.g.*, Tensorflow[16] and PyTorch[17]).

With the rise of deep neural networks (DNNs) for computer vision tasks,[18] open-source imaging mass spectrometry databases (*e.g.*, Metespace[19]) have emerged to provide increasingly larger annotated training datasets. However, a user-friendly data processing package that can be easily interfaced to deep learning packages is lacking. *Python*[20] is increasingly recognized as a

versatile and hardware-accelerated programming language for machine learning in imaging mass spectrometry.[21-25] *Python* supports Nvidia CUDA-based[26] GPU acceleration with an established set of accelerated packages, such as RAPIDS,[27] Numba,[28] and standard deep learning packages. Herein, we have produced an open-source data processing and machine learning package for imaging mass spectrometry, termed iMSminer. iMSminer provides accelerated data processing and exploratory analysis functionalities to streamline data processing and mining of imaging mass spectrometry datasets. Furthermore, this package can be easily interfaced to standard deep learning packages by providing inputs to deep learning, which simplifies computational workflow for advanced use cases. A demonstration of interfacing iMSminer as inputs to a standard deep learning package is provided (https://github.com/Prentice-lab-UF/iMSminer). Moreover, iMSminer can be easily implemented on cloud computing platforms, (*e.g.*, Google Colaboratory;[29] see https://github.com/Prentice-lab-UF/iMSminer), for accessing powerful computational resources. As a modularized open-source package, iMSminer provides a useful baseline for experienced programmers to refactor for special-purpose modelling tasks. With partial GPU- and compiler-based acceleration, the package presented here provides high-performance data preprocessing and mining, though further acceleration is underway with more comprehensive parallelization and optimization in future releases.

METHODS

Case study 1: multi-ROI, multi-condition, and multi-replicate metabolite MALDI imaging of mouse heart and pancreas

Mouse heart and pancreas tissues for imaging mass spectrometry were removed from animal organs, frozen on dry ice, and then stored at −80°C until analysis. 12-μm mouse heart tissues were sectioned using a Leica CM 3050S Research Cryostat (Leica Biosystems, Wetzlar, Germany) using −25°C chamber temperature and −23°C object temperature, prior to thaw mounting onto indium tin oxide-coated microscope slides (Delta Technologies, Loveland, CO,

USA). A 10 mg/mL solution of 1,5-diaminonaphthalene (DAN) MALDI matrix layer was applied to the microscope slide using an HTX M5 TM Sprayer (HTX Technologies, LLC, Chapel Hill, NC, USA) with 0.1 mL/min flow rate over 4, 6, or 8 passes. MALDI imaging mass spectrometry was performed on a 7T solariX Fourier transform ion cyclotron resonance (FT-ICR) mass spectrometer equipped with a dynamically harmonized ParaCell XR (Bruker Daltonics, Bremen, Germany). Analysis was performed in negative ion mode from *m/z* 75 to 500 with a 1 megaword transient (~0.4 s ICR transient length). A 117 $\pm$ 75 *m/z* continuous accumulation of selected ions (CASI) window was used to perform gas-phase enrichment of low *m/z* metabolites. The MALDI source is equipped with a Smartbeam II Nd:YAG MALDI laser and was used to sample at a pixel spacing of 200 µm in the x and y dimensions using 200 laser shots per pixel (minimum laser focus, 2 kHz frequency) with Smart Walk enabled. Data preprocessing via iMSminer employed automatic peak picking with a *z*-score threshold of 3, a minimum number of data points of 6, and a limit of quantification of 10*noise. Image clustering was performed with 8 clusters and a perplexity of 5. *In situ* segmentation was performed with 6 clusters and a perplexity of 15.

