

Marquette University

e-Publications@Marquette

Master's Theses (2009 -)

Dissertations, Theses, and Professional
Projects

Deep Learning Approach for Dynamic Sampling for High-Throughput Nano-DESI MSI

David Helminiak
Marquette University

Follow this and additional works at: https://epublications.marquette.edu/theses_open



Part of the [Engineering Commons](#)

Recommended Citation

Helminiak, David, "Deep Learning Approach for Dynamic Sampling for High-Throughput Nano-DESI MSI" (2021). *Master's Theses (2009 -)*. 710.

https://epublications.marquette.edu/theses_open/710

DEEP LEARNING APPROACH FOR DYNAMIC SAMPLING
FOR HIGH-THROUGHPUT NANO-DESI MSI

by

David Steven Helminiak

A Thesis submitted to the
Faculty of the Graduate School, Marquette University,
in Partial Fulfillment of the Requirements for
the Degree of Master of Science

Milwaukee, Wisconsin

December 2021

ABSTRACT
DEEP LEARNING APPROACH FOR DYNAMIC SAMPLING
FOR HIGH-THROUGHPUT NANO-DESI MSI

David Steven Helminiak

Marquette University 2021

Mass Spectrometry Imaging (MSI) extracts molecular mass data to form visualizations of molecular spatial distributions. The involved scanning procedure is conducted by moving a probe across and around a rectilinear grid, as in the case of nanoscale Desorption Electro-Spray Ionization (nano-DESI) MSI, where singular measurements can take up to ~5 seconds to acquire high-resolution (better than 10 μm) results. This temporal expense creates a high inefficiency in sample processing and throughput. For example, in a high-resolution nano-DESI study, a single mouse uterine tissue section (2.5 mm by 1.7 mm) had an acquisition time of ~4 hours to acquire 104,400 pixels. Anywhere from ~25-30% of those pixels were outside the actual tissue, and a further portion of those locations lacked relevant information.

An existing method, a Supervised Learning Approach for Dynamic Sampling (SLADS), utilizes information obtained during an active scan to infer, using a least-squares regression, regions of interest that most likely contain meaningful information, and a computationally inexpensive weighted mean interpolation to perform sparse sample reconstruction. This approach could potentially be used to significantly improve throughput in this and other biological tissue scanning applications. However, existing SLADS implementations were neither designed nor optimized for leveraging or handling the 3rd dimension in MSI of molecular spectra. Further, integrating more recent advances in machine learning since the last SLADS publication issuance, such as Convolutional Neural Network (CNN) architectures, offers additional performance gains. The objective of this research is the updating, re-design, and optimization of the SLADS methodology, to form a Deep Learning Approach for Dynamic Sampling (DLADS) for high-resolution biological tissues and integration with nano-DESI MSI instrumentation.

ACKNOWLEDGEMENTS

David Steven Helminiak

My deepest thanks go out to my family and friends who have never ceased in their encouragement and support. This work and my continued academic pursuits would not have been possible without the drive born of their words and actions. A special thanks goes to Fr. Thomas Manahan S.J. for his spiritual and mental guidance, candid and impartial council, not to mention his unceasing prayers.

God is Good... All the Time...
All the Time... God is Good!

I also extend my thanks to my advisor, Dr. Dong Hye Ye, as this has been and continues to be an invaluable research experience. Thank you for providing me this opportunity, as well as offering your continued insights into this research and the machine learning field.

It is with gratitude that I note my fellow lab member, Najib Aboobacker, as well as current and former Marquette Faculty members: Dr. Chris Ababei, Dr. Jonathan Fleischmann, Dr. Majeed Hayat, Dr. Hyunjae Park, Dr. James Richie, Dr. Susan Schneider, Dr. Chandana Tamma, Ms. Katie Tarara, and Dr. Edwin Yaz. Given your support, enthusiasm, and proactive generation of numerous educational/research opportunities, you are the true difference at Marquette. I express a similar appreciation to Dr. Julia Laskin and Hang Hu from Purdue University for lending their knowledge, support, and resources over the course of this work's development.

Additional thanks go to my committee: Dr. Richard Povinelli, Dr. Henry Medeiros, and Dr. Fred Frigo, for their mentorship, knowledge, wisdom, and above all, patience.

Above all else, Thanks Be To God! "Behold, you have instructed many, and you have strengthened the weak hands. Your words have upheld him who was stumbling, and you have made firm the feeble knees." – Job 4:3-4

-David Steven Helminiak

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	ii
LIST OF TABLES	v
LIST OF FIGURES	vi
1. INTRODUCTION	1
1.1 Motivation.....	1
1.2 History and Review.....	4
1.2.1 DESI MSI.....	4
1.2.2 Deep Learning.....	5
1.2.3 U-Net Architecture.....	5
1.2.4 Sparse Sampling.....	6
1.3 Objectives	9
1.4 Development Steps	11
2. DATA	13
2.1 Hardware and Software.....	13
2.2 Datasets	13
3. DEVELOPMENT	15
3.1 Data Pre-Processing	15
3.1.1 m/z Visualization	15
3.1.2 Alignment and Rescaling.....	16
3.1.3 m/z Selection.....	18
3.2 Methods.....	24
3.2.1 Sampling Overview	24

3.2.2 Pointwise Acquisition	25
3.2.3 Linewise Acquisition	27
3.2.4 c Value Regularization.....	28
3.3 Updated SLADS and DLADS	30
3.3.1 Code Design and Notes.....	30
3.3.2 DLADS Architecture	32
3.3.3 Reduction in Distortion.....	34
3.4 c Value Optimization	36
3.5 Physical Integration	38
4. RESULTS	40
4.1 Metrics	40
4.2 Ablation Studies.....	40
4.2.1 DLADS Inputs	41
4.2.2 DLADS Model Parameters	42
4.2.3 DLADS Augmentation	44
4.2.4 SLADS.....	45
4.3 Simulations	47
5. CONCLUSIONS.....	53
5.1 Discussion.....	53
5.2 Future Work	55
5.3 Concluding Remarks.....	57
BIBLIOGRAPHY	58

LIST OF TABLES

Table 1	Uterine sample dimensions and information.....	14
Table 2	Determined monoisotopic m/z for uterine samples	20
Table 3	Results for optimizing c value	36
Table 4	Results for varying inputs in DLADS.....	41
Table 5	Results for varying DLADS hyperparameters	43
Table 6	Results for augmenting input for DLADS	45
Table 7	Results for varying input, window, and RD generation in SLADS	46
Table 8	Results for pointwise simulation of testing set	48
Table 9	Results for linewise simulation of testing set.....	48
Table 10	Progressive results for simulation of a testing sample.....	52

LIST OF FIGURES

Figure 1	Raster scanning pattern and Otsu thresholding applied to sample	1
Figure 2	TIC of uterine samples acquired through nano-DESI MSI	14
Figure 3	Example nano-DESI MSI spectrum with a m/z visualization	15
Figure 4	Misaligned and realigned visualizations of a m/z	17
Figure 5	Asymmetric average of m/z images with measurement mask	18
Figure 6	Symmetric average of m/z images with measurement mask	18
Figure 7	Visualization of determined monoisotopic m/z	20
Figure 8	Binary foreground masks for uterine samples	21
Figure 9	Score distribution of m/z values for a uterine sample	22
Figure 10	Distribution of chosen m/z values for the uterine samples	23
Figure 11	m/z images characteristic of underlying tissue geometry	23
Figure 12	Procedural SLADS and DLADS sampling framework	24
Figure 13	Model training procedure	25
Figure 14	Modified U-Net architecture	33
Figure 15	Visualizations from data augmentation network	34
Figure 16	Overview of the original and summation RD	35
Figure 17	AUC for varying c , RD, and windows	37
Figure 18	PSNR progression with static window and original RD	37
Figure 19	PSNR with dynamic window and original RD	38
Figure 20	PSNR with dynamic window and original RD	38
Figure 21	Physical integration framework	39
Figure 22	Progression of PSNR for m/z reconstructions for varying inputs	42

Figure 23	Progression of PSNR for m/z reconstructions using MAE	43
Figure 24	Progression of PSNR for m/z reconstructions using MSE	44
Figure 25	Progression of PSNR for m/z reconstructions with augmentation	45
Figure 26	Progressive PSNR of m/z reconstructions for SLADS-LS	46
Figure 27	Progressive PSNR of m/z reconstructions for SLADS-Net	47
Figure 28	Ground-truth averaged m/z images for testing samples	48
Figure 29	Progressive PSNR of m/z reconstructions for pointwise testing	49
Figure 30	Progressive PSNR of m/z reconstructions for linewise testing	49
Figure 31	Progressive PSNR between ERD and RD for pointwise testing	50
Figure 32	Progressive PSNR between ERD and RD for linewise testing	50
Figure 33	Measurement masks and reconstructions for first test sample	51
Figure 34	Measurement masks and reconstructions for second test sample	51

CHAPER 1 INTRODUCTION

1.1 Motivation

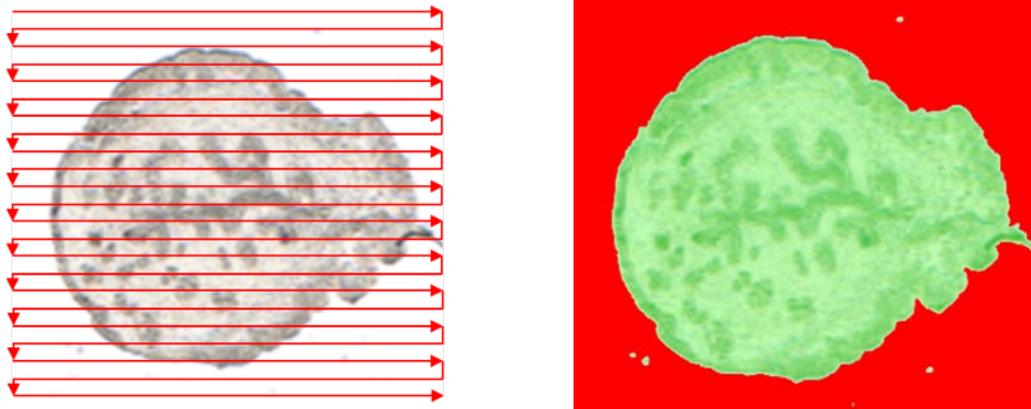


Figure 1: Traditional rectilinear scanning pattern (left) overlaid on an optical view of a mouse uterine section, and Otsu thresholding applied to the FOV (right), showing 53% of sampling locations occupy the foreground (green) and 47% in the background (red)

Traditional spectroscopy and microscopy scanning technologies typically require around ~5-10 seconds per pixel for high-fidelity acquisitions. Rectilinear (Fig. 1) scanning remains the preferred industry standard, as it ensures that all of the desired data is obtained. However, high-resolution samples (with precisions of $<10 \mu\text{m}$) can take hours to scan. There exists the potential to spend a majority of scanning time on acquiring non-relevant information. Such data could be as simple as the locations around the intended sample, but within the acquisition equipment's Field of View (FOV), or more broadly, any information not relevant to an experimentalist's scanning objective. Simply preventing scanning in the background, as shown in Figure 1, can double throughput without loss of any meaningful information. Specifically, within a biological context, if a researcher is solely interested in a tissue's internal distribution of epithelium, then it

would be unnecessary to also scan locations unlikely to contain it. Overall, there results a high inefficiency for throughput, particularly for Mass Spectrometry Imaging (MSI).

A general approach that has seen growing adoption has been performing limited random sparse sampling, then relying on deep learning to perform reconstruction of a low-error estimation of the ground-truth. However, the analysis of these models' reasoning and determinations remains problematic to reverse-engineer. Improved levels of understanding are necessary for highly regulated or precision applications, such as in material and biological research, to increase general adoption of deep learning models in areas where it would prove most beneficial.

Alternatively, there exist dynamic sampling algorithms, which optimize subsequent measurement locations through statistical models based on previous measurements obtained during a live-scan. This does not preclude the use of deep learning as a potential for enhanced post-processing. Rather, it ensures that the only locations obtained are probabilistically expected to contain desired information. An approach, based on this methodology, was previously created in the form of the Supervised Learning Approach for Dynamic Sampling (SLADS) algorithm, that addresses the aforementioned issues through the incorporation of stochastic processes into a compressed sensing method. Statistical features, extracted from a reconstruction of the sample (generated and updated throughout the acquisition process based on data obtained during the scan), are used to regularly estimate the potential entropy reduction for currently unmeasured locations. This dynamic determination of optimized scanning locations maximizes scanning throughput, minimizes the computational expense of deep

learning reconstruction (produced instead through Inverse Distance Weighted (IDW) mean interpolation), and removes the need for detailed analysis for result explanation.

SLADS has primarily seen application and development for CT, as well as Raman and Electron Back Scatter Diffraction (EBSD) microscopy. However, it has potential to improve throughput for similar technologies, such as nanoscale Desorption Electro-Spray Ionization (nano-DESI) MSI, where the high resolution scan of a mouse uterine section (Figure 1), with a size of only 2.4x1.7 mm (with 104,400 individual scanning locations), required ~4 hours to acquire [1]. Given its high spatial resolution, relative low-cost, and ability to be performed at regular intervals on a single sample, MSI has largely become popular in clinical and research settings for examining the spatial distributions (localization) of biomolecules and their progressive interactions. This makes MSI in general, a prime target for integration with dynamic sampling algorithms for improved throughput, as offered by the SLADS methodology.

This thesis' development was further motivated by its funding, as provided by the National Institutes of Health (NIH) Common fund, through the Office of the Strategic Coordination/Office of the NIH Director under award UG3HL145593. This award was provided under the Human BioMolecular Atlas Program (HUBMAP) consortium [2] for Transformative Technology Development (TTD). HUBMAP's overall goal is to map human tissues at single cell resolution, a subset of which focuses on providing proof-of-principle for developing technologies. While the hardware and procedural aspects of nano-DESI MSI continues to be developed by Purdue University, there is an opportunity to improve its throughput beyond other existing MSI methods, through an integration of a dynamic sampling algorithm.

1.2 History and Review

1.2.1 DESI MSI

DESI MSI deposits a charged solvent and then extracts it from a sample surface (typically biological) for the ionization and desorption of hundreds of molecules simultaneously, up to ~2000 Da (Daltons). Extracting information from the acquired spectra allows for a 2D (two-dimensional) visualization of specific molecular distributions. DESI is historically challenged with regards to its spatial resolution, as compared with other MSI methods including MALDI (Matrix-Assisted Laser Desorption Ionization) and SIMS (Secondary Ion Mass Spectroscopy), which can operate in the 1-2 μm range. SIMS has the disadvantage of being a destructive method and only operating up to ~1,000 Da, while MALDI typically requires a high level of tissue preparation to prevent sample degradation, as well as operations being performed within a vacuum [3].

Recent advances in DESI produced from Purdue University, labeled as nano-DESI MSI [1], are rapidly allowing for spatial fidelity levels (the minimum distance between identifiably distinct features) similar to SIMS and MALDI, with acquisitions demonstrated at better than 10 μm . DESI does not require complicated pre-treatment and imparts minimal damage to the sample, which allows for additional complementary analyses, as might be possible with subsequent MALDI scans. This particular implementation currently relies on XCalibur software, produced by Thermo Fisher Scientific Inc. (Waltham, MA), to control the sampling probe positioning, speeds, and the actual acquisition of location spectra. Sample locations are scanned line-by-line through either custom or commercial hardware platforms, with the resulting information stored in a proprietary “RAW” format.

1.2.2 Deep Learning

Deep learning exists as a subset of Machine Learning (ML) and is derived from the broader field of Artificial Intelligence (AI). AI concerns the creation of intelligent entities developed by understanding methods through which humans interact with the world. ML focuses more strongly on aspects of thinking and acting rationally, basing decisions on experience to maximize a measure of performance [4]. The procedures for producing these decisions for a specific problem are formalized in algorithms, that is computational procedures for processing inputs [5]. Success with ML largely depends on the ability of an employed model to capture the complexity of the involved concepts, as well as the quality and quantity of data to be analyzed (or rather experienced). More specific to this work is the process of supervised learning, where data used for training a model has known labels and evaluation of relative performance is conducted over separate datasets [6]. This allows for an evaluation of a model's ability to generalize, or perform well when presented with previously unseen inputs. Deep learning uses models created from artificial neural networks with multiple layers (the network depth) to progressively represent high-level, abstract features. Generally speaking, deeper and broader networks can represent increasingly complex data and transformations. The resultant model representation can be leveraged to encode (learn) features from an input and map, or decode the information to a desired, trained output.

1.2.3 U-Net Architecture

A particularly successful model for processing 2-dimensional (2D) information (such as images) is the Convolutional Neural Network (CNN). CNN models use convolutional layers, as inspired by biological visual cortices, to capture information in a

receptive field, forming representative features, or activation maps. CNN models are commonly used for classification, encoding increasingly abstract representations by decreasing spatial dimensions through pooling operations and proportionately increasing the number of representative convolutional filters.

The U-Net architecture is a CNN variant first published in 2015 for the segmentation of biomedical images [7]. U-Net comprises symmetric encoding and decoding halves, giving the architecture a distinctive U-shape when visualized. The encoding section compresses spatial dimensions and increases depth, as in a typical CNN. The decoding section then upscales its inputs, progressively halving the number of convolutional filters and re-combining with the encoded feature maps through skip connections. When used for classification, the final layer(s) use a 1x1 convolutional layer to map channels to a given number of desired output classes.

U-Net allows for the architecture's output to have the same spatial dimensions as the input. Further, since only convolutional operations are performed (in contrast to Multi-Layer Perceptron (MLP) or "Dense" neural networks), the U-Net can be constructed to handle arbitrarily sized inputs and outputs. This makes U-Nets ideal candidates for image-to-image translation tasks such as denoising, coloration, and depth estimation, where the inputs are generally geometrically representative of the outputs.