Case study 2: high mass resolution lipid MALDI imaging of rat brain

Rat brain tissues for imaging mass spectrometry were removed from animal organs, frozen on dry ice, and then stored at −80°C until analysis. 10-µm rat brain tissues were sectioned using a Leica CM 3050S Research Cryostat (Leica Biosystems, Wetzlar, Germany), prior to thaw mounting onto indium tin oxide-coated microscope slides (Delta Technologies, Loveland, CO, USA). 1,5-diaminonaphthalene (DAN) MALDI matrix layer was sublimated to the microscope slide using an in-house sublimation apparatus. MALDI imaging mass spectrometry was performed on a 7T solariX Fourier transform ion cyclotron resonance (FT-ICR) mass spectrometer equipped with a dynamically harmonized ParaCell XR (Bruker Daltonics, Bremen, Germany). Analysis was performed in negative ion mode from *m/z* 400 to 2000 with ~0.5 s time-domain transient length, resulting in a resolution of ~35,000 *FWHM* at *m/z* ~760. The MALDI source is equipped with a

Smartbeam II Nd:YAG MALDI laser and was used to sample at a pixel spacing of 100 µm in the x and y dimensions using 200 laser shots per pixel (large laser focus, 2 kHz frequency) with Smart Walk enabled. Data preprocessing via iMSminer employed automatic peak picking with a *z*-score threshold of 3, a minimum number of data points of 6, and a limit of quantification of 10*noise. Image clustering was performed with 8 clusters and a perplexity of 5. *In situ* segmentation was performed with 10 clusters and a perplexity of 15.

Case study 3: high spatial resolution MALDI imaging of mouse urinary bladder

A detailed description of experimental methods employed was described under case study 1 in the Methods section by Vitek et al.[13] Data preprocessing via iMSminer employed even binning with 30,000 *FWHM* at *m/z* 400 scaled by a factor of 2, peak picking with a *z*-score threshold of 3, a minimum number of data points of 6, and a limit of quantification of 10*noise. Regression binning is advisable if a regression equation for mass resolving power vs. *m/z* is available. Image clustering was performed with 6 clusters and a perplexity of 3. *In situ* segmentation was performed with 7 clusters and a perplexity of 15.

RESULTS

We set out to develop a software that is easily adoptable to non-programmers with interactive input prompting for user inputs. This package enables user-friendly and streamlined data preprocessing and mining capabilities to empower comprehensive analysis of hyperspectral molecular images. iMSminer is broadly composed of two classes: Preprocess() (**Figure 1a**) and DataAnalysis() (**Figure 1b**). These functions are supported with CUDA-accelerated computing on GPU, just-in-time compiler, or vectorized Python packages. Preprocess() first imports imzML dataset by interactive input prompting for directory path. Preprocess() also performs chunking to enable processing of larger-than-memory imzML datasets on local machines. DataAnalysis() then

imports preprocessed intensity matrix and coordinate arrays by interactive input prompting for directory path. Region of interest (ROI) selection tools interactively receive user inputs. Structured data are analyzed with analyte-specific statistical tools or unsupervised *in situ* segmentation and image clustering.

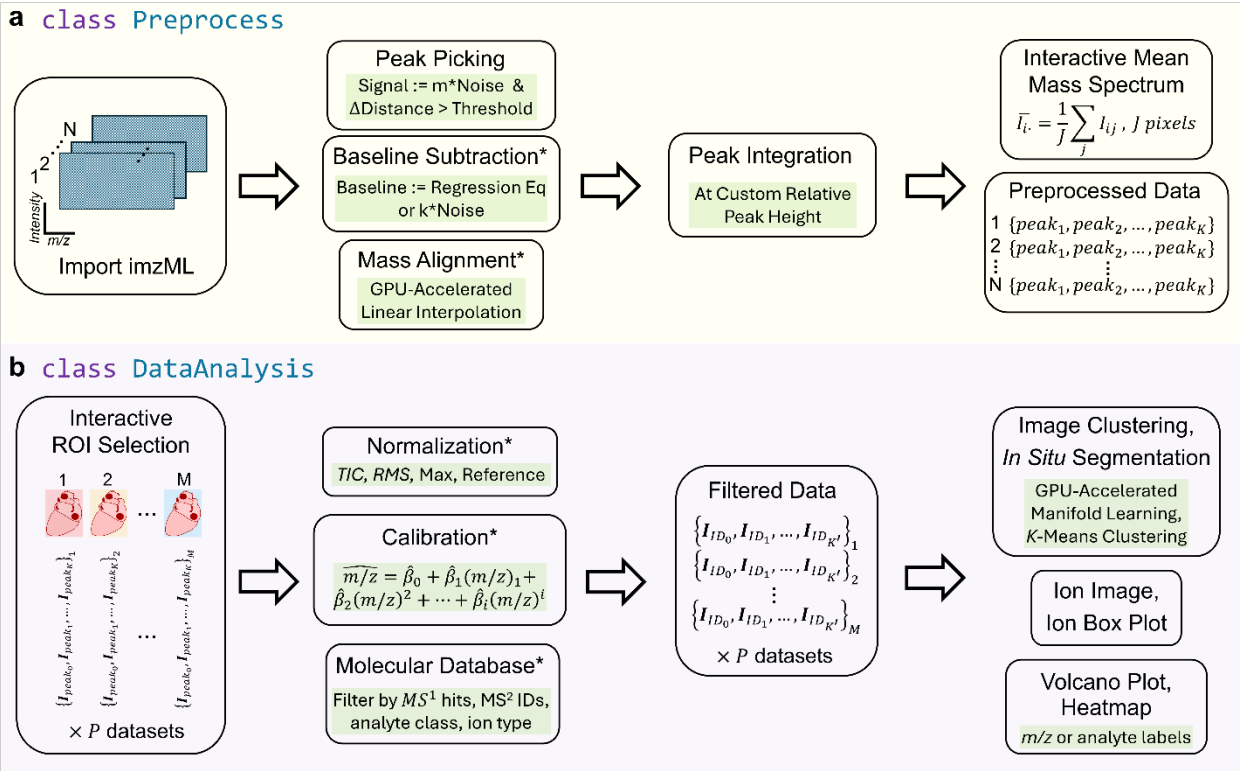

Figure 1. iMSminer is an open-source Python package composed of two classes: Preprocess() and DataAnalysis(). **a**, Preprocess() imports imzML datasets and performs peak picking, baseline subtraction, mass alignment, and peak integration with interactive input prompting. Raw imzML datasets are reduced to an intensity matrix and coordinate arrays. **b**, DataAnalysis() imports preprocessed data and supports interactive ROI selection, normalization, and calibration. Users may perform MS1 search and filter analytes of interest for downstream analysis. Unsupervised analysis, ion images, and ion statistics are generated based on input analytes. Users may visualize *m/z* or analyte labels corresponding to features in volcano plot or *t*-distributed stochastic neighbor embedding (*t*-SNE) of image clusters. * denotes optional functions.

The functionalities of iMSminer are illustrated using a multi-ROI (m=6), multi-condition (n=3), and multi-replicate (p=5) MALDI imaging mass spectrometry dataset of serially sectioned mouse heart and pancreas tissues (**Figure 2a**). Varying thicknesses of a MALDI matrix layer were

applied to serial tissue sections to simulate a multi-ROI and multi-condition sample cohort. iMSminer broadly enables untargeted univariate statistical analysis and global molecular pattern mining. For fold-change statistics, pairwise fold-change statistics across ROIs are summarized in volcano plots. Moreover, pairwise comparisons across ROIs are summarized in box plot visualizations, along with visualization of each corresponding ion images. A representative volcano plot shows differential molecular profiles between mouse hearts (MHs) and mouse pancreases (MPs) subjected to 6 passes of matrix spray (**Figure 2b**). This representative volcano plot revealed differences in metabolic expression between mouse heart and pancreas sections subjected to 6 matrix passes. A representative box plot analysis of *m/z* 133.015 shows pairwise ROI comparisons of mean ion intensity (**Figure 2c**). Box plots summarize mean ion statistics across pairs of ROIs, revealing differences between ROIs (*e.g.*, mouse heart sections subjected to 4 matrix passes versus 8 matrix passes). Ion images reveal *in situ* molecular distribution and relative quantification (**Figure 2d**). Global molecular pattern mining is enabled using *k*-means clustering and visualizes clustering analysis either in *t*-SNE or *in situ* ion heatmaps. This unsupervised analysis is performed to either group pixels by similar molecular profiles via *in situ* segmentation or to group analytes by similar *in situ* distributions via image clustering. *In situ* segmentation via *k*-means clustering is visualized in *t*-SNE (**Figure 2e**), along with the corresponding ROI labels visualized in *t*-SNE (**Figure 2f**). *In situ* segmentation groups pixels by similarity in global molecular profiles. Pixel clusters from *in situ* segmentation are then mapped to *in situ* distributions to link molecular profiles to tissue locations (**Figure 2g**). Additionally, similar molecular co-localizations are mined by unsupervised image clustering. *K*-means clusters of ion images and their corresponding ion images are visualized in *t*-SNE (**Figure 2h** and **Figure 2i**, respectively). Image clustering groups analytes by spatial co-localization and enhances interpretation of untargeted analysis. A representative mean image from cluster 4 is shown as an example of molecular co-localization (**Figure 2j**). The mean intensities from mean images