1.2.4 Sparse Sampling

In existing scholarly literature, there exist a variety of sparse sampling methods that may be broadly separated into static and dynamic categories. The static approaches perform scans with entirely pre-determined locations, these can be generated uniformly, randomly, or through stochastic models [8, 9, 10]. There also exist static patterns where

the sample geometry is sufficiently understood that a manually created mask can be consistently relied upon to only capture desired information [11]. Dynamic methods [12, 13] employ information obtained during the scanning process to inform future locations and generally are based in compressed sensing techniques and machine learning.

One of the more successful dynamic methods is SLADS, which selects scan locations based on those which maximize a Reduction in Distortion (RD) measured between a ground-truth and reconstruction. The formalized SLADS framework was introduced in 2016 by Godaliyadda et al. [14], based on the authors' earlier work in 2014 [15], comprising the Model-Based Dynamic image Sampling (MBDS) algorithm. MBDS was based on a Monte Carlo method, but was demonstrated to be a slower and less precise predecessor to SLADS.

SLADS was then employed in 2017 by Scarborough et al. [16] for dynamic X-ray crystalline protein acquisition, seeking to lower the amount of damage imposed onto a sample by exposure to high energy X-rays. Only 9.0% of a sample was required to be obtained synthetically and ~5% experimentally for a Normalized Distortion (ND) level of $\sim 10^{-3}\%$; effectively, this means a 20-fold reduction in applied radiation dosage. In addition, new location selections within 1-10 ms were achieved, meaning that the computational times during implementation became negligible. The technique was applied again in 2018 by Zhang et al. [17] in the context of confocal Raman microscopy, yielding a 6-fold improvement in the number of pixel measurements needed to result in a $\sim 0.1\%$ image difference, relative to full raster scans.

SLADS was also used in 2018 [18] to provide a significant reduction in the number of measurements needed for Electron Back Scatter Diffraction (EBSD)

microscopy. The implementation roughly reduced traditional acquisition times on the order of ~60-95%, depending on acceptable reconstruction error. SLADS was able to achieve below 10^{-5} ND levels with only 6.94% of scanned locations in a synthetic image. Experimentally, it achieved ND levels less than 10^{-4} with 13% of a sampled image. Its produced reconstructions were additionally shown as superior to random and low-discrepancy sampling, which for the synthetic image (sampled up to 20%) attained higher ND levels by 3 orders of magnitude.

SLADS can be modified through training to be more effective according to specific application contexts. A 2018 study by Zhang et. al. [19] used SLADS to reduce the total required sampling locations for Energy Dispersive X-ray Spectroscopy (EDS) up to 90% without noticeable degradation of resulting scan fidelity. Another study also by Zhang et al. in 2018 [20] performed the first multi-model study for the approach. Training methods, including least-square and support vector regression, as well as a Multi-Layer Perceptron (MLP) network named SLADS-Net were examined. SLADS-Net was shown to provide improved performance when training images dissimilar to those within a testing set, offering increased ability to generalize. When training and testing images are similar, non-linear learning methods were observed to have equivalent performance to the least-squares implementation. Zhang et. al additionally published in 2018: U-SLADS [21], employing Hierarchical Gaussian Mixture Models (HGMMs) to produce feature sets, allowing improvements to dynamic dendrite sampling procedures.

A slightly contrasting approach, Probabilistic Approach to Dynamic Image Sampling (PADIS), was published in 2020 by Grosche et al. [22]. PADIS was designed for single-channel SEM (Scanning Electron Microscope) images, relying on a probability

mass function. Therein exists specific mention of overcoming the tendency of SLADS and SLADS-Net to under-sample regions with apparent homogeneity and issues with non-linear scaling runtimes, with respect to image sizes and sampling densities. When considering samples with fine details and textures, PADIS was shown to outperform SLADS and SLADS-Net, where they were observed to overfocus on acquisition of locations along the edges of sample structures.

Although the limited number of publications on SLADS have shown notable improvements over traditional and even more complex, modern machine learning approaches, it has not yet seen either widespread adoption or extensive research. The initial released SLADS code relied on a simple Least-Squares (SLADS-LS) regression. The only major update produced on this methodology, SLADS-Net, used a simple MLP network. Therein, the model's performance was nearly identical to that of the original SLADS, with only a marginal improvement to generalization capability. Both SLADS-LS and SLADS-Net used only consistently sized, often synthetically generated images, with very similar information content. Perhaps the most significant limitation to be considered is the problem of dimensionality. While SLADS has been shown to reduce needed measurements in 2D scenarios, a third-dimension causes a significant increase in computational overhead, data sparsity, and the risk of obfuscating data with dimensional reduction strategies [23].

1.3 Objectives

This research aims to update the SLADS methodology for experimental integration with nano-DESI MSI to maximize throughput (building on preliminary work [24]). This adaption of a dynamic sampling algorithm intends to leverage the multiple

mass-per-charge (m/z), or molecular spectral distributions to determine locations where scanning should be performed. The algorithm should have the ability to be calibrated with respect to equipment's spatial resolution, equipment FOV, and function for multiple tissue types. While a method for direct integration with physical hardware should be created, evaluation of the resultant algorithms should be performed with simulated scanning of available nano-DESI MSI scans, realizing low-error estimations of the ground-truth visualizations of multiple m/z . The SLADS methodology should be further advanced through implementation of a more advanced deep learning neural network architecture, in a Deep Learning Approach for Dynamic Sampling, or DLADS. Ideally for ease of use, hyperparameters (as many as possible) should be removed, optimized in advance, or automatically selected.

Nano-DESI data is unique in that individual locations being scanned are not necessarily consistent in horizontal and vertical dimensionality. Further, the acquired lines are not necessarily consistent in terms of start/end position, sampling frequency, or specifically measured mass/intensity pairs. The design and function for updated implementation(s) of SLADS must be able to dynamically account for these variations.

It must also be considered that SLADS was designed to choose measurement locations point-by-point in a predefined 2D-coordinate system for a single channel of information. As mentioned in Section 1.2.1, the XCalibur software employed in nano-DESI MSI, acquires data line-by-line and cannot be straightforwardly modified for point-by-point scanning. While the vendor could modify this behavior in future, to ensure that a physical proof-of-concept for integration is presently achievable, line-by-line or linewise operational modes should be developed and evaluated against pointwise scanning.

1.4 Development Steps

The methods used during over the course of development firstly focused on the integration of DESI MSI data and the SLADS methodology, both in simulation and for potential real-world implementation. Secondly, simulation was performed and compared with both previously published models and a more advanced machine learning architecture. More specific milestones for the project are described as follows:

- 1) A previously published SLADS implementation was first deconstructed and rebuilt into DLADS, which employed a CNN with deep learning.
- 2) Both SLADS and DLADS models were adjusted to accommodate and integrate nano-DESI MSI data, treating each molecular spatial distribution as an independent sample. At this stage, the data could be heavily pre-processed (normalized by collaborators, imported as image files, resized as needed, etc.), in advance of employing the models, in order to judge the merits of further development and the direction of research efforts.
- 3) SLADS and DLADS models were modified to incorporate average multiple m/z 's spatial distribution information into a single representative image. The data could still be heavily pre-processed in advance of training/testing the models, but the m/z images could no longer be treated independently.
- 4) Conjunctive efforts were conducted with experimentalist collaborators, in the formation of an approach for direct integration of SLADS/DLADS models, with physical nano-DESI MSI instrumentation.
- 5) A new method was employed for the determination of sampling locations, given the MSI equipment's probe movement constraints (e.g. a group-based acquisition method limited to single lines/rows).

- 6) The DLADS model was optimized, through integration of multiprocessing libraries, vectorization of array calculations, removal/automation of hyperparameters, and light model ablation studies.
- 7) MSI files should be read directly with SLADS and DLADS, with all molecular visualizations handled internally to their coded implementations. At this stage, experimentalists became able to use configuration files to specify how the data should be processed during the scan (normalization methods, specific ion mass/charge ranges/values, etc.).
- 8) An alternate method for producing ground-truth RD images for training was created, taking into account the multiple m/z spatial distributions obtained during scans.
- 9) A final model ablation study should be performed, showing direct comparison with prior published SLADS and SLADS-Net models.

Chapter 2 DATA

2.1 Hardware and Software

SLADS and DLADS development was conducted on a dual-socket Intel Xeon-2650v2 2.5 GHz platform (overclocked to 3.0 GHz) with 16 cores (32 threads through hyperthreading) assigned to parallelizable tasks, 128 GB DDR3, and an NVIDIA GTX 1080TI with 11 GB GDDR5X. Implementation of the constructed models for physical integration was planned, performed and verified to function (in simulation) with an Intel i5-8500 3.0 GHz system with 8 GB DDR4 and no discrete GPU. Operation on this latter computer was only possible when configured to disable multiprocessing and utilize pre-trained models. An up-to-date list of third-party packages/libraries, along with the code utilized for this project are available through an online Github repository at the time of publication, under the version 0.8.6: <https://github.com/Yatagarasu50469/SLADS>.

2.2 Datasets

The primary datasets employed for model development were provided by the Purdue University Department of Chemistry and acquired with nano-DESI MSI on a Thermo Fisher Scientific Q-Exactive HF-X Orbitrap mass spectrometer. These datasets include 10 mouse uterine samples, randomly divided using an 60/20/20% training/validation/testing split. Figure 2 shows visualizations of the samples' Total Ion Chromatograms (TIC), which simply sums all intensities within a spectrum for any given sample location.

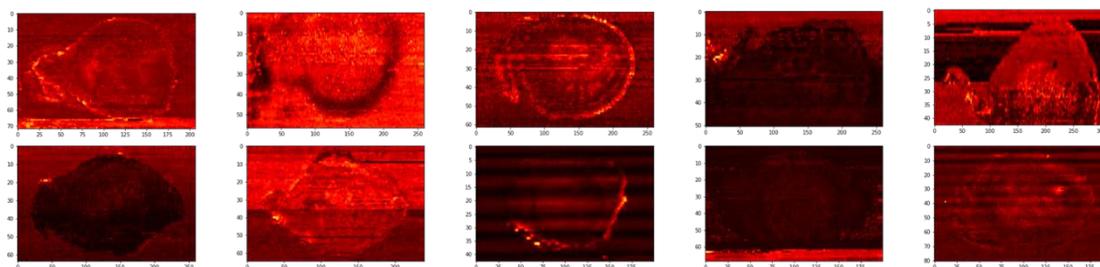


Figure 2: TIC of uterine samples acquired through nano-DESI MSI with reference names available from Table 1, in left-right, top-down order

Several of the provided uterine samples were missing MSI files, thereby lacking entire lines of information. The provided FOV vertical dimensions were therefore adjusted to compensate for the absent lines (Table 1). While undesirable, such data can still be accepted for training, validation, and testing cases within the created/updated algorithms. However, it should be noted that for actual implementation, it is not possible for manual manipulation/obfuscation of acquired MSI files.

Table 1: Provided sample dimensions and information with modified height to account for missing information

Sample Name	Set	Final Width (mm)	Height (mm)	# Missing Lines	Final Height (mm)	Sampling Rate ($\mu\text{m/s}$)
Slide1-Wnt-3	Test	3.1	2.400	3	2.304	15
Slide5-RR-2	Train	3.9	2.135	4	1.995	15
Slide6-V2-2	Test	2.6	1.953	1	1.922	10
Slide6-WT-1	Train	2.6	1.530	0	1.530	10
Slide6-WT-2	Train	3.2	1.333	0	1.333	10
Slide6-Wnt-1	Train	2.6	1.980	2	1.920	10
Slide9-RR-1	Train	3.6	2.450	5	2.275	15
Slide9-V2-3	Train	2.0	1.290	0	1.290	10
Slide9-WT-2	Val.	3.0	2.310	1	2.277	15
Slide9-Wnt-2	Val.	3.0	2.542	1	2.511	15

CHAPTER 3 DEVELOPMENT

3.1 Data Pre-Processing

3.1.1 m/z Visualization

Principally, MSI obtains spectra of measured intensities against mass-per-charge (m/z), for specified locations (Fig. 3). Integrating the obtained spectra values within a given precursor mass tolerance window allows for 2D representation of a molecule's spatial distribution. The window size depends on the physical MSI hardware mass error rating, determined through instrument calibration and ideally taking into account the potential for overlap between molecular isotopes. The instrument mass error is specified in parts-per-million (ppm) and should be known by experimentalists prior to sample acquisitions. The resulting window range in units of m/z , for a given instrument error (Δ) and central m/z value (mz), is found through Eq. 1. An example of a visualized m/z window centered on an m/z of 219.02664 can be seen in Figure 3.

$$[mz * (1 - \Delta * 10^{-6}), mz * (1 + \Delta * 10^{-6})] \quad (1)$$

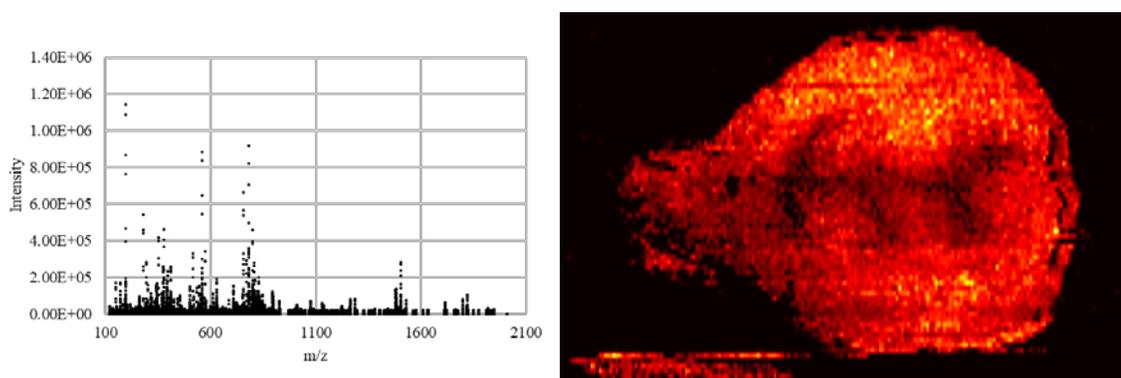


Figure 3: Example spectrum for a scanned position within a uterine sample (left), with a post-processed image (described in Section 3.1.2), extracted 20 ppm m/z visualization, centered on 219.02664 m/z across all spectra (right)

3.1.2 Alignment and Rescaling

Dynamic sampling algorithms require prior knowledge regarding the type of data intended for acquisition. This information can be split into two categories: (1) sample-specific details and (2) sample type. The first includes: the maximum number of lines that are going to be acquired within the equipment's FOV, the FOV's physical width and height, the intended equipment sampling rate, the option of a single representative monoisotopic (or internal standard based) m/z value, the minimum and maximum m/z values expected to be acquired for any location's spectrum, a precursor mass tolerance, and the Fourier transform resolution (used in the production of the spectrum during acquisition). The second refers to a list of m/z values, representative of the molecules desired for acquisition, given a specific tissue type.

As noted in Section 1.2, locations scanned with nano-DESI are not necessarily symmetric in their dimensionality, rather more often existing in an asymmetric coordinate system. Further, these locations neither have the same number of spectra/locations acquired per line, nor are they predictably consistent when location sampling actually occurs. This latter point has been visualized in Figure 4, where the misaligned rows of a uterine sample are re-aligned through linear interpolation to 1,000 new positions, as was done in the initial versions of the provided datasets. This procedure was updated to map original position measurement times according to a set of new times generated according to the originally intended instrument sampling rate and FOV width. The final FOV width is defined by the sample width divided by the scanning rate, rounded to the nearest integer. The final aligned grid for interpolation stretches from 0 to the final width, with regular spacing specified by the scan rate.

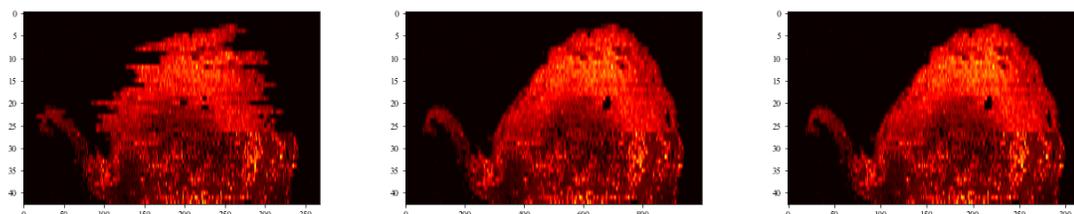


Figure 4: Misaligned visualization of a m/z (left) for a uterine sample (2.6x1.333 mm), early re-alignment with 1000 new values (middle), and its final re-aligned version according to sampling rate

The former issue of non-symmetric (or asymmetric) scanning location dimensionality created issues where, as with the underlying assumption of location size consistency, there was a demonstrated tendency towards sampling along the vertical axis primarily due to the employed reconstruction and RD generation methods. First, IDW mean reconstruction uses the distances between measured and unmeasured locations to inform its resultant values. The vertical distances were shorter in internal representation, compared to their actual physical dimensions, causing a discrepancy. Second, the ground-truth RD is based on the absolute difference between the ground-truth and a reconstruction, thereby more emphasis was placed on scanning vertically, since neighboring horizontal values tended to be less varied in intensity, being physically closer to one another. This ultimately produced “stretching” artifacts in sampling masks and reconstructions, an example of which can be seen in Figure 5.

This behavior was also quite inconsistent, as the vertical spacing between lines and intended horizontal sampling rates (noted in Section 2.2.2) also varied between provided samples. The specified FOV dimensions and intended equipment sampling rate were therefore used to rescale m/z visualizations to ensure scanned locations had symmetric dimensionality, or hereafter described as being in a square coordinate system

(Fig. 6). This rescaling is performed along the vertical axis, in order to preserve the horizontal resolution.