corresponding to *k*-means clusters and pairwise ROI comparisons were visualized in box plot. A representative box plot analysis is shown for cluster 4 (**Figure 2k**).

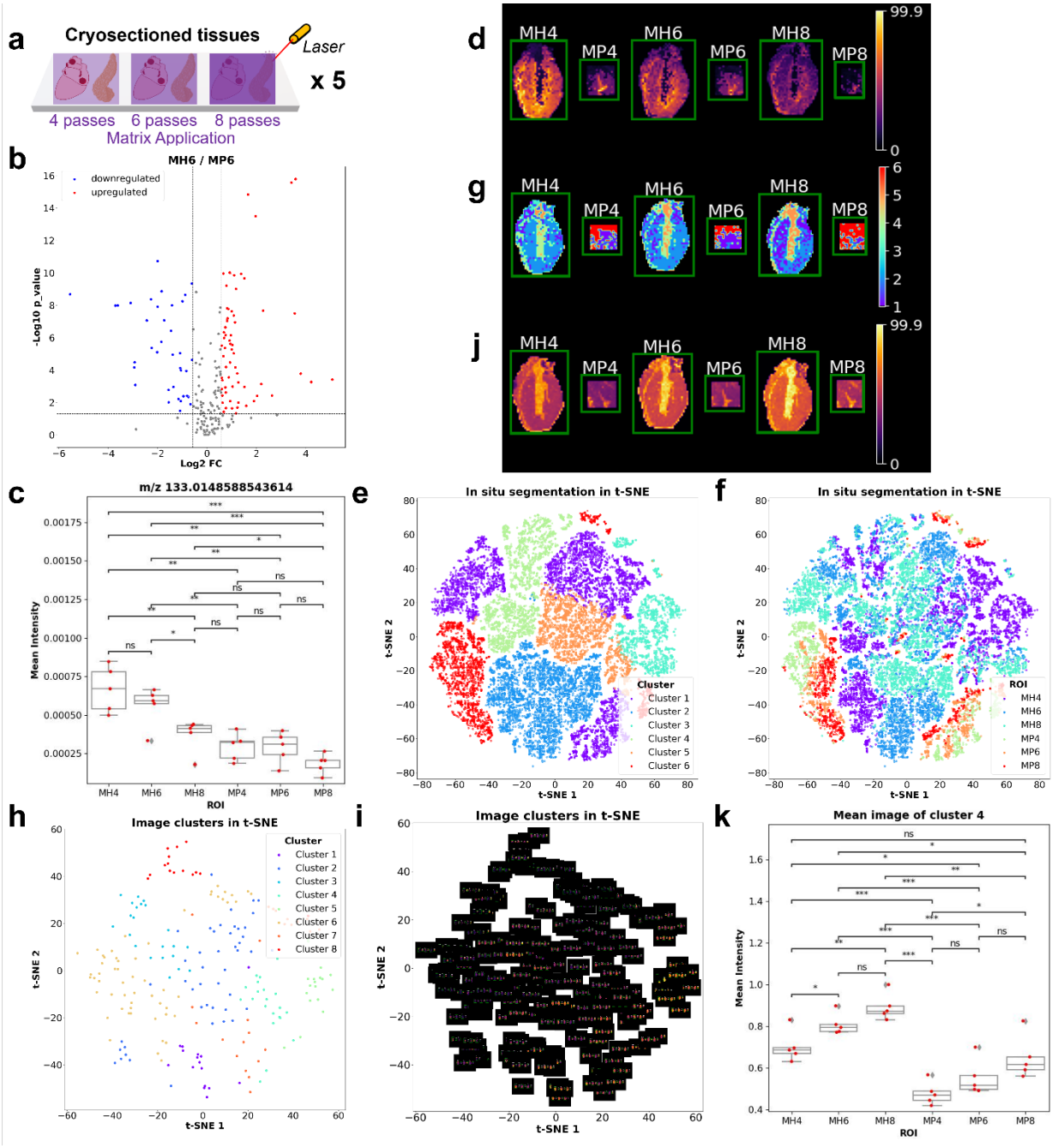

Figure 2. Data preprocessing, statistical analysis, and unsupervised exploratory analysis on multi-ROI (m=6), multi-condition (n=3), and multi-replicate (p=5) high-mass-resolution mouse heart and pancreas imaging. **a**, Five technical replicates of serially sectioned mouse heart and pancreas samples were subjected to three different amounts of 1,5-diaminonaphthalene matrix spray. **b**, Representative pairwise ROI fold-change analysis via volcano plot. **c**, Representative ion box plot of mean intensity vs. ROI with pairwise comparison summary statistics across

replicates. **d**, Representative ion image with ROI annotations. **e**, *K*-means clusters from *in situ* segmentation visualized in *t*-SNE. **f**, *t*-SNE of pixels annotated with ROI labels. **g**, *In situ* mapping of *k*-means clusters from *in situ* segmentation. **h**, *t*-SNE of *k*-means clusters of ion images. **i**, Ion images visualized in *t*-SNE. **j**, Representative *in situ* mapping of mean image from *k*-means clusters of images. **k**, Representative mean images of *k*-means cluster 4 across replicates summarized in box plot of mean intensity vs. ROI with pairwise comparison summary statistics.

Case study 2 analyzed a high mass resolution lipid MALDI imaging study of one hemisphere of a rat brain. Representative ion images of oppositely co-localized ions are shown for *m/z* 773.527 (**Supplementary Figure 1a**) and *m/z* 746.564 (**Supplementary Figure 1b**). *In situ* segmentation via k-means clustering separated pixels corresponding to brain regions displaying different co-localization patterns (**Supplementary Figure 1c**). This *in situ* segmentation analysis successfully