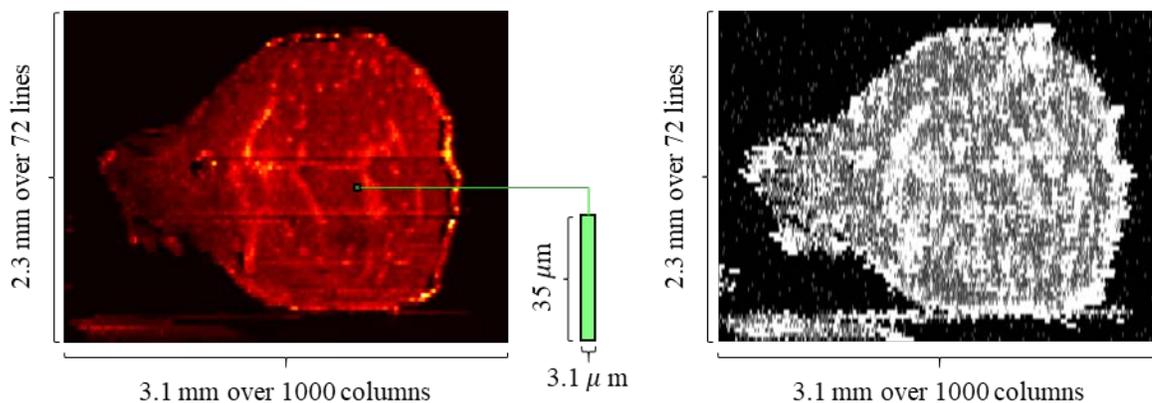


Figure 5: An average of multiple m/z images (rows re-aligned to 1,000 values) for a uterine sample in the original, asymmetric coordinate system (left) with a derived, early SLADS pointwise measurement mask (right), with 40% measured FOV

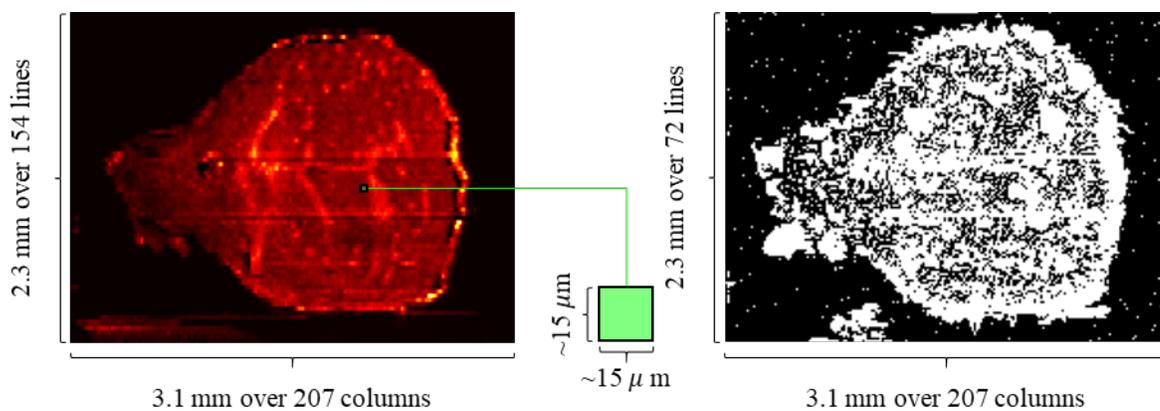


Figure 6: An average of multiple m/z images for a uterine sample in the symmetric, square coordinate system (left) with a derived, early SLADS pointwise measurement mask (right), with 40% measured FOV

3.1.3 m/z Selection

The dynamic sampling algorithms have two primary use cases for MSI applications, where an experimentalist desires to either 1) only obtain data particular to specific molecules, or 2) limit acquisition to the tissue data in the equipment FOV.

Regardless, the experimentalist must possess knowledge of either the m/z corresponding with the desired molecules, or those m/z values commonly representative of the tissue geometry. However, even without the requisite domain knowledge regarding what m/z might be of interest, the second tissue geometry case did provide a straightforward approach for verifying algorithm performance. By calculating the overlap between different visualized spectra and using a foreground mask, a list of representative m/z values could be created and then further compared to isolate m/z common to all samples.

The samples did not possess markers for registration of the MSI data against corresponding optical images and hand-generated binary masks, by domain experts, of the sample foregrounds could not be provided. Therefore, the creation of these masks had to be automated (for consistency) and derived solely from information within the samples' spectra. Within the metadata contained for each of the MSI RAW files, a variable named the "Calculated Monoisotopic m/z " was stored (if determined by the vendor through their proprietary "Xtract algorithm" to have been reliably calculated and enabled by the experimentalist) at the sample acquisition time, which when used to normalize the TIC, appears representative of the foreground. The monoisotopic m/z was not stored for every RAW file, nor identified as the same value even in the same sample. Although the exact algorithm for the value's determination and confidence of its veracity is proprietary, it is defined in documentation as: "The mass-to-charge ratio of the monoisotopic mass that the Xtract algorithm calculated from the isotopic peak envelope for a specific charge state," where the monoisotopic mass is "The weighted average of the monoisotopic masses of each charge state" [25].

A simplified approach for a reliable determination of this value was developed and verified to be within 20 ppm of the averaged, monoisotopic m/z listed in the uterine MSI files. While supporting evidence for this procedure and observations was not found in existing publications [26, 27], there were two cases observed in the available samples. The monoisotopic m/z was the higher value of either 1) the most commonly occurring (the mode) peak in all sample spectra, or 2) the m/z with the highest intensity among the sum of all spectra. The most common non-zero monoisotopic m/z for all MSI files for each sample, as compared with the values determined by this research approach, are seen in Table 2, with the visualizations shown in Figure 7.

Table 2: Determined monoisotopic m/z for uterine samples

Sample Name	Tissue	Mode from RAW	Determined
Slide1-Wnt3	Uterine	560.3673	560.36731
Slide5-RR-2	Uterine	560.3697	560.36969
Slide6-V2-2	Uterine	560.3688	560.36877
Slide6-WT-1	Uterine	560.3711	560.37115
Slide6-WT-2	Uterine	560.3694	560.36945
Slide6-Wnt-1	Uterine	560.3709	560.37085
Slide9-RR-1	Uterine	560.3694	560.36945
Slide9-V2-3	Uterine	560.3698	560.36981
Slide9-WT-2	Uterine	560.3684	560.36841
Slide9-Wnt-2	Uterine	560.3694	560.36932

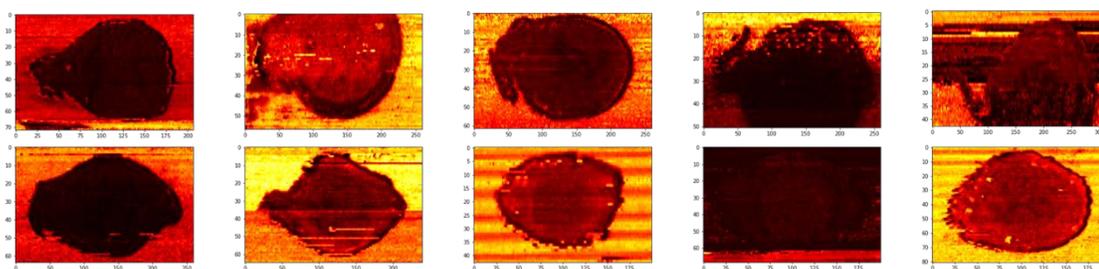


Figure 7: Determined monoisotopic m/z visualized with 20 ppm windows, with reference names available from Table 2, in left-right, top-down order

Since manually produced ground-truth masks could not be provided, nor confidently produced without domain specific knowledge, a combination of image processing techniques was employed to consistently produce foreground masks. The specified parameters are arbitrary, but were empirically found to successfully isolate the primary tissue region of all the uterine samples. Any values less than the 1st percentile were removed to prevent overly expanded values during division. The TIC was normalized by the visualized monoisotopic m/z, with the resultant values biased for cleaner thresholding by setting values lower/equal to the 40th percentile to zero and placing an upper limit at the 70th percentile. Successive operations were applied in the form of Otsu thresholding, morphological closing (dilation followed by erosion), binary dilations to fill small enclosed regions, a Gaussian blur ($\sigma=1$), a final binary thresholding to values above 0.5, whereupon the largest cohesive area was extracted as the final mask. The resultant binary masks for each of the uterine samples are shown in Figure 8.

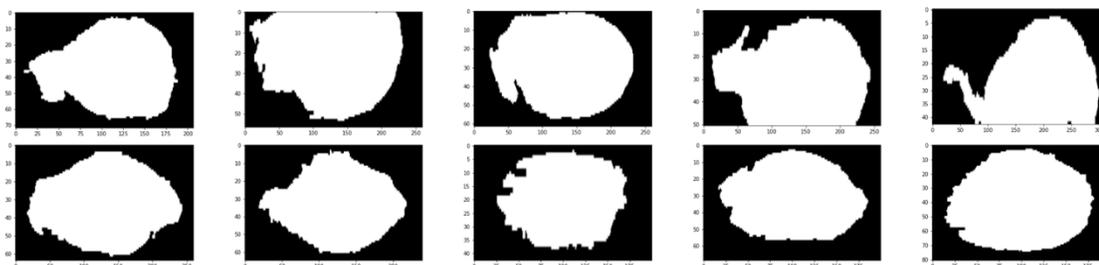


Figure 8: Binary masks of the foreground for each of the uterine samples, with names referentially available from Table 2, in left-right, top-down order

For each sample in a given tissue dataset, the spectra for all scanned locations were merged, summing intensities at identical locations. Starting at the m/z with the highest intensity, any measurements made within its mass tolerance window (20 ppm)

were removed from consideration. This process was continued until all points within the spectra had been examined, thereby removing all overlap between the remaining m/z values' isotopic envelopes. These m/z were visualized and compared against the corresponding foreground masks. Each m/z was scored according to the difference between the Intersection Over Union (IOU) with respect to the foreground and the background (Eq. 2). m/z with a score equal/greater than an empirically chosen value of 0.5 were "chosen" for the sample, those lesser/or equal to 0 were labeled as "very rejected" locations, and all else were noted as "rejected." An example of this distribution, in combination with visualized chosen locations, is shown in Figure 9. The chosen m/z values of all samples with overlapping mass tolerance windows were averaged together to generate a final set of m/z commonly representative of the foreground tissue geometries. The distribution of chosen m/z for each sample can be seen in Figure 10, with visualizations of the common 11 m/z for the uterine dataset shown in Figure 11.

$$Score = \frac{Intersection}{Union}(Foreground) - \frac{Intersection}{Union}(Background) \quad (2)$$

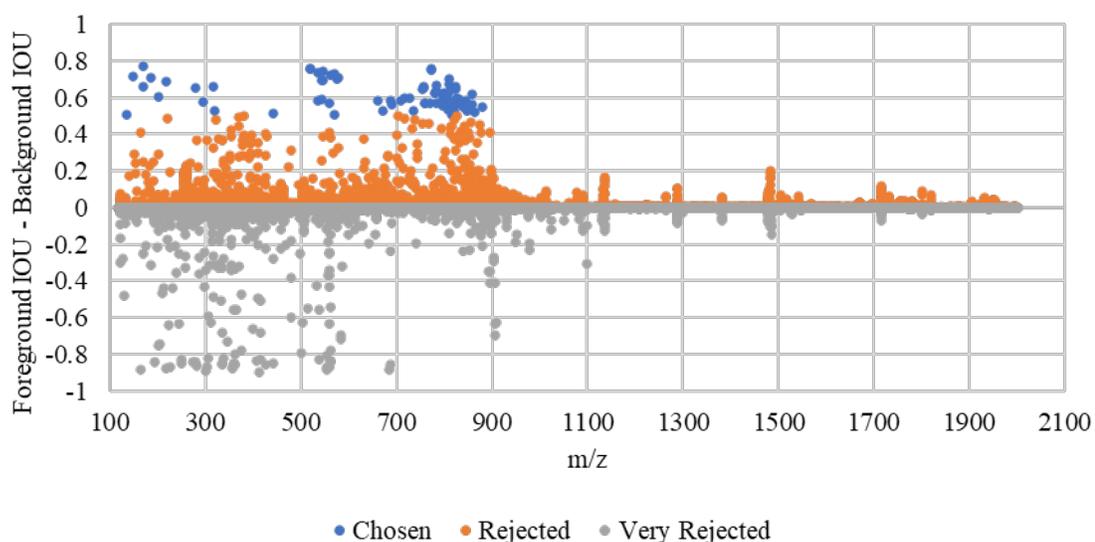


Figure 9: Score distribution of m/z values for the uterine sample Slide1-Wnt3

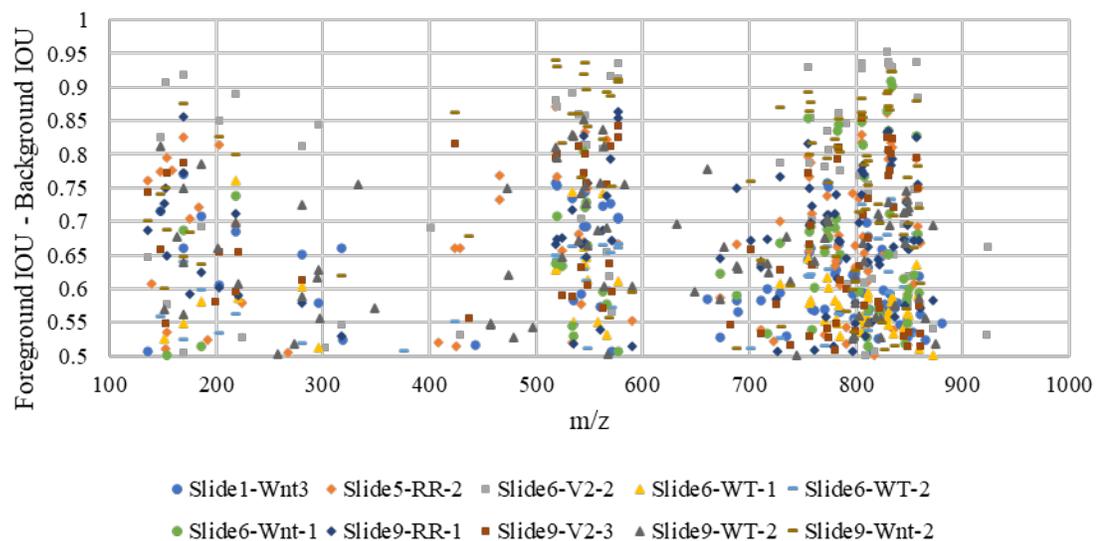


Figure 10: Distribution of chosen m/z values identified for the uterine samples as representative of their underlying geometry

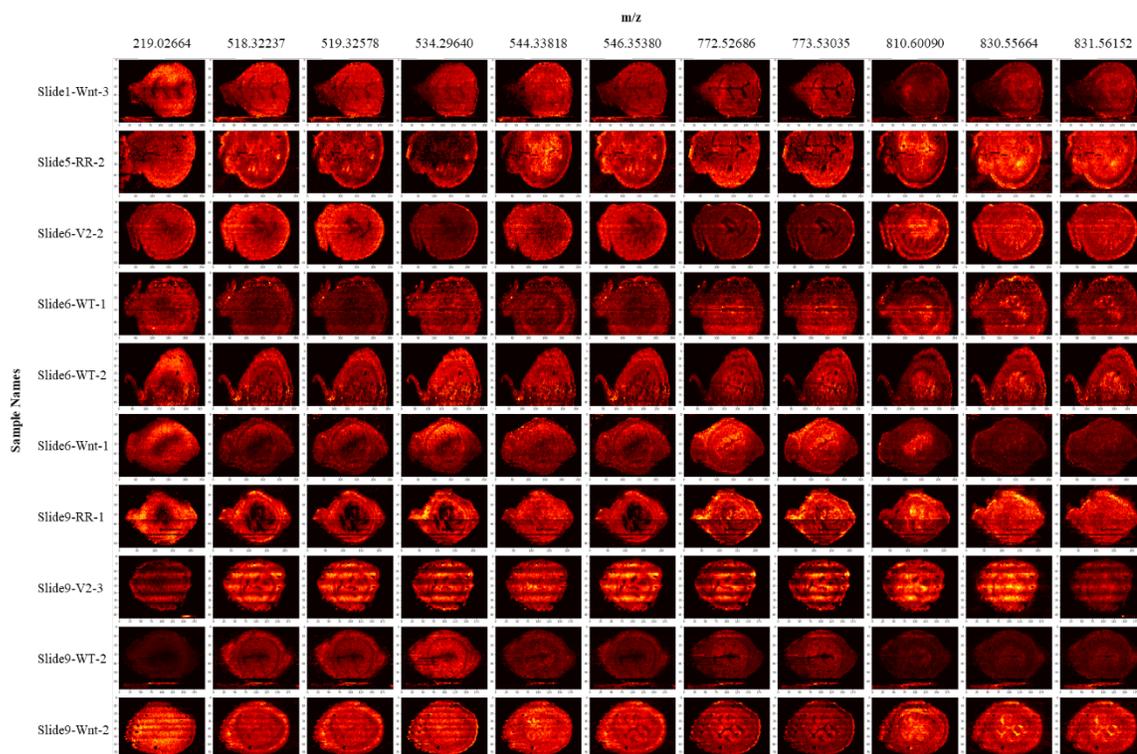


Figure 11: Ground-truth m/z images, normalized by the monoisotopic m/z , found to be commonly characteristic of the underlying uterine sample tissue geometry

3.2 Methods

3.2.1 Sampling Overview

Regardless of particular operational mode or application, an overview of the sampling procedure for SLADS can be seen in Figure 12. First, an initial set of predetermined measurement locations is established with a random 1% of the total FOV area for pointwise and a full line at half of the sample height for linewise. After an initial scan of these locations, a trained model is used to determine an Expected Reduction in Distortion (ERD) for unmeasured locations. Location(s) which maximize the ERD are chosen and passed to the physical scanning equipment. After acquisition has been completed, the sparse measurements are used to perform a reconstruction of the ground-truth using computationally efficient IDW mean interpolation. This process is repeated until pre-specified stopping criteria are met.

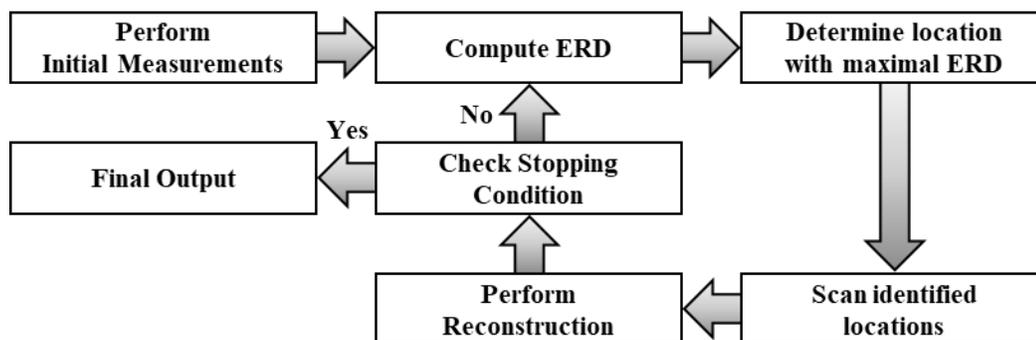


Figure 12: Procedural sampling framework implemented with SLADS and DLADS

Figure 13 shows the overall procedure for training the SLADS-LS, SLADS-Net, and DLADS models. Random pointwise masks, at sampling densities from 1% to 40% of the scanning area, are created for each sample. These masks are applied to the ground-truth image(s), across the m/z channels in the case of nano-DESI MSI data.

Reconstructions are generated from only the measured values and used in combination with the ground-truth data to produce RD (Section 3.3.3). This information, in addition to extracted feature vectors (from the reconstruction produced for the averaged measured m/z) for SLADS-LS and SLADS-Net, constitutes the training and validation datasets, then used to train models for the production of ERD.

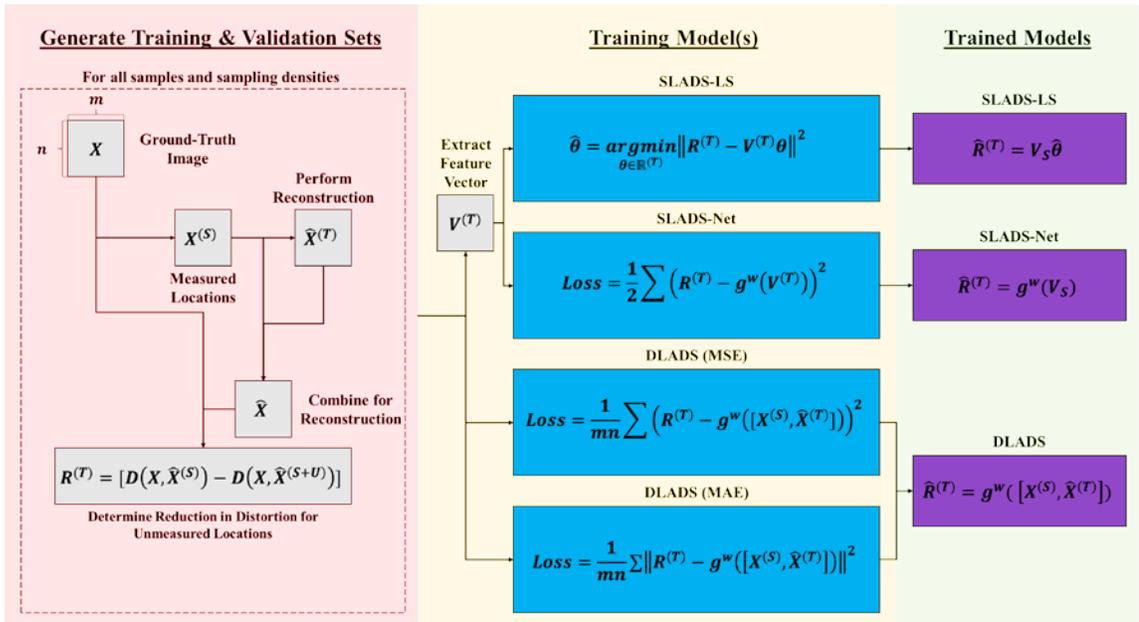


Figure 13: Model training procedure, referencing variables defined in Section 3.2.2

3.2.2 Pointwise Acquisition

Given a ground-truth image X with width m and height n , where Ω is the set of all locations therein, there exists a set S with k measured locations ($S = \{s^{(1)}, s^{(2)}, \dots, s^{(k)}\}$) and corresponding values $X^{(S)}$. The remaining q unmeasured locations are defined in the set T ($T = \{t^{(1)}, t^{(2)}, \dots, t^{(q)}\}$). A reconstruction of the ground-truth image \hat{X} consists of the measured values ($\hat{X}^{(S)} = X^{(S)}$) and reconstruction values for unmeasured locations ($\hat{X}^{(T)}$), determined through IDW mean interpolation.

The dynamic sampling procedure maximizes acquisition of locations where the reconstruction is the most incorrect. The improvement to the reconstruction produced for scanning an unmeasured location is captured in a RD array/image, notated as R , possessing matching dimensions to X . The R value for measured locations ($R^{(S)}$) is 0 and for unmeasured locations ($R^{(T)}$) may be found through Equation 3. Therein, $D(\cdot, \cdot)$ consists of the absolute difference between two given images. During actual implementation, X cannot be known, meaning an ERD \hat{R} for unmeasured locations $\hat{R}^{(T)}$ must be determined based on the currently available information (Eq. 4).

$$R^{(T)} = D(X, \hat{X}^{(S)}) - D(X, \hat{X}^{(S+T)}) \quad (3)$$

$$\hat{R}^{(T)} = \mathbb{E}[R^{(T)} | X^{(S)}] \quad (4)$$

SLADS-LS determines the ERD as the product (Eq. 5) of a feature set for unmeasured locations $V^{(T)}$, extracted from \hat{X} , and parameters $\hat{\theta}$ (Eq. 6), determined through least-squares regression. $V^{(T)}$ consists of measures of gradient, standard deviation, and density as described in the SLADS-LS publication [14].

$$\hat{R}^{(T)} = V^{(T)} \hat{\theta} \quad (5)$$

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^{(T)}}{\operatorname{argmin}} \|R^{(T)} - V^{(T)} \theta\|^2 \quad (6)$$

SLADS-Net uses a MLP network (50 neurons in 5 layers), represented as a function $g(\cdot)$ to produce $\hat{R}^{(T)}$ (Eq. 7). The network weights w are converged upon using an Adam solver with Learning Rate (LR) of 1e-3, minimizing a squared loss (Eq. 8).

$$\hat{R}^{(T)} = g^w(V^{(T)}) \quad (7)$$

$$\text{Loss} = \frac{1}{2} \sum \left(R^{(T)} - g^w(V^{(T)}) \right)^2 \quad (8)$$

DLADS further updates $g(\cdot)$ with a modified U-Net CNN architecture (Section 3.3.2) for the determination of the whole \hat{R} image, though $\hat{R}^{(s)}$ are set to 0 in operation. Applying DLADS with MSE (Eq. 9), or MAE (Eq. 10) losses with only the reconstruction image as input would produce an ERD through Equation 11. Determination of the final inputs, hyperparameters, optimizer, and loss are performed in Section 4.2.

$$Loss = \frac{1}{mn} \sum (R - g^w(\hat{X}))^2 \quad (9)$$

$$Loss = \frac{1}{mn} \sum \|R - g^w(\hat{X})\|^2 \quad (10)$$

$$\hat{R} = g^w(\hat{X}) \quad (11)$$

Given the computational cost of determining a reconstruction after the acquisition of every pointwise position, a group-based pointwise acquisition method was developed. This allows a user to specify a percentage of the total scannable area that should be acquired between reconstruction steps. The algorithm temporarily uses reconstruction values as measured values, then recalculates the ERD, or updates only regions of the RD that are affected.

3.2.3 Linewise Acquisition

Besides the sets of measured and unmeasured points, there exists a set L of horizontal lines/rows in X ($L = \{l^{(1)}, l^{(2)}, \dots, l^{(n)}\}$), comprised of lines with measured points $J \subset L = \{l \in L: (\exists s \in l)\}$ and lines with only unmeasured points $K \subset L = \{l \in L: (\nexists s \in l)\}$. For the scope of this research, the ability to revisit lines, after any points have been scanned on them, was disabled, though this remains a configurable option. The next line to scan is chosen by finding the line with maximal sum ERD, (i.e. $argmax_{l \in K} (\sum \hat{R}^{t \in l})$). The points to be scanned on that line are then either (as specified by the user during configuration) sparsely determined, or selected between a determined

start and stop position. The sparse determination sorts the unmeasured locations on the chosen line and then selects a top percentage (user-defined) of them for acquisition. The start/end point method performs the same operations, but selects all points between the minimum and maximum position for acquisition. These methods are referred to as percent-linewise and segment-linewise respectively. A safety mechanism to ensure the timely completion of a linewise scan was included in this procedure, where if less than 1% of Ω are chosen, then all points on the chosen line are scanned.

3.2.4 *c* Value Regularization

Determination of the actual RD for use in model training was notably problematic, in terms of computational expense, both within the original SLADS-LS and SLADS-Net publications. Both in the original publications and herein, the regression models are constructed based on a training database, where for each training sample, random sampling masks are produced at h densities in the predefined set: $P = \{p_1, p_2, \dots, p_h\}$. For every potential unmeasured point, at every considered density, a reconstruction has to be generated, considering if that point had been measured. For nano-DESI data, where multiple m/z are being considered, this becomes an even greater concern, requiring additional calculations and time proportional to the number of m/z . In order to make the generation of a training database more tractable, SLADS uses an approximated RD, by only considering a local region of influence for each unmeasured point, and limiting the strength of that influence by a regularization parameter c . This method has also been adapted in DLADS and the updated SLADS algorithms.

Given an unmeasured location $t^{(1)}$, a weighted factor σ can be calculated for it (Eq. 12) as the distance to the nearest measured value, divided by a c value. $R^{(t^{(1)})}$ then

can be approximated (Eq. 13) according to $\sigma(t^{(1)})$, applied to the distortion between the reconstruction ($\hat{X}^{(S)}$), without having measured $t^{(1)}$ and X .

$$\sigma^{t^{(1)}} = \frac{\min_{s \in S} \|s - t^{(1)}\|}{c} \quad (12)$$

$$R^{(t^{(1)})} \approx \sum_{s \in S} \exp \left\{ -\frac{1}{2(\sigma^{t^{(1)}})^2} \|s - t^{(1)}\|^2 \right\} D(X, \hat{X}^s) \quad (13)$$

Originally, a window W was used in SLADS, manually set at an optimized static size of 15x15 to bound the considered region for each unmeasured location, further reducing computational overhead. This research includes an option for a dynamic variation, with a radius set at 3 times the sigma value of each unmeasured point. The radius is doubled and rounded up to the nearest odd integer to ensure that the produced Gaussian signal is centered on the specified location. Zero padding is then used for locations where the radius overlaps with the image dimensions.

The parameter c still needs to be optimized for each application, with o possible values in a user defined set: $C = \{c_1, c_2, \dots, c_o\}$. Original SLADS variations trained multiple models, one for each potential optimization parameter c . The model and c value, which minimized the total distortion between ground-truth and reconstruction $D(X, \hat{X})$, over a simulated scan's progression, was selected for testing and implementation.

This research updated the implementation of SLADS and DLADS to determine an optimal parameter prior to any model training, by simulating the use of each potential c with an oracle, determining the approximated $R^{(U)}$ to guide measurements. Rather than the relative distortion, the Peak Signal to Noise Ratio (PSNR) (Eq. 14), based on the Mean Squared Error (MSE) (Eq. 15) was considered. The parameter, which maximizes the averaged PSNR of all m/z image reconstructions over the course of simulated

scanning, is selected for training the indicated regression model. Particularly for the neural network implementations of DLADS and SLADS-Net, this reduces the required training time according to the number of possible c values.

$$PSNR = 10 \log_{10} \frac{\max X^2}{MSE} \quad (14)$$

$$MSE = \frac{1}{mn} \sum (X - \hat{X})^2 \quad (15)$$

3.3 Updated SLADS and DLADS

3.3.1 Code Design and Notes

The updated form of SLADS and the new DLADS architecture are combined in a single program for ease of comparison and configuration, all run within a single “root” directory. A link to a repository containing this program can be found in Section 2.1.1, therein exists more details regarding configuration, third-party package requirements, and operation for versions produced during this project’s development.

Overall, there are 3 principal operating modes: training, testing, and implementation, which can be enabled in a configuration file, or files to be run in sequence. Every sample must have a set of prior knowledge included with it, as discussed in Section 3.1.2. While it is recommended that the specified sampling rate matches that specified for the actual scanning equipment, a higher sampling rate could be specified to allow for greater horizontal position specificity. This may be desired if the physical equipment becomes capable of varying the precision of movement/speeds.

The vendors for a number of MSI platforms, including the hardware used to acquire the nano-DESI samples herein, have released dynamic linked libraries for allowing external programs to read their proprietary file formats. Python bindings to these libraries have been published in the third-party library, Multiplerz [28]. This

reliance on vendor-provided libraries generally limits functionality to only the Windows operating system, though a workaround (the procedure for which is described in the linked repository) was determined for the RAW files used for this project, enabling functionality on mainstream Linux distributions, including CentOS 7-8 and Ubuntu 18.04-20.04. The ability for the updated SLADS and DLADS to directly read MSI files, removes an early source of significant computational overhead, where MSI files had to be processed externally, saved onto disk, and then re-read into memory for actual use.

For the training mode, samples placed in the appropriate subdirectory are read in and split into training and validation sets as the user configures. When information is read from the MSI files, for calculation of m/z reconstructions, RD, and ERD, square dimensionality (discussed in Section 3.1) must be maintained to prevent stretching artifacts. However, actual measurement locations must be determined and stored in the original “asymmetric” dimensionality for compatibility with physical equipment.

Both training and validation samples are passed to the optimization routine for c values. If only a single c was provided, then it is automatically returned as the optimal parameter. Otherwise, simulated pointwise scanning of the samples is performed, with a random 1% of all possible locations measured before using an oracle and the approximate RD to inform future scanning locations. The PSNR of the reconstructions for all m/z images at each measurement step is stored, with the resulting curves averaged across all samples for a given c value. Integrating the area under the curves, the c value with the highest corresponding value is chosen as optimal (performed for different configurations in Section 3.4). This value is passed to a method for generation of the training and

validation databases. The resulting information is stored to disk for consistency during potential model ablation studies.

The training and validation datasets then pass to the actual model training routines. Only the training database is used for training SLADS, SLADS-Net, and DLADS models, with the validation set reserved for use in early stopping criteria in DLADS (Section 3.3.2), as well as for optimizing hyperparameters and choosing optimal models for final evaluations (Section 4.2). The produced model is saved to disk allowing for easy distribution and use in actual deployments.

3.3.2 DLADS Architecture

The fundamental structure of the employed neural network is a version of the U-Net architecture (Fig. 14) [29], modified to take in single or multiple channels of 2D data and produce a single 2D ERD image with spatial dimensions corresponding to the input(s). While different combinations of inputs were simulated (Section 4.2.1), ultimately this network uses the reconstructions of the 11 chosen m/z images (Section 3.1.3), each split into 2 images, one containing the measured values and the other with the reconstruction values (for a total of 22 channels with m by n dimensions), as inputs.

The inputs are first passed through 2 back-to-back 2D convolutional layers and then another 2D convolutional layer with a stride of 2, in order to halve the width/height. Doubling the number of filters at each “depth”, this process is repeated until reaching a bottleneck layer, whereupon the data is progressively upscaled with bilinear interpolation (in combination with another 2D convolutional layer) and run through another 2 back-to-back 2D convolutional layers. This continues until reaching the original input dimensions. After each upscaling, a skip connection is used to combine the progressed

output with the corresponding outputs produced during the downsampling half of the network. All of the convolutional layers use 3x3 kernels, He initialization (following common practice), and ReLU activations, with the exception of the final 3 which use 1x1 kernels. Bilinear upscaling is used to avoid checkerboard artifacts [30] during upscaling, produced by the original U-Net's usage of convolutional transposition layers. Dropout layers with a rate of 0.5 are used on the bottleneck and preceding layer to additionally regularize the network, reducing overfit. Batch sizes were kept to 1 in order to allow dynamically sized input/outputs without resizing, or padding.

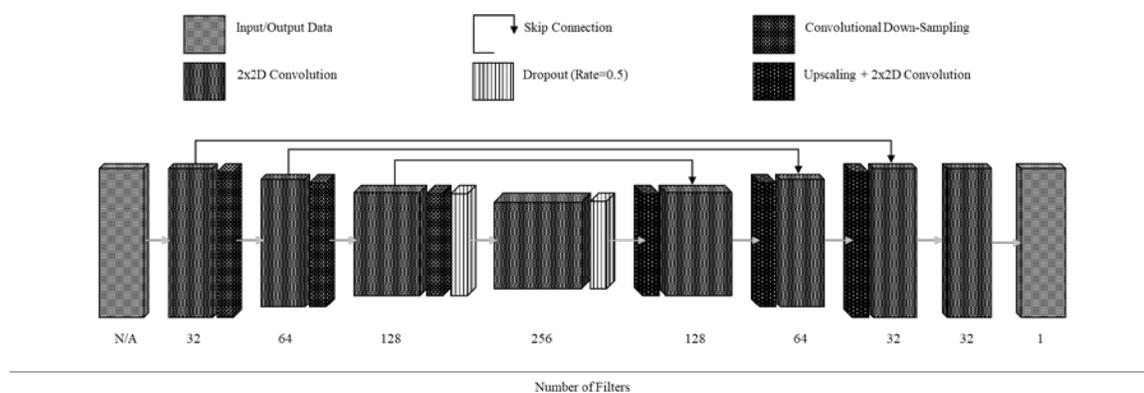


Figure 14: U-Net architecture modified for a image-to-image translation task, taking single or multiple 2D m/z visualizations/features as input(s) and producing a single-channel 2D ERD image with spatial dimensions matching with the input(s)

This architecture was also designed with configurable early stopping criteria, kept consistent throughout all simulations run herein. A minimum of 10 epochs are performed before starting to consider the early termination criteria and saving the best-found weights. If a model's validation loss fails to be reduced within 50 epochs, training is stopped, with the model weights restored where the validation loss was minimal.