separated brain regions in mouse brain section, including molecular layer, granular layer, white matter, cortex, corpus callosum, and hippocampus (**Supplementary Figure 1d**). For instance, cluster 8 captures the localization of *m/z* 773.527 to the molecular layer, while cluster 4 captures the localization of *m/z* 746.564 to the white matter. Image clustering via *k*-means clustering visualized in *t*-SNE (**Supplementary Figure 1e**), along with *in situ* heatmaps (**Supplementary Figure 1f**), demonstrates sensitivity to separation based on molecular co-localization. Herein, by grouping images by co-localization, hundreds of ion images can be compressed into 8 clusters to boost interpretation of tissue structures revealed by untargeted imaging mass spectrometry.

Case study 3 analyzed a high spatial resolution MALDI imaging experiment of mouse urinary bladder. Representative ion images of oppositely co-localized ions are shown for *m/z* 851.637 (**Supplementary Figure 2a**) and *m/z* 826.570 (**Supplementary Figure 2b**). For instance, cluster 2 captures the localization of *m/z* 826.570 to the urothelium, while cluster 4 captures the lack *of m/z* 826.570 in the lumen (**Supplementary Figure 2c**). This unsupervised *in situ* segmentation analysis successfully separated lamina propria, urothelium, and lumen of mouse urinary bladder section (**Supplementary Figure 2d**), which is congruent with the analysis results by Cardinal.[13] The segmentation results by iMSminer are more pixelated due to non-spatial

awareness of *k*-means clustering. However, fine-tuning weights using data-driven approaches such as deep learning may better capture spatial autocorrelation in ion images. Image clustering also demonstrated sensitivity to molecular co-localization (**Supplementary Figure 2e, 2f**), which enables grouping of ion images into similar co-localization patterns.