An optional data augmentation stage was also implemented for application to and supplementation of the input training data. If enabled, this step will create additional

variations of the input set (number of times is a user-configurable option), randomly performing the following operations: horizontal and vertical flips, width and height adjustments within 25% of the input dimensions, rotation within 45 degrees, and translation within 25% of the starting position. A random sampling of a demonstration image [31] passed through these steps is shown in Figure 15.

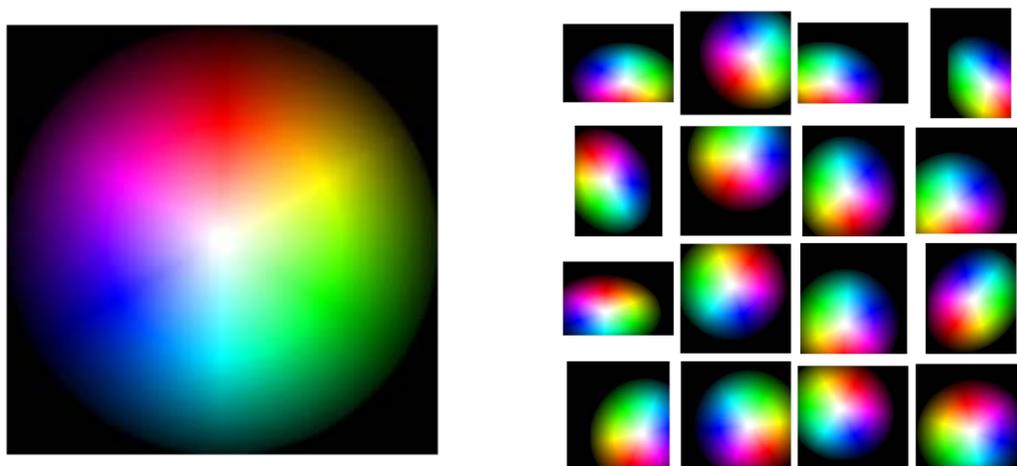


Figure 15: An image of a color wheel (left) passed through the data augmentation stage to create additional random variations (right)

3.3.3 Reduction in Distortion

Section 3.2.3 describes the process by which the RD is approximated in order to make its calculation tractable for practical implementation with 2D images. However, as noted in Section 1.2.4 and similarly deduced during development, the use of a third dimension presents the risk of obfuscating data when compressing the data through dimensional reduction. Figure 16 shows two different procedures for handling the 3D MSI data, where the 3rd dimension comprises different m/z windows. Starting with what is called the original averaging RD generation method, this averages together the m/z channels together into singular 2D images. Thereafter, the same approach used in the

original SLADS and SLADS-Net publications are used to produce the approximated RD: taking the difference between the reconstructed and ground truth images, applying Gaussian windows to the regions centered on unmeasured locations, and then summing the resultant values.

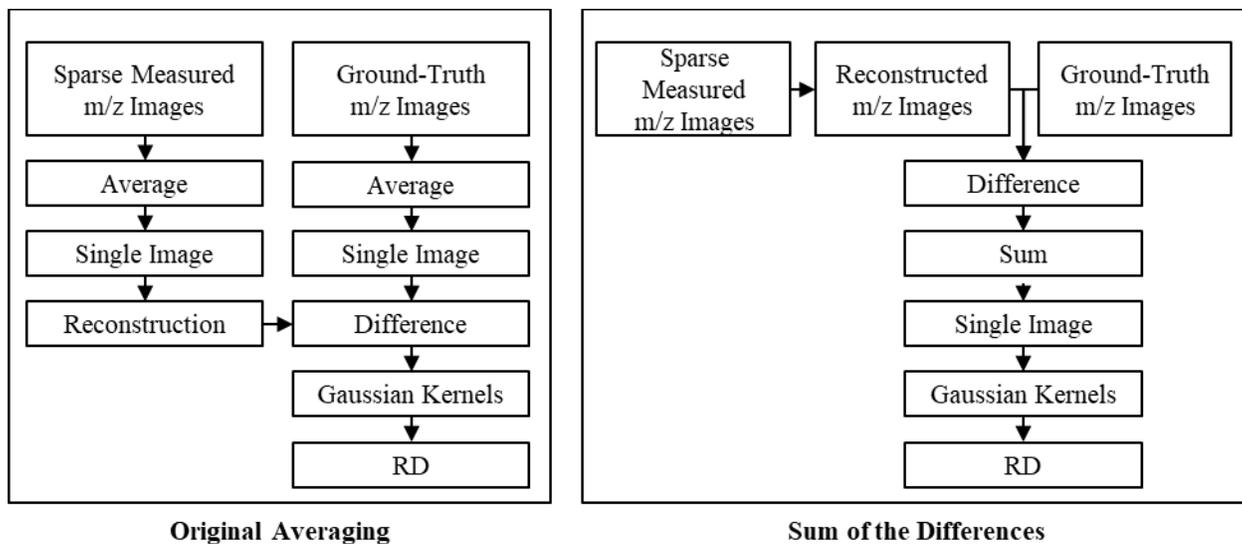


Figure 16: Overview of the original RD generation method (left) and new summation RD generation method (right)

Since the m/z images may hold geometrically diverse patterns or large variations in value intensities, some subset of channels or even a single one could dominate the determination of the RD for the whole sample. In order to overcome this limitation and remove the potential for bias towards any particular m/z channel (based on geometry or values), an alternative sum of the differences, or summation RD generation method, was devised. Reconstructions are performed for each m/z image/channel independently and the difference taken with respect to their ground-truth counterparts. This 3D set of differences is then summed together to produce a single 2D image, after which the multiplication by (a) Gaussian window(s) and final RD generation steps are performed.

3.4 c Value Optimization

Simulated pointwise scanning was performed on the training and validation samples for potential c values in the set: [1, 2, 4, 8, 16, 32, 64, 128, 256]. The average PSNR of the IDW reconstructions generated for the chosen m/z (Section 3.1.3) were then averaged across the samples. The c value with the greatest Area Under the resulting Curves (AUC) was selected as the optimal value. Three studies were performed to consider: 1) RD generated with the static 15x15 window and original RD generation method (used in the original SLADS-LS and SLADS-Net), 2) the dynamic window paired with the original RD generation method, and 3) the dynamic window paired with the new summation RD generation method. The results for these studies are shown in Figures 18, 19, and 20 respectively, with the corresponding AUC distributions shown in Table 3 and Figure 17. Regardless of the window type and RD generation method used, the best c value was found to be 16. When using the original RD generation, the dynamic window was comparable in performance to the optimized static window. The summation RD showed an improvement in the resulting AUC over both original variants.

Table 3: AUC for varying c values with static and dynamic windows with original and summation RD generation methods

		AUC		
		Static	Dynamic	Dynamic
		Original	Original	Summation
c Value	Window			
	RD Method			
	1	1157.424	1176.046	1173.251
	2	1172.115	1179.733	1181.947
	4	1192.439	1194.218	1201.159
	8	1205.932	1207.109	1216.138
	16	1220.652	1220.411	1231.677
	32	1219.602	1219.602	1231.569
	64	1217.172	1217.172	1229.845
128	1217.537	1217.537	1229.18	
256	1217.537	1217.537	1229.18	

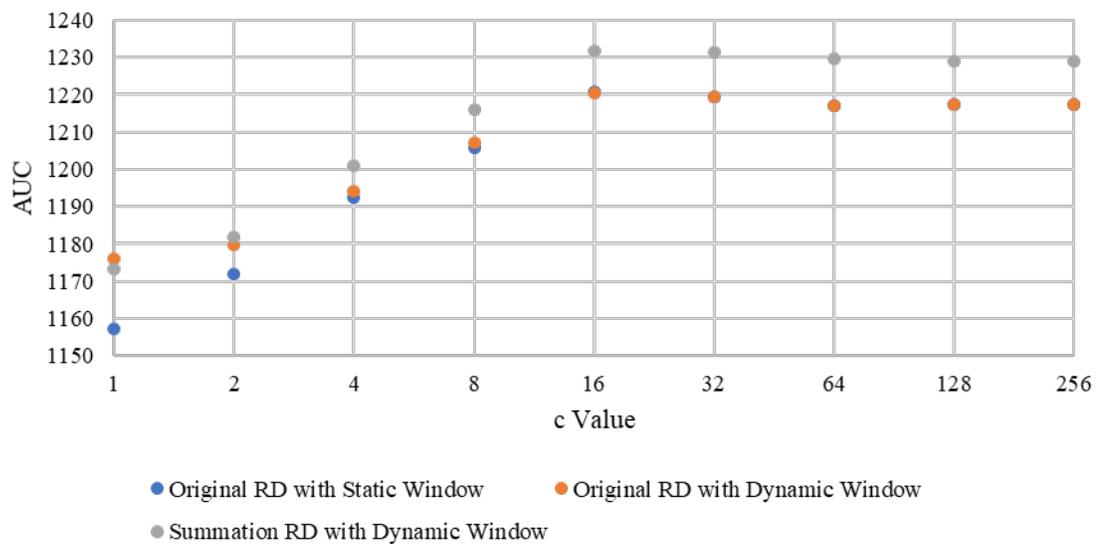


Figure 17: Distribution of AUC for varying c values with different window and RD generation methods

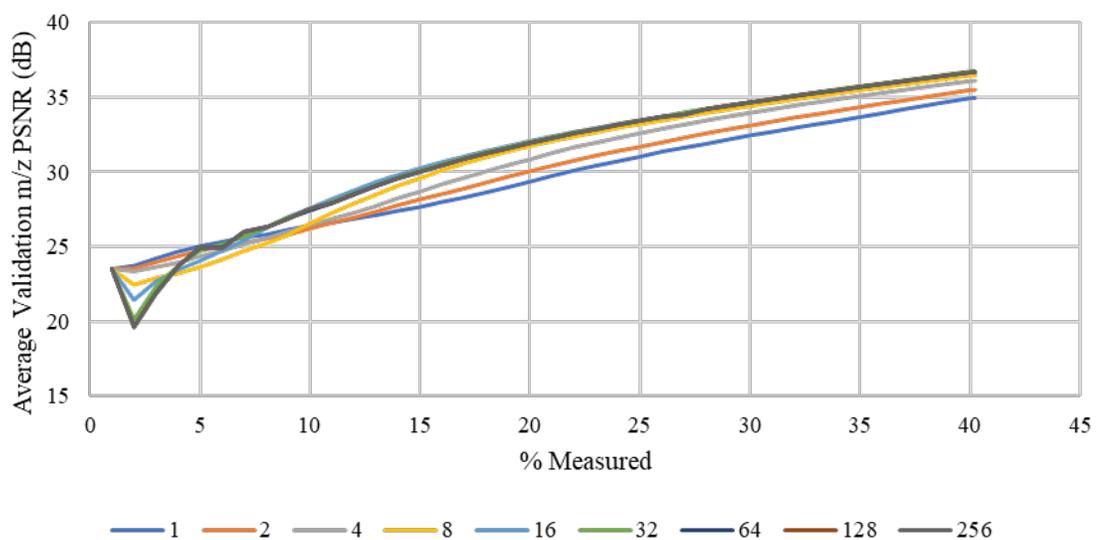


Figure 18: Progression of average PSNR (dB) of m/z IDW reconstructions with a static window and original RD generation method for varying c values

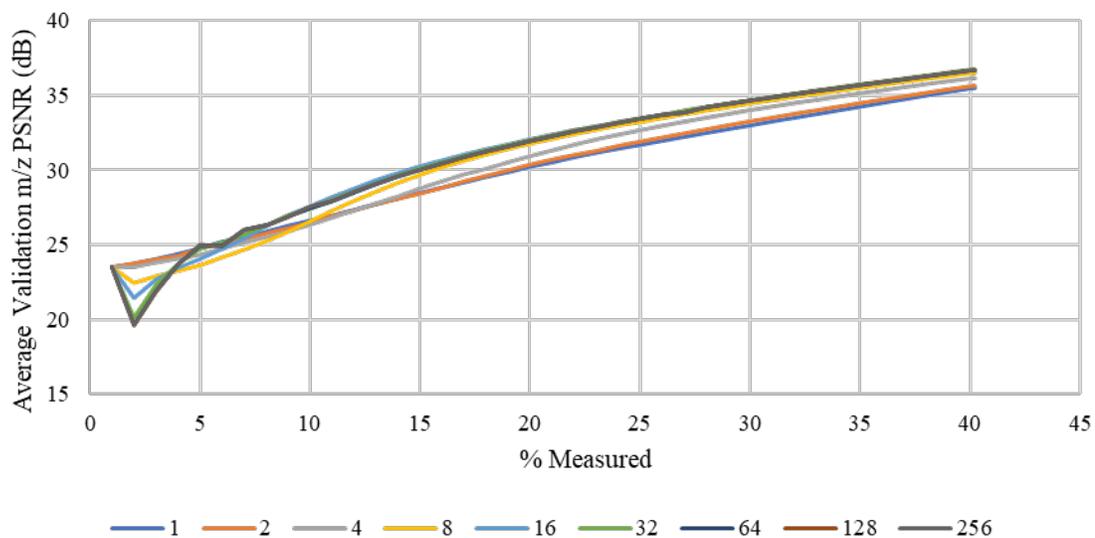


Figure 19: Progression of average PSNR (dB) of m/z IDW reconstructions with a dynamic window and original RD generation method varying c values

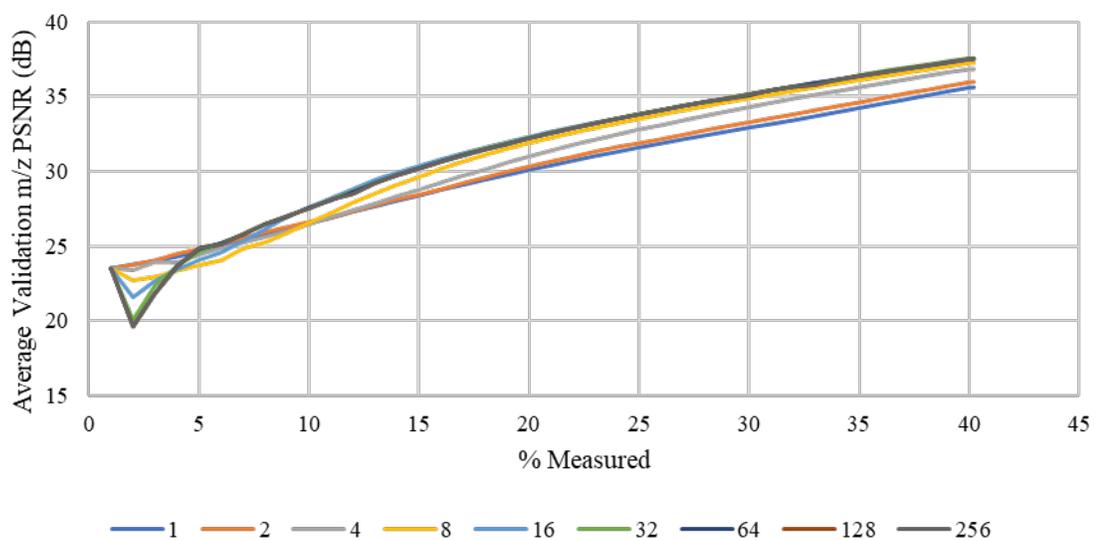


Figure 20: Progression of average PSNR (dB) of m/z IDW reconstructions with a dynamic window and summation RD generation method varying c values

3.5 Physical Integration

In preparation for actual integration with physical nano-DESI MSI hardware (Fig. 21), software mechanisms have been implemented and validated in simulation. Due to the proprietary nature of the MSI platforms, there cannot be direct message passing between

the dynamic sampling algorithms and the actual hardware. However, this project's collaborators devised a LabVIEW interface to provide a limited degree of control allowing for the selection of a position in the vertical axis, along with horizontal positions to start and stop measuring. In order to maximize potential for future integration with a variety of scanning platforms, a simple system was devised to signal the equipment. Enabling the implementation flag during configuration, SLADS/DLADS will wait for a file to be placed in a specific location, which triggers the algorithm to read any data files. This information is processed into awaiting data structures, constructed based on information provided to the program prior to initialization. The measured information then passes through a SLADS/DLADS model to produce an ERD, whereby positions to scan are determined and saved to a new file, signaling the equipment. This process repeats until the algorithm reaches its specified termination criteria.