To assess the processing speed of iMSminer, execution times of preprocessing and unsupervised analysis were recorded on a laptop system equipped with a 13th Gen Intel® Core™ i9-13900HK CPU, a NVIDIA GeForce RTX 4080 Laptop GPU, and 32 GB of random-access memory (RAM). Timed functions include preprocessing one dataset without mass alignment using iMSminer, preprocessing one dataset without mass alignment using Cardinal v3.4.3, mass alignment of 100 pixels on GPU, mass alignment of 100 pixels on CPU, *in situ* segmentation on GPU, and *in situ* segmentation on CPU (**Supplementary Figure 3**). For case study 1 (*i.e.*, ~3000 pixels and ~1,800,000 *m/z* bins per dataset, 219 analytes), the execution times recorded in seconds of these functions were 82.7 ± 10.5, *NA*, 1.6 ± 0.1, 27.0 ± 0.2, 1.7 ± 0.3, and 32.2 ± 1.3, respectively. For case study 2 (*i.e.*, 21,510 pixels, 419,094 *m/z* bins, 207 analytes), the execution times recorded in seconds were 198.1 ± 3.8, *NA*, 1.3 ± 0.7, 20.5 ± 0.2, 1.3 ± 0.1, and 34.0 ± 1.1, respectively. For case study 3 (*i.e.*, 34,840 pixels, 90,000 *m/z* bins, 152 analytes), the execution times recorded in seconds were 20.4 ± 1.1, 316.3 ± 8.2, 1.5 ± 0.1, 40.3 ± 0.5, 2.2 ± 0.1, and 82.0 ± 7.4, respectively. For case studies 1 and 2, Cardinal failed to perform preprocessing without mass alignment due to R session crashing upon calling summarizeFeatures(). For case study 3, iMSminer enabled faster (~15.5 fold) preprocessing without mass alignment than Cardinal v3.4.3, which is expected given the relative speeds of the programming and processing foundations.

CONCLUSIONS

Computational processing and data mining are important workflows for comprehensive and interpretable unravelling of label-free imaging mass spectrometry datasets. Herein we have developed a software package, termed iMSminer, that enables user-friendly, accelerated

extraction of molecular co-localization, characteristic molecular profile assessment, and fold-change statistics. The utility of this package is illustrated using multi-ROI, multi-condition, and multi-replicate imaging mass spectrometry datasets. Furthermore, this package can be easily interfaced to standard deep learning packages in Python for more advanced use cases. The performance, functionalities, and user-friendliness of iMSminer make the package a strong candidate for application in routine and advanced imaging mass spectrometry data analytics.

ACKNOWLEDGEMENTS

The authors thank Matthias-Erich N. Born (MSV000095124) and MSV000086099 for providing datasets for reanalysis in case studies 2 and 3, respectively. This work was supported by the National Institutes of Health (NIH) under Award R01GM138660 and (National Institute of General Medical Sciences (NIGMS)). Y.T.L. was supported by a Barry Goldwater Scholarship, two UF College of Liberal Arts & Sciences (CLAS) Scholars awards, a Bristol-Myers Squibb Science Scholars award, and a Keaffaber Scholar award.