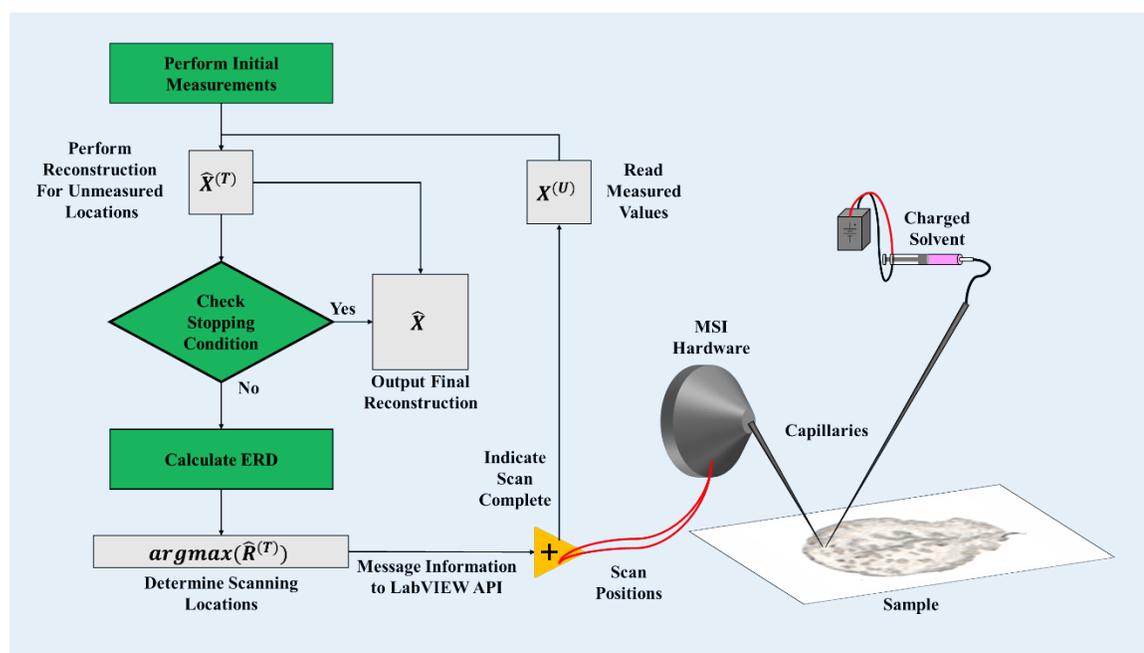


Figure 21: Overarching view of the proposed integration between SLADS and DLADS algorithms with physical nano-DESI MSI hardware

CHAPTER 4 RESULTS

4.1 Metrics

Trained SLADS and DLADS models are evaluated through simulated scanning of the testing samples for final evaluation and through the validation samples for optimization/ablation of different models. Principal metrics considered are: the averaged PSNR (Eq. 14) scores of IDW reconstructions for each m/z channel (Section 3.1.3), the average percent measured for a validation/testing set to reach 33 dB PSNR, and the best Mean Absolute Error (MAE) (Eq. 16) or MSE (Eq. 15) loss.

$$MAE = \frac{1}{mn} \sum |X - \hat{X}|^2 \quad (16)$$

The 33 dB PSNR threshold for IDW reconstruction of the m/z visualizations was specified empirically, as providing reasonable reconstructions of the tissue structures, though values between 35-40 dB were seen to be visually indistinguishable from the ground-truth. PSNR was chosen as the principal evaluation metric due to its prior use in the original SLADS-LS and SLADS-Net publications. It should also be noted that the ERD and RD are rescaled to between 0 and 1 before evaluation, principally because interest should be focused on the relative variation between them (i.e. the difference between which position(s) the ERD and RD indicate should be scanned next).

4.2 Ablation Studies

For all simulations within this section, the initialization follows the sampling procedure (Section 3.2.1), with a consistent seed used for initial mask generation. The summation RD method (Section 3.3.3) with a consistent c value of 16 (Section 3.4) was employed to generate the ground-truth RD throughout this section and Section 4.3.

4.2.1 DLADS Inputs

Different inputs with a standard version of the modified U-Net are evaluated to find which produces an ERD useful during scanning. This network comprises the structure in Section 3.2.2 with a Nadam optimizer, Mean Absolute Error (MAE), a LR of 1e-4, summation RD, dynamic window, no additional augmentation, and simulation with pointwise acquisition (termination at 40% measured). Results are tabulated in Table 4 and the progressive average PSNR of the m/z reconstructions shown in Figure 22. The best set of inputs was the reconstruction and measured values for each m/z. A model with reconstruction and measured values had the least training time needed to reach the 33 dB PSNR, was the first to reach that threshold, produced the highest PSNR, and required the least training relative to that final PSNR.

Table 4: Results for varying inputs to a standardized version of the modified U-Net

Ref.	Inputs [# of Channels]	Val. Loss	Best Epoch (BE)	Avg. % to 33 dB PSNR (AdB)	AdB ⁻¹ *BE ⁻¹	Final Avg. PSNR (dB) of m/z Recons. (FdB)	FdB/BE
0	Reconstruction values for each m/z [11]	0.0858	10	23.887	0.00419	35.247	3.525
1	Measured values for each m/z [11]	0.09876	116	26.526	0.00032	35.598	0.307
2	Reconstruction and measured values for each m/z [22]	0.08233	10	21.894	0.00457	36.645	3.665
3	Averaged reconstruction values across all m/z [1]	0.08002	26	23.326	0.00165	35.673	1.372
4	Averaged measured values across all m/z [1]	0.09801	110	29.294	0.00031	34.834	0.317
5	Averaged reconstruction and measured values across all m/z [2]	0.0784	18	22.100	0.00251	36.634	2.035

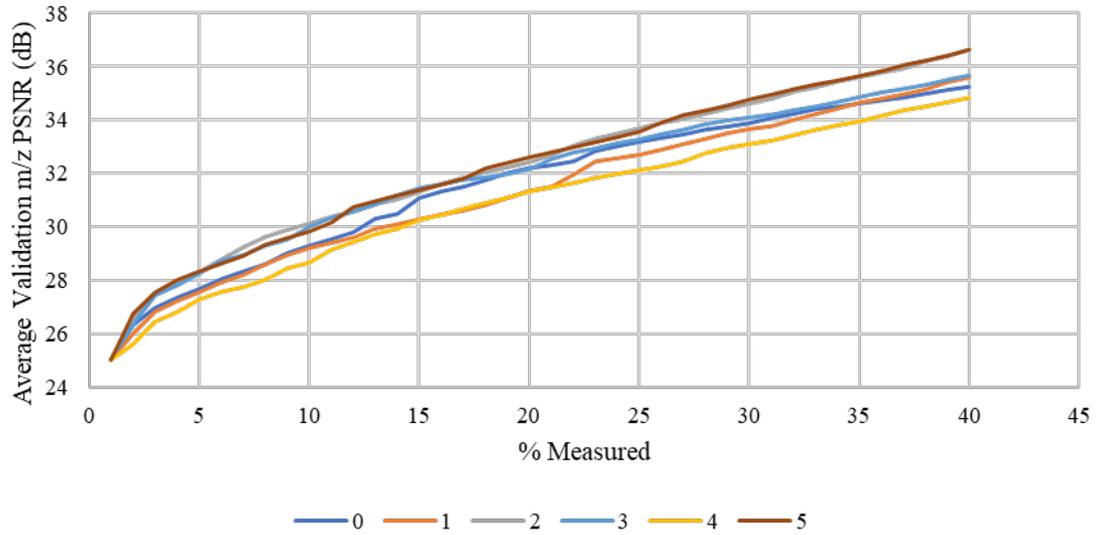


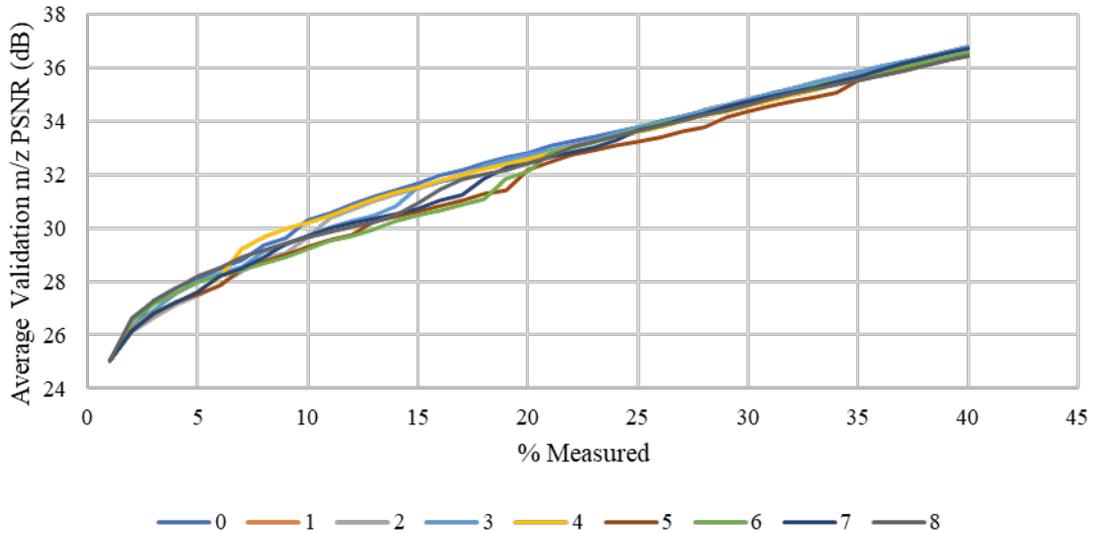
Figure 22: Progressive average PSNR (dB) of m/z reconstructions for simulated pointwise scanning of the uterine validation set, referencing labels in Table 4

4.2.2 DLADS Model Parameters

Using the reconstruction and measured values for each m/z as input, a secondary study was similarly conducted to find which combination of network parameters would produce the most effective ERD, with the least amount of training. This was accomplished by varying the model's loss function (considering MAE and MSE), optimizer (considering Adam, Nadam, and RMS Propagation), and LR (considering $1e-3$, $1e-4$, and $1e-5$). The results are tabulated in Table 5 and the progressive average PSNR of the m/z reconstructions with MAE and MSE shown in Figures 23 and 24 respectively. DLADS trained with a Nadam optimizer and LR of $1e-3$ failed to remain stable for both MAE and MSE losses. The Adam optimizer with a LR of $1e-4$ and MAE loss provided the best overall training time (best epoch) to performance ratios, both in terms of final PSNR and ability to reach the 33 dB PSNR threshold.

Table 5: Results for varying loss, optimizer, and LR with the modified U-Net

Ref.	Optimizer	LR	Val. Loss	Best Epoch (BE)	Avg. % to 33 dB PSNR (AdB)	AdB ⁻¹ *BE ⁻¹	Final Avg. PSNR (dB) of m/z Recons. (FdB)	FdB/BE
Loss			MAE					
0	Adam	1e-3	0.08176	39	20.726	0.00124	36.756	0.942
1	Nadam		N/A	N/A	N/A	N/A	N/A	N/A
2	RMSProp		0.08305	108	21.531	0.00043	36.774	0.341
3	Adam	1e-4	0.07914	13	21.464	0.00358	36.793	2.830
4	Nadam		0.08097	29	21.783	0.00158	36.529	1.260
5	RMSProp		0.08365	25	23.492	0.00170	36.444	1.458
6	Adam	1e-5	0.08042	132	22.013	0.00034	36.563	0.277
7	Nadam		0.07982	114	22.912	0.00038	36.715	0.322
8	RMSProp		0.08454	65	21.906	0.00070	36.444	0.561
Loss			MSE					
9	Adam	1e-3	0.09315	33	20.312	0.00149	36.845	1.117
10	Nadam		N/A	N/A	N/A	N/A	N/A	N/A
11	RMSProp		0.07758	65	21.959	0.00070	36.800	0.566
12	Adam	1e-4	0.07482	107	21.347	0.00044	36.869	0.345
13	Nadam		0.07582	30	21.302	0.00156	36.590	1.220
14	RMSProp		0.07709	113	21.687	0.00041	36.686	0.325
15	Adam	1e-5	0.07399	201	23.271	0.00021	36.419	0.181
16	Nadam		0.0883	307	23.811	0.00014	35.278	0.115
17	RMSProp		0.09603	51	30.115	0.00065	34.656	0.680

**Figure 23:** Progressive average PSNR (dB) of m/z reconstructions for the uterine validation set using the MAE loss function, referencing labels in Table 5

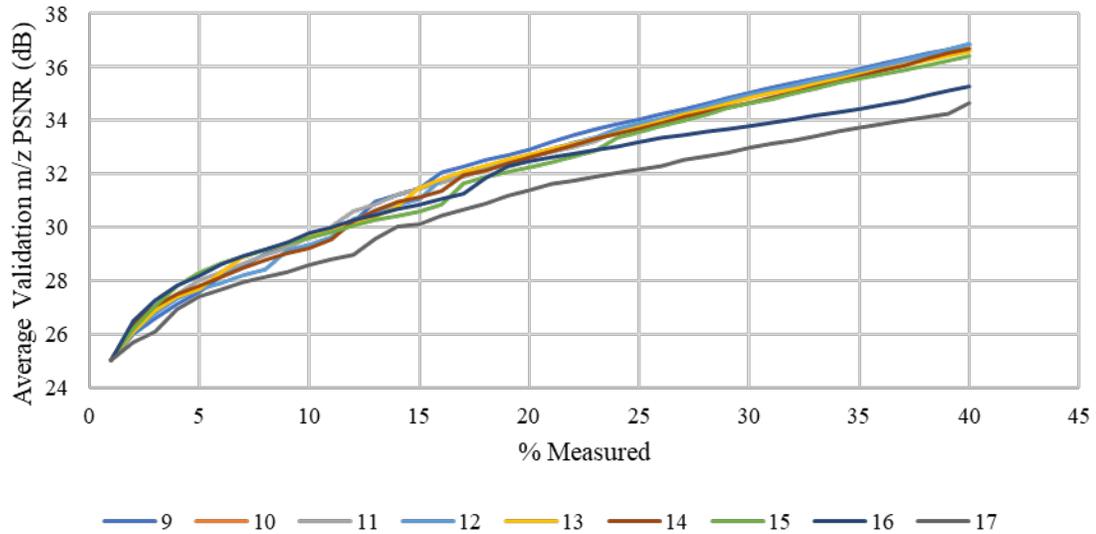


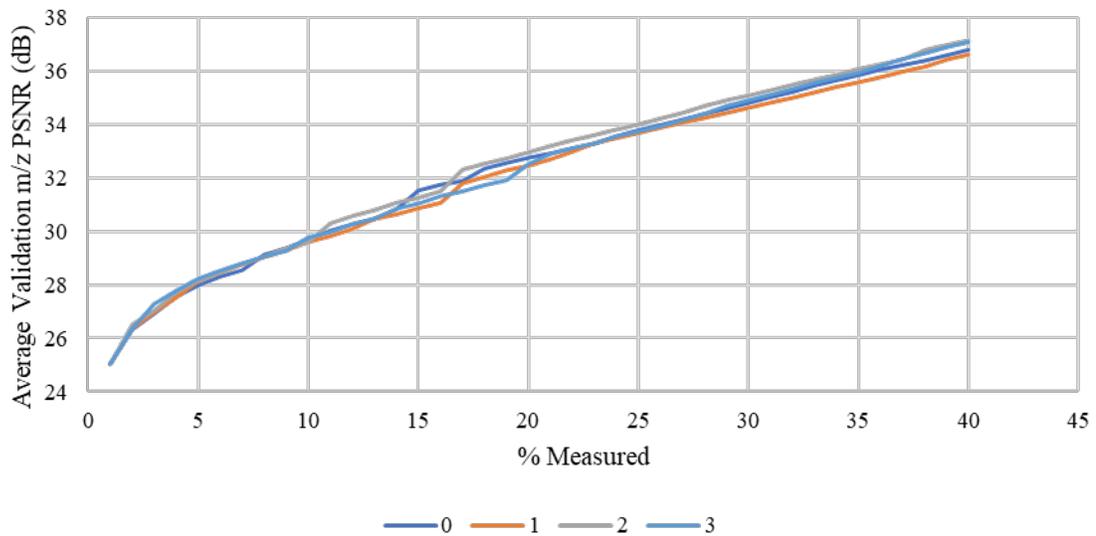
Figure 24: Progressive average PSNR (dB) of m/z reconstructions for the uterine validation set using the MSE loss function, referencing labels in Table 5

4.2.3 DLADS Augmentation

Another study was conducted examining use of the augmentation functionality (Section 3.3.2). Again using the reconstruction and measured values of the m/z images as inputs), with a Adam optimizer, MAE loss, and a LR of $1e-4$, 3 models (doubling, tripling, and quadrupling the training dataset) were compared against the previous simulation without augmentation, each run possessing 240, 480, 720, and 960 training images respectively. Table 6 summarizes the results and Figure 25 illustrates the progressive average PSNR of the m/z reconstructions. Although there are gains at increased levels of augmentation, there is a steep drop, in terms of computational expense, relative to the number of involved training steps (i.e. the best epoch found times the number of training images), to yielded performance. The best relative performance was achieved without dataset augmentation, though this strongly indicates that more varied initial data could lead to further advancements.

Table 6: Results for augmenting input image set for training the modified U-Net

Ref.	Input/Output Pairs	Val. Loss	Best Epoch [# of Steps]	Avg. % to 33 dB PSNR (AdB)	AdB ⁻¹ *Steps ⁻¹	Final Avg. PSNR (dB) of m/z Recons. (FdB)	FdB/Steps
0	240	0.07914	13 [3120]	24.464	0.000013	36.793	0.012
1	480	0.07706	45 [21600]	22.072	0.000002	36.614	0.002
2	720	0.0757	40 [28800]	20.187	0.000002	37.144	0.001
3	960	0.07728	38 [36480]	21.492	0.000001	37.078	0.001

**Figure 25:** Progressive average PSNR of m/z reconstructions with augmentation for the uterine validation set, referencing labels in Table 6

4.2.4 SLADS

In order to examine the relative merit of the introduced dynamic window and summation RD generation method, variations considering the static window and original RD generation method were simulated with the SLADS-LS and SLADS-Net models. An additional simulation was conducted using only a single chosen m/z (219.02664) as an input, with evaluation performed over the multiple common m/z, to highlight the advantage of considering multiple channels. Table 7 summarizes the results and Figures

26 and 27 illustrate the progressive average PSNR of the m/z reconstructions for SLADS-LS and SLADS-Net respectively. Both models performed very similarly, though the former was able to be the first to 33 dB with summation RD generation and a dynamic window and had the highest final PSNR with original RD generation paired with a static window. Given an emphasis on maximizing throughput, the summation RD with dynamic window model is found to be the optimal SLADS implementation.

Table 7: Results for varying the input, window, and RD generation in SLADS models

Ref.	m/z	RD	Window	Avg. % to 33 dB PSNR	Final Avg. PSNR (dB) of m/z Recons.
Model	SLADS-LS				
0	Single	Original	Static	29.871	34.935
1	Multiple			23.359	36.374
2		Dynamic	23.278	36.357	
3			Summation	22.717	36.336
Model	SLADS-Net				
0	Single	Original	Static	28.347	34.879
1	Multiple			22.854	36.120
2		Dynamic	23.081	36.080	
3			Summation	23.498	36.182

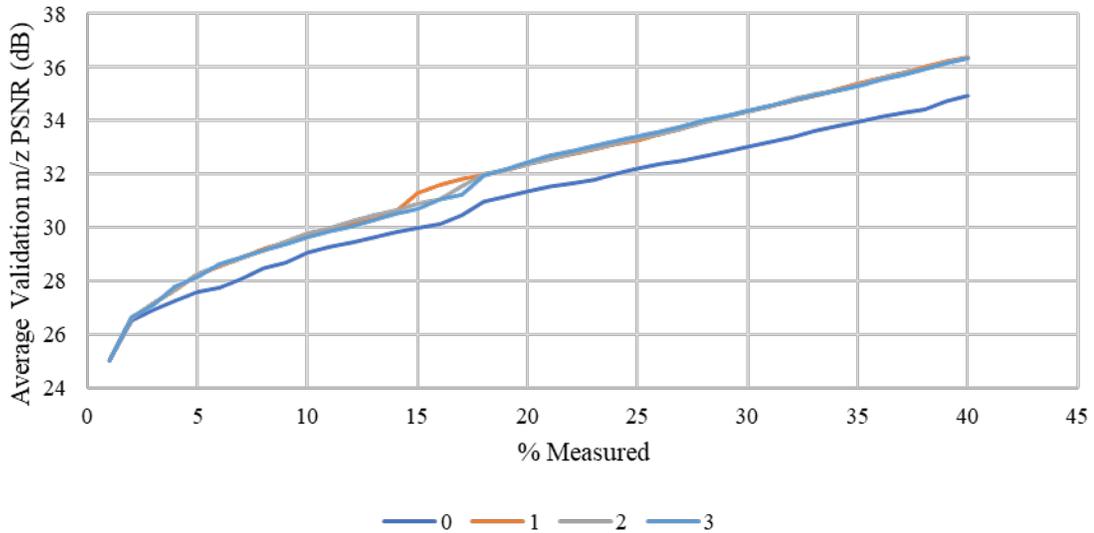


Figure 26: Progressive average PSNR of m/z reconstructions for the uterine validation set using SLADS-LS, referencing labels in Table 7

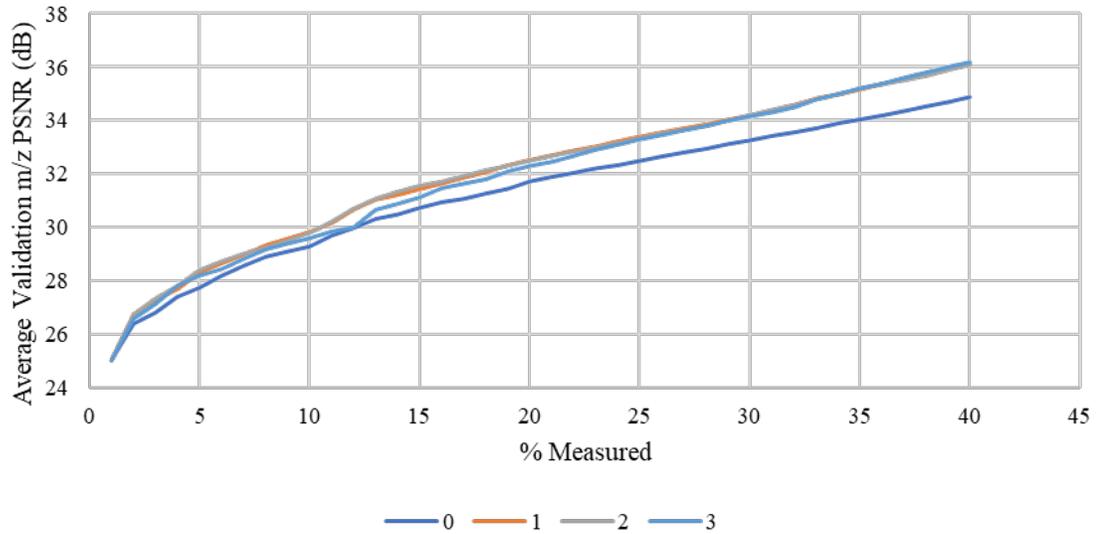


Figure 27: Progressive average PSNR of m/z reconstructions for the uterine validation set using SLADS-Net, referencing labels in Table 7

4.3 Simulations

Final simulations were performed with the testing dataset (Fig. 28) with DLADS (reconstruction and measured values as inputs, MAE loss, $1e-4$ LR, and no augmentation), the legacy SLADS-LS, and SLADS-LS with dynamic window and summation RD. Linewise acquisition was additionally simulated for the DLADS model. The average m/z reconstruction results are shown in Tables 8 and 9, progression of PSNR visualized in Figures 29 and 30, and progressive PSNR between the ERD and RD presented in Figures 31 and 32 for pointwise and linewise acquisition modes respectively. Visualizations of the measured masks and reconstructions for the testing samples are shown in Figures 33 and 34 with averaged PSNR of the m/z reconstructions presented in Table 10. The optimized SLADS-LS model was the fastest to reach 33 dB PSNR across the m/z reconstructions with DLADS producing the highest final PSNR at ~40%. The DLADS model was also the most successful at producing ERD similar to the RD, outperforming SLADS even when using a linewise acquisition mode.

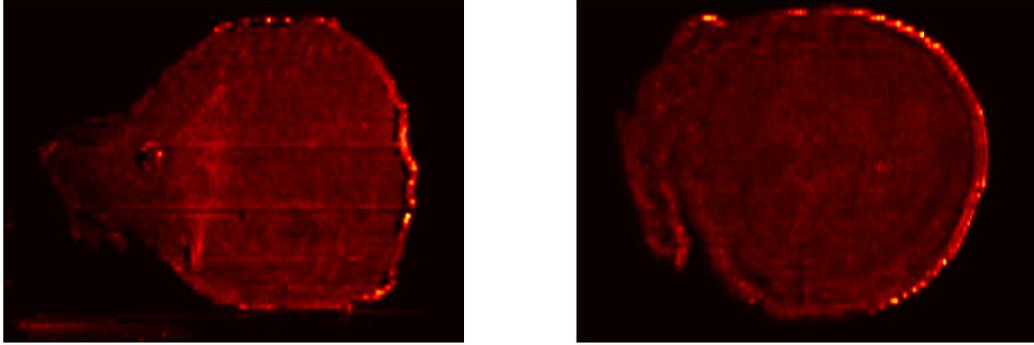


Figure 28: Ground-truth averaged m/z images (normalized by monoisotopic m/z), for the uterine testing samples Slide1-Wnt3 (left) and Slide6-V2-2 (right)

Table 8: Results for simulation of the uterine testing set for SLADS and DLADS models with pointwise acquisition

Ref.	Model	m/z	RD	Window	Avg. % to 33 dB PSNR	Final Avg. PSNR (dB) of m/z Recons.
0	SLADS-LS	1	Original	Static	13.860	37.952
1	SLADS-LS	11	Summation	Dynamic	11.032	38.453
2	DLADS	11	Summation	Dynamic	11.516	38.991

Table 9: Results for simulation of the uterine testing set for DLADS with linewise acquisition, 11 m/z, summation RD, and use of dynamic windows

Ref.	Mode	m/z	Avg. Final %	Avg. % to 33 dB PSNR	Final Avg. PSNR (dB) of m/z Recons.
0	Percent	11	35.453	32.385	33.804
1	Segment	11	69.640	51.694	40.466

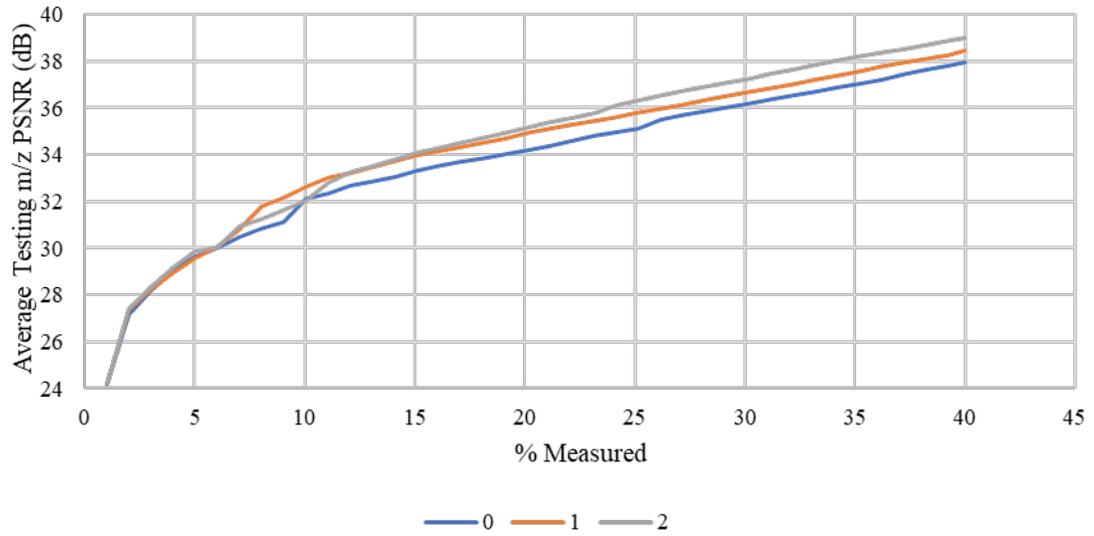


Figure 29: Progressive average PSNR of m/z reconstructions for the uterine testing set acquired with pointwise acquisition, referencing labels in Table 8

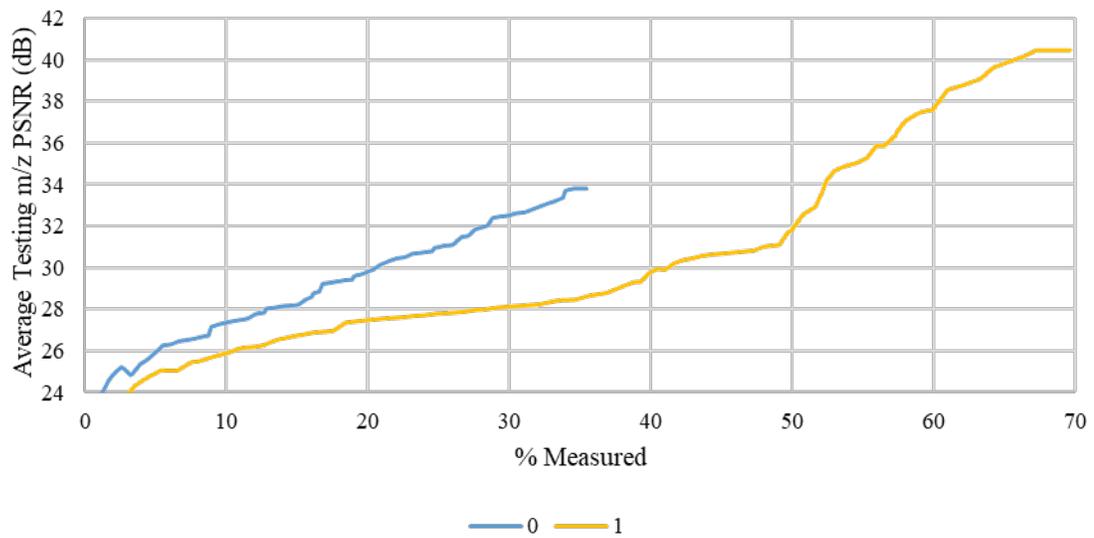


Figure 30: Progressive average PSNR of m/z reconstructions for the uterine testing set acquired with linewise acquisition, referencing labels in Table 9

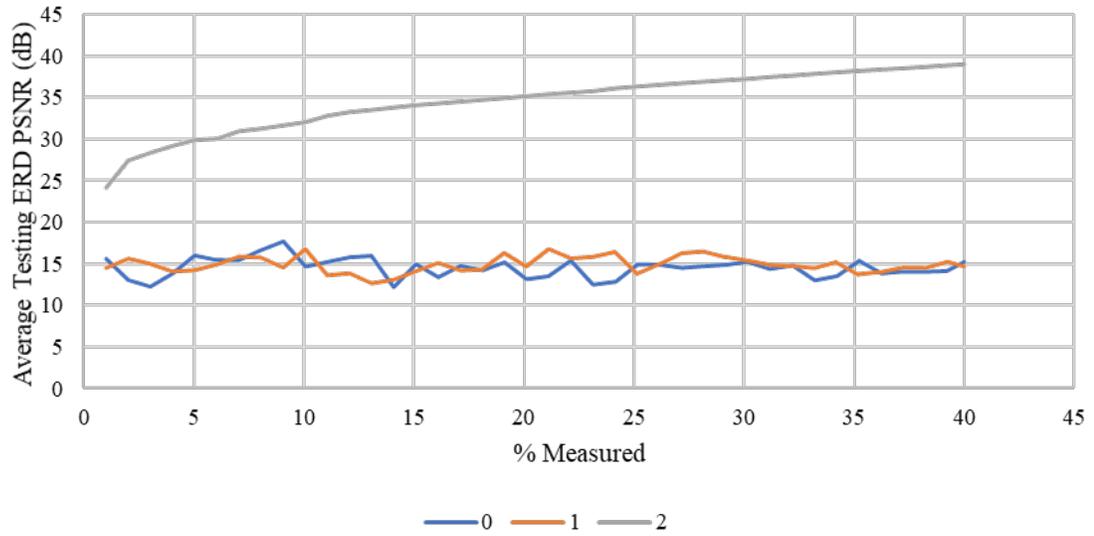


Figure 31: Progressive average PSNR between ERD and RD for the uterine testing set acquired with pointwise acquisition, referencing labels in Table 8

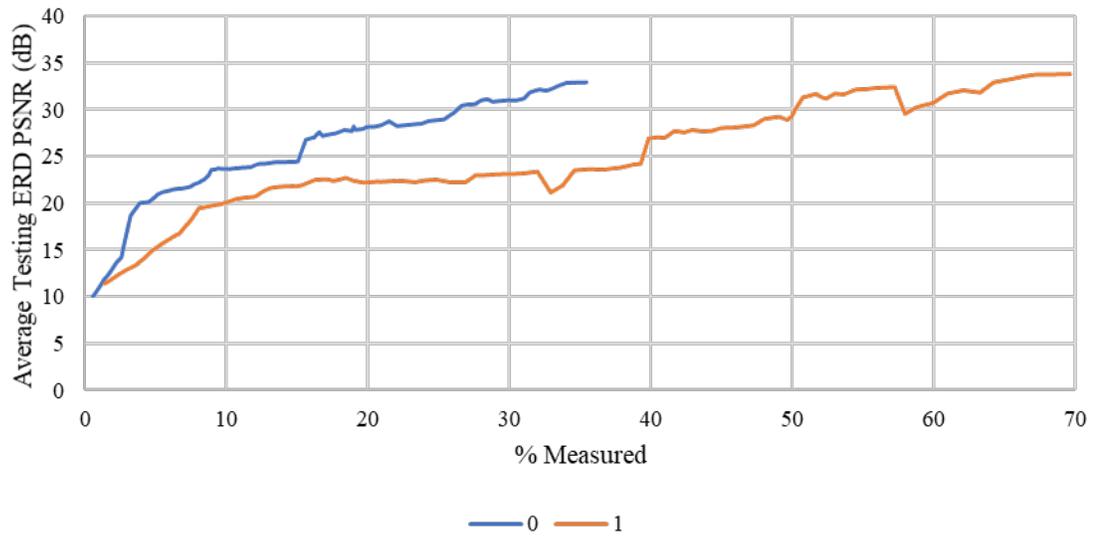


Figure 32: Progressive average PSNR between ERD and RD for the uterine testing set acquired with linewise acquisition, referencing labels in Table 9

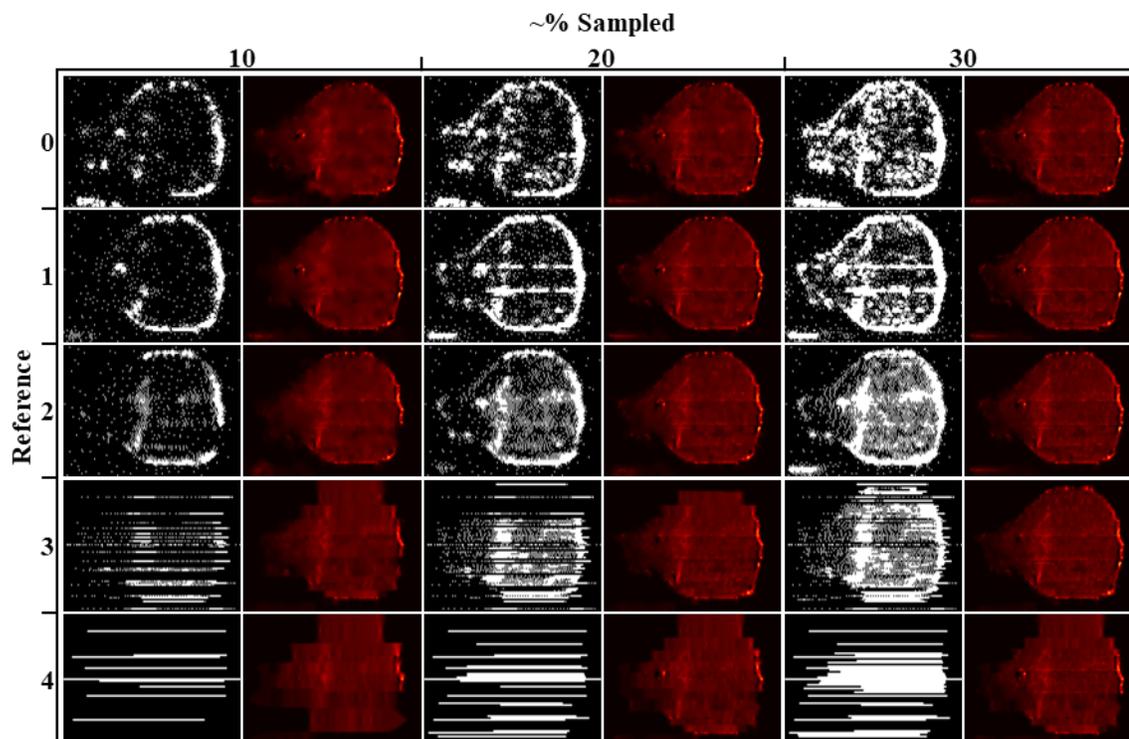


Figure 33: Progressive measurement masks and associated averaged m/z reconstructions for the uterine testing sample Slide1-Wnt-3