CODE AVAILABILITY

The iMSminer package is open-sourced on GitHub (https://github.com/Prentice-lab-UF/iMSminer) and available on PyPI (https://pypi.org/project/iMSminer/). Tutorials and documentation are provided on GitHub. Herein, python v3.11.7 was used with packages bokeh v3.3.4, cudf v24.04.01, cuml v24.04.00, cupy v13.1.0, iMSminer v1.0.0, matplotlib v3.8.0, msalign v0.2.0, numba v0.59.1, numpy v1.26.4, opencv v4.9.0, pandas v1.5.3, pyimzml v1.5.4, ray v2.21.0, scikit-learn v1.2.2, scipy v1.11.4, seaborn v0.11.2, statannotations v0.6.0, and statsmodeles v0.14.0. Mouse heart and pancreas datasets are available on MassIVE (MSV000095123).

REFERENCES

1. McDonnell, L. A.; Heeren, R. M. A., Imaging mass spectrometry. *Mass Spectrometry Reviews* **2007,** *26* (4), 606-643.
2. Norris, J. L.; Caprioli, R. M., Imaging mass spectrometry: A new tool for pathology in a molecular age. *PROTEOMICS – Clinical Applications* **2013,** *7* (11), 733-738.
3. Alexandrov, T., Spatial metabolomics and imaging mass spectrometry in the age of artificial intelligence. *Annual review of biomedical data science* **2020,** *3*, 61-87.
4. Verbeeck, N.; Caprioli, R. M.; Van de Plas, R., Unsupervised machine learning for exploratory data analysis in imaging mass spectrometry. *Mass spectrometry reviews* **2020,** *39* (3), 245-291.
5. Hu, H.; Laskin, J., Emerging Computational Methods in Mass Spectrometry Imaging. *Advanced Science* **2022,** *9* (34), 2203339.
6. Robichaud, G.; Garrard, K. P.; Barry, J. A.; Muddiman, D. C., MSiReader: An Open-Source Interface to View and Analyze High Resolving Power MS Imaging Files on Matlab Platform. *Journal of the American Society for Mass Spectrometry* **2013,** *24* (5), 718-721.
7. Ràfols, P.; Torres, S.; Ramírez, N.; Del Castillo, E.; Yanes, O.; Brezmes, J.; Correig, X., rMSI: an R package for MS imaging data handling and visualization. *Bioinformatics* **2017,** *33* (15), 2427-2428.
8. Källback, P.; Shariatgorji, M.; Nilsson, A.; Andrén, P. E., Novel mass spectrometry imaging software assisting labeled normalization and quantitation of drugs and neuropeptides directly in tissue sections. *Journal of proteomics* **2012,** *75* (16), 4941-4951.
9. Lillja, J.; Duncan, K. D.; Lanekoff, I., Ion-to-image, i2i, a mass spectrometry imaging data analysis platform for continuous ionization techniques. *Analytical Chemistry* **2023,** *95* (31), 11589-11595.
10. Schmid, R.; Heuckeroth, S.; Korf, A.; Smirnov, A.; Myers, O.; Dyrlund, T. S.; Bushuiev, R.; Murray, K. J.; Hoffmann, N.; Lu, M., Integrative analysis of multimodal mass spectrometry data in MZmine 3. *Nature biotechnology* **2023,** *41* (4), 447-449.
11. Veselkov, K.; Sleeman, J.; Claude, E.; Vissers, J. P.; Galea, D.; Mroz, A.; Laponogov, I.; Towers, M.; Tonge, R.; Mirnezami, R., BASIS: High-performance bioinformatics platform for processing of large-scale mass spectrometry imaging data in chemically augmented histology. *Scientific reports* **2018,** *8* (1), 4053.
12. Bemis, K. D.; Harry, A.; Eberlin, L. S.; Ferreira, C.; Van De Ven, S. M.; Mallick, P.; Stolowitz, M.; Vitek, O., Cardinal : an R package for statistical analysis of mass spectrometry-based imaging experiments. *Bioinformatics* **2015,** *31* (14), 2418-2420.
13. Bemis, K. A.; Föll, M. C.; Guo, D.; Lakkimsetty, S. S.; Vitek, O., Cardinal v.3: a versatile open-source software for mass spectrometry imaging analysis. *Nature Methods* **2023,** *20* (12), 1883-1886.
14. Luu, G. T.; Condren, A. R.; Kahl, L. J.; Dietrich, L. E. P.; Sanchez, L. M., Evaluation of Data Analysis Platforms and Compatibility with MALDI-TOF Imaging Mass Spectrometry Data Sets. *Journal of the American Society for Mass Spectrometry* **2020,** *31* (6), 1313-1320.
15. Team, R. C., R: A Language and Environment for Statistical Computing. **2021**.
16. Martín, A.; Ashish, A.; Paul, B.; Eugene, B.; Zhifeng, C.; Craig, C.; Greg, S. C.; Andy, D.; Jeffrey, D.; Matthieu, D.; Sanjay, G.; Ian, G.; Andrew, H.; Geoffrey, I.; Michael, I.;