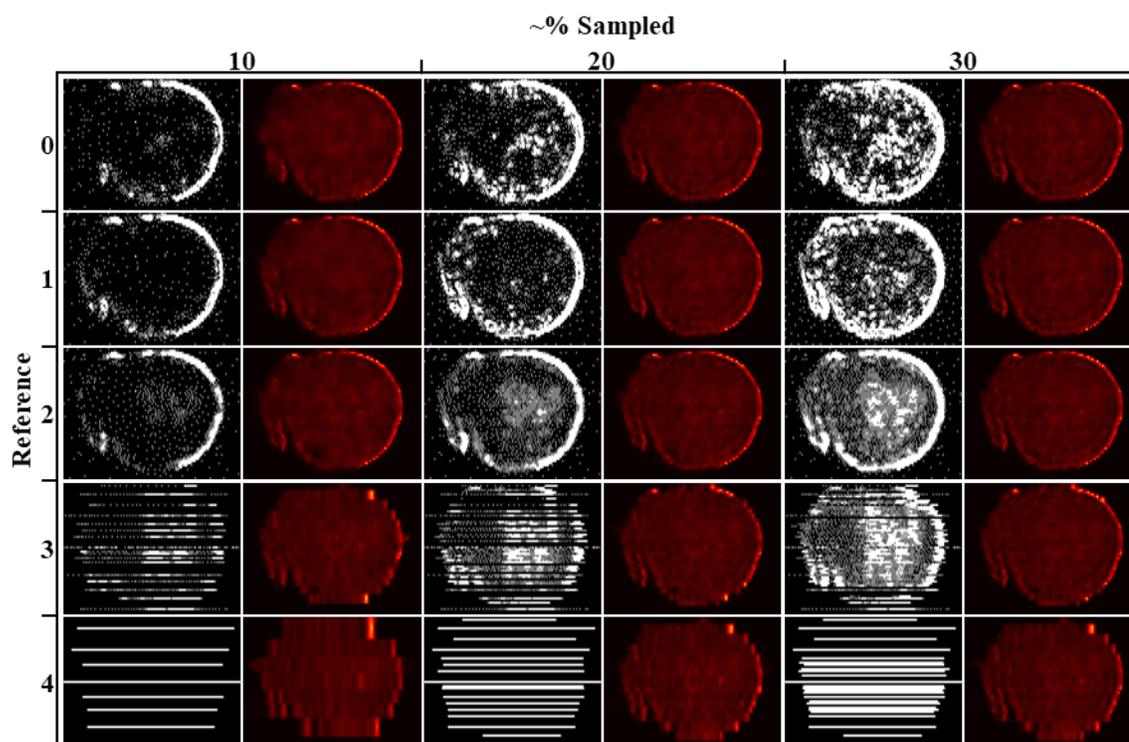


Figure 34: Progressive measurement masks and associated averaged m/z reconstructions for the uterine testing sample Slide6-V2-2

Table 10: Progressive PSNR of m/z reconstructions at varying percentages for uterine testing sample Slide1-Wnt3

Ref.	Model	Mode	m/z	RD	Window	PSNR (dB) of m/z Reconstructions at ~% Measured		
						10	20	30
Sample	Slide1-Wnt3							
0	SLADS-LS	Pointwise	1	Original	Static	31.58	33.52	35.75
1	SLADS-LS	Pointwise	11	Summation	Dynamic	32.47	34.80	36.59
2	DLADS	Pointwise	11	Summation	Dynamic	31.11	34.70	36.90
3	DLADS	Percent-Linewise	11	Summation	Dynamic	28.33	30.90	33.82
4	DLADS	Segment-Linewise	11	Summation	Dynamic	27.05	28.42	29.23
Sample	Slide6-V2-2							
0	SLADS-LS	Pointwise	1	Original	Static	32.65	34.85	36.61
1	SLADS-LS	Pointwise	11	Summation	Dynamic	32.78	35.07	36.77
2	DLADS	Pointwise	11	Summation	Dynamic	32.96	35.61	37.56
3	DLADS	Percent-Linewise	11	Summation	Dynamic	26.45	29.21	31.40
	DLADS	Segment-Linewise	11	Summation	Dynamic	24.95	26.63	27.07

CHAPTER 5 CONCLUSIONS

5.1 Discussion

The SLADS methodology has been successfully modified for simulation and integration with nano-DESI MSI data and equipment. These modifications include the dynamic and automatic determination of effective window sizes for approximated ground-truth RD generation, compensation for DESI's characteristic asymmetrical coordinate system and unpredictable sampling rate, creation of line-bounded acquisition modes, consideration of multiple molecular channels (both in model architecture and RD generation), as well as improving the employed machine learning architectures. This work has produced updated forms of the SLADS-LS and SLADS-Net implementations and a CNN neural network architecture, more specifically a modified U-Net, culminating in formation of the DLADS algorithm. A primary limitation of prior SLADS implementations, the lack of ability to incorporate a third-dimension [23] has now been addressed, with evidenced advantages to doing so.

The updated, multichannel SLADS-LS and SLADS-Net models are demonstrated to outperform their single-channel predecessors respectively by averages of 34.467% and 21.457% in reaching a 33 dB PSNR target during pointwise testing. DLADS advances on the best SLADS-LS model by an average 0.538 dB and on the legacy SLADS-LS model by 1.039 dB at ~40% measured FOV. DLADS further had consistently higher average PSNR of m/z image reconstructions beyond ~12% measured FOV compared with the best SLADS-LS model. While the simulation performance between multichannel SLADS and DLADS models may be viewed as relatively insignificant, the ERD produced by

DLADS during testing was noteworthy with an average 19.686 dB PSNR above that produced for the best SLADS-LS network. All pointwise acquisitions demonstrated reductions in the number of required measurements for a reasonable estimation of the ground-truth m/z images by ~85-90%. The percent-linewise and segment-linewise acquisition methods were able to reach the 33 dB threshold, with decreases in the number of required measurements between ~50-70%.

There was sometimes a slight advantage shown to using the new automated dynamic window method to produce approximated RD ground-truth images. During simulation of the validation set, there was a decrease in the number of epochs needed to reach an average 33 dB PSNR in m/z image reconstructions by 0.347% for SLADS-LS, but an increase of 0.988% for SLADS-Net. This could be further improved in the future by increasing the multiple of σ , up from 3, when calculating a window size to employ. There was a consistent improvement to reach the 33 dB mark in simulation of the validation set when using the summation RD generation, decreasing 2.349% for SLADS-LS and 1.791% for SLADS-Net. These figures are also underrepresented in this case, since the scenario presented in this work, where the chosen m/z are all representative of the same geometry, may be considered uncommon. Alternate combinations of non-complimentary m/z, where the resultant visualizations may obfuscate geometrical features during averaging, would additionally suffer in the original RD generation. For multiple channels, the RD should be descriptive across all of them, as done in the summation approach.

One of the primary criticisms of the SLADS methodology [22] is the tendency to oversample structural edges. This clustering behavior can be clearly seen in Figure 29.

SLADS-LS and SLADS-Net both rely on extracting representative feature vectors from a reconstructed image and only consider unmeasured locations in isolation from each other. The DLADS architecture's use of convolutional layers to learn, not only values for specific locations, but also local and global spatial relationships, leads to a reduction in measurement clustering and better coverage of the sample foreground.

5.2 Future Work

Nano-DESI MSI presented a number of difficulties peculiar to DESI scanning, which will prevent it fully benefiting from SLADS and DLADS potential performance gains. Most notably was the arbitrary limitation by equipment vendors to scanning line-by-line in a static raster pattern. While the nano-DESI research being conducted at Purdue has indicated that selection of a start and end point (segmented-linewise) for each line can be implemented, this work has shown a clear reduction in performance relative to the percent-linewise acquisition mode, allowing singular locations on a chosen line, and even more so relative to pointwise operation. Even in simulation, there were additional and continual challenges inherent to the provided data, particularly inconsistencies in which and how many positions were scanned on each line, the non-square asymmetric coordinate system, and proprietary MSI file format. While there remains, a clear advantage shown in simulations to its integration with nano-DESI workflows, there may be more benefit when combined with alternative forms of MSI. In particular MALDI MSI acquires data point-by-point and can be generally seen to have more flexibility in terms of modification to its operation, making it a prime target for future development. The allowance for 3D data and improved performance despite low availability of data, could also easily extend this work beyond just MSI technologies.

The ability to verify hardware integration was constrained by collaborator access to hardware and their confidence regarding algorithm functionality for the proposed application. While it was strongly desired that actual verification, rather than simply simulation, of compatibility with hardware be completed in time for this publication, this did not come to fruition. However, this goal remains a primary objective for this research.

This study limited itself from the employment of more advanced, adversarial networks in the interest of demonstrating applicability to integration with nano-DESI MSI and consideration for development beyond the original SLADS-LS and SLADS-Net publications. However, the relative performance gains of Generative Adversarial Networks (GANs), particularly for image-to-image translation tasks, as demonstrated by CycleGAN, Pix2Pix, and StyleGAN networks [32, 33, 34] shows great promise for integration with the SLADS and DLADS methodologies.

Another major limitation in this study has been the availability of high-resolution nano-DESI MSI data. Since the technology has only recently been capable of performing acquisitions with spatial fidelity at $<10 \mu\text{m}$, there exists very little data to use in training machine learning models in general, let alone deep learning networks. Although it was strongly desired to perform a cross-tissue study, there was insufficient overlap between the available tissue types, in terms of m/z commonly representative of the tissues' foregrounds. Should domain specific knowledge of molecules common to both become available, then such a study would be straightforward to perform. Further, this possibility will become more realizable, as greater quantities of high-resolution DESI data become available, allowing for further enhancement of SLADS and DLADS.

A more practical examination of RD generation should be should be considered, more specifically when domain specific knowledge of potentially overlapping, yet geometrically diverse m/z becomes available. It remains hypothesized that there exist scenarios where the averaging procedure, used in the original RD generation procedure, will obfuscate relevant features and impair the sampling process, where the implemented summation method can function regardless.

5.3 Concluding Remarks

The SLADS methodology has been shown in prior work to significantly reduce the number of required measurements for X-ray crystalline scanning, Raman microscopy, EDS, EBSD, and SEM technologies. This research has advanced prior machine learning models developed with this approach to perform dynamic sparse sampling with consideration for a third dimension, which will enable integration with other imaging modalities in future. Further, methods were engineered to compensate for the characteristics and limitations of present-day nano-DESI MSI technology. A more advanced machine learning based CNN was optimized in the form of the DLADS algorithm, leading to improved predictions on the order of 20 dB PSNR, simulated reductions in required measurements of ~85-90% for pointwise and between ~50-70% for line-by-line acquisitions of uterine tissue samples acquired with nano-DESI MSI.

BIBLIOGRAPHY

- [1] R. Yin, K. Burnum-Johnson, X. Sun, *et al.*, “High spatial resolution imaging of biological tissues using nanospray desorption electrospray ionization mass spectrometry,” *Nature Protocols* **14**, 3445–3470 (2019).
- [2] “The human body at cellular resolution: the NIH human biomolecular atlas program,” *Nature* **574**, 187–192 (2019).
- [3] E. Gemperline, B. Chen, and L. Li, “Challenges and recent advances in mass spectrometric imaging of neurotransmitters,” *Bioanalysis* **6**, 525–540 (2014).
- [4] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, Upper Saddle River, New Jersey (2010).
- [5] T. H. Cormen, C. E. Leiserson, R. L. Rivest, *et al.*, *Introduction to Algorithms*, MIT Press, Cambridge, Mass (2009).
- [6] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*, The MIT Press, Cambridge, Massachusetts (2018).
- [7] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” (2015).
- [8] K. Hujsak, B. Myers, E. Roth, *et al.*, “Suppressing electron exposure artifacts: An electron scanning paradigm with bayesian machine learning,” *Microscopy and Microanalysis* **22**(4), 778—788 (2016).
- [9] H. Anderson, J. Ilic-Helms, B. Rohrer, *et al.*, “Sparse imaging for fast electron microscopy,” in *Computational Imaging XI*, C. Bouman, I. Pollak, and P. Wolfe, Eds., **8657**, 94–105, International Society for Optics and Photonics, SPIE (2013).
- [10] R. Ohbuchi and M. Aono, “Quasi-Monte Carlo rendering with adaptive sampling,” (1996).
- [11] Z. Wang and G. Arce, “Variable density compressed image sampling,” *IEEE Transactions on Image Processing* **19**, 264–270 (2010).
- [12] T. Merryman and J. Kovacevic, “An adaptive multirate algorithm for acquisition of fluorescence microscopy data sets,” *IEEE Transactions on Image Processing* **14**, 1246–1253 (2005).
- [13] C. Jackson, R. F. Murphy, and J. Kovacevic, “Intelligent acquisition and learning of fluorescence microscope data models,” *IEEE Transactions on Image Processing* **18**, 2071–2084 (2009).

- [14] G. Godaliyadda, D. Ye, M. Uchic, *et al.*, “A supervised learning approach for dynamic sampling,” *Electronic Imaging* **2016**(19), 1–8 (2016).
- [15] G. Godaliyadda, G. Buzzard, and C. Bouman, “A model-based framework for fast dynamic image sampling,” in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1822–1826 (2014).
- [16] N. Scarborough, G. Godaliyadda, D. Ye, *et al.*, “Dynamic x-ray diffraction sampling for protein crystal positioning,” *Journal of Synchrotron Radiation* **24**, 188–195 (2017).
- [17] S. Zhang, Z. Song, G. Godaliyadda, *et al.*, “Dynamic sparse sampling for confocal raman microscopy,” *Analytical Chemistry* **90**, 4461–4469 (2018).
- [18] G. Godaliyadda, D. Ye, M. Uchic, *et al.*, “A framework for dynamic image sampling based on supervised learning,” *IEEE Transactions on Computational Imaging* **4**, 1–16 (2018).
- [19] Y. Zhang, G. Godaliyadda, N. Ferrier, *et al.*, “Reduced electron exposure for energy-dispersive spectroscopy using dynamic sampling,” *Ultramicroscopy* **184**, 90 – 97 (2018).
- [20] Y. Zhang, G. Godaliyadda, N. Ferrier, *et al.*, “Slads-net: Supervised learning approach for dynamic sampling using deep neural networks,” (2018).
- [21] Y. Zhang, X. Huang, N. Ferrier, *et al.*, “U-slads: Unsupervised learning approach for dynamic dendrite sampling,” (2018).
- [22] S. Grosche, M. Koller, J. Seiler, *et al.*, “Dynamic image sampling using a novel variance based probability mass function,” *IEEE Transactions on Computational Imaging* **6**, 1440–1450 (2020).
- [23] H. M. Cartwright, Ed., *Machine Learning in Chemistry*, Theoretical and Computational Chemistry Series, The Royal Society of Chemistry (2020).
- [24] D. Helminiak, H. Hu, J. Laskin, *et al.*, “Deep learning approach for dynamic sparse sampling for high-throughput mass spectrometry imaging,” *Electronic Imaging* (2021).
- [25] Thermo Fisher Scientific Inc., *Thermo FreeStyle User Guide Software Version 1.4*.
- [26] Z.-F. ei Yuan, C. Liu, H.-P. Wang, *et al.*, “pParse: A method for accurate determination of monoisotopic peaks in high-resolution mass spectra,” *PROTEOMICS* **12**, 226–235 (2011).

- [27] S. Carr, A. Burlingame, and M. Baldwin, *Mass Spectrometry in Biology Medicine*, Humana Press, Totowa, NJ (2000).
- [28] W. M. Alexander, S. B. Ficarro, G. Adelmant, *et al.*, “multiplierz v2.0: A python-based ecosystem for shared access and analysis of native mass spectrometry data,” *PROTEOMICS* **17**, 1700091 (2017).
- [29] G. Mandlbürger, M. Kolle, H. Nubel, *et al.*, “BathyNet: A deep neural network for water depth mapping from multispectral aerial images,” *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science* **89**, 71–89 (2021).
- [30] A. Odena, V. Dumoulin, and C. Olah, “Deconvolution and checkerboard artifacts,” *Distill* **1** (2016).
- [31] S. Van der Walt, J. L. Schonberger, J. Nunez-Iglesias, *et al.*, “scikit-image: image processing in python,” *PeerJ* **2**, e453 (2014).
- [32] J.-Y. Zhu, T. Park, P. Isola, *et al.*, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” (2020).
- [33] P. Isola, J.-Y. Zhu, T. Zhou, *et al.*, “Image-to-image translation with conditional adversarial networks,” (2018).
- [34] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” (2019)