Jia, Y.; Rafal, J.; Lukasz, K.; Manjunath, K.; Josh, L.; Dan, M.; Rajat, M.; Sherry, M.; Derek, M.; Chris, O.; Mike, S.; Jonathon, S.; Benoit, S.; Ilya, S.; Kunal, T.; Paul, T.; Vincent, V.; Vijay, V.; Fernanda, V.; Oriol, V.; Pete, W.; Martin, W.; Martin, W.; Yuan, Y.; Xiaoqiang, Z., TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. **2015**.
17. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Köpf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; Chintala, S., PyTorch: An Imperative Style, High-Performance Deep Learning Library. *arXiv [cs.LG]* **2019**.
18. Krizhevsky, A.; Sutskever, I.; Hinton, G. E., ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, Pereira, F.; Burges, C. J.; Bottou, L.; Weinberger, K. Q., Eds. Curran Associates, Inc.: 2012; Vol. 25.
19. Palmer, A.; Phapale, P.; Chernyavsky, I.; Lavigne, R.; Fay, D.; Tarasov, A.; Kovalev, V.; Fuchser, J.; Nikolenko, S.; Pineau, C.; Becker, M.; Alexandrov, T., FDR-controlled metabolite annotation for high-resolution imaging mass spectrometry. *Nature Methods* **2017,** *14* (1), 57-60.
20. Van Rossum, G.; Drake, F. L., Jr., *Python tutorial*. Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands: 1995.
21. Behrmann, J.; Etmann, C.; Boskamp, T.; Casadonte, R.; Kriegsmann, J.; Maaβ, P., Deep learning for tumor classification in imaging mass spectrometry. *Bioinformatics* **2018,** *34* (7), 1215-1223.
22. Hu, H.; Bindu, J. P.; Laskin, J., Self-supervised clustering of mass spectrometry imaging data using contrastive learning. *Chemical science* **2022,** *13* (1), 90-98.
23. Abdelmoula, W. M.; Lopez, B. G.-C.; Randall, E. C.; Kapur, T.; Sarkaria, J. N.; White, F. M.; Agar, J. N.; Wells, W. M.; Agar, N. Y., Peak learning of mass spectrometry imaging data using artificial neural networks. *Nature communications* **2021,** *12* (1), 5544.
24. Xie, Y. R.; Castro, D. C.; Rubakhin, S. S.; Trinklein, T. J.; Sweedler, J. V.; Lam, F., Multiscale biochemical mapping of the brain through deep-learning-enhanced high-throughput mass spectrometry. *Nature Methods* **2024**, 1-10.
25. Hernly, E.; Hu, H.; Laskin, J., MSIGen: An Open-Source Python Package for Processing and Visualizing Mass Spectrometry Imaging Data. **2024**.
26. Nickolls, J.; Buck, I.; Garland, M.; Skadron, K., Scalable Parallel Programming with CUDA: Is CUDA the parallel programming model that application developers have been waiting for? *Queue* **2008,** *6* (2), 40-53.
27. Team, R. D., RAPIDS: Libraries for End to End GPU Data Science. **2023**.
28. Lam, S. K.; Pitrou, A.; Seibert, S., Numba: a LLVM-based Python JIT compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, Association for Computing Machinery: 2015.
29. Bisong, E., Google Colaboratory. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*, Apress: Berkeley, CA, 2019; pp 59-64.