REDUCING POSITIONAL UNCERTAINTY IN FTIR-TOMOGRAPHY DATA ACQUISITION WITH AUTOMATION AND MACHINE LEARNING

by

Sugato Ray

A Thesis Submitted in

Partial Fulfilment of the

Requirements for the Degree of

Master of Science

in Physics

at

The University of Wisconsin-Milwaukee

August 2023

ABSTRACT

REDUCING POSITIONAL UNCERTAINTY IN FTIR-TOMOGRAPHY DATA ACQUISITION WITH AUTOMATION AND MACHINE LEARNING

by

Sugato Ray

The University of Wisconsin-Milwaukee, 2023
Under the Supervision of Professor Carol Hirschmugl

Fourier Transform Infrared Micro-Spectroscopy (FTIR-MS) is an advanced analytical method employed to examine materials' molecular structure and chemical composition at a microscopic level. By merging the principles of infrared spectroscopy and microscopy, it offers comprehensive insights into the molecular vibrations within a sample. The key components are: (i) infrared spectroscopy which involves analyzing the interaction of infrared light with molecules, (ii) Fourier Transform allows the conversion of a time-domain signal into its frequency-domain representation. In FTIR, an interferogram is obtained, which contains information about the different frequencies present. In summary, FTIR-MS considers infrared spectroscopy to enable the analysis of molecular vibrations and chemical composition at a microscopic level. Whereas tomography is a method of imaging that generates precise, cross-sectional images or slices of an object or sample. By reconstructing collected data, it enables the visualization of internal structures and characteristics of the object. During tomography, the object is measured or imaged from multiple angles, capturing various projections. This study focuses on developing the capability to accurately capture imaging data with FTIR-MS meant for

downstream 3D reconstruction with CT. The focus of this study is on measurement of the data and not on the CT-based reconstruction of the acquired FTIR-MS data. The aim is to measure dataset for downstream CT or Limited Angle Computed Tomography (LACT) using FTIR-MS by correcting the positioning problem primarily in the following three steps – **(i)** lab automation using real-time positioning correction method, **(ii)** using feature engineering through the use of Principal Component Analysis (PCA) to find the "edge" of the sample and the background, and **(iii)** microloop's center detection using two methodologies. A Heuristic Solution is adopted to detect the center by exploiting the microloop's geometry and the other solution involved image segmentation on synthetic data using Deep Learning (DL) based solution to identify the microloop's center. To approach a DL based solution, synthetic dataset was generated using variations in microloop structure, sample location and sample shape/size. The UNet based DL model was only trained on synthetic data and had no exposure to the original IR Tomo Data measured in the lab.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I am deeply grateful for the unwavering support and assistance provided by numerous exceptional individuals throughout this journey. It is with heartfelt appreciation that I extend my sincerest gratitude to my esteemed advisor, Professor Carol J. Hirschmugl. Her unwavering encouragement, guidance, patience, and immense reservoir of knowledge have not only provided me with a sense of solace but also steered me through challenging and uncertain times, enabling me to evolve personally and professionally. Their presence has been invaluable.

I would also like to express my gratitude to the distinguished members of my committee, Professors Valerică Raicu, Jolien Creighton, Ionel Popa and Ahmad Hosseinizadeh, whose support and insightful suggestions have proven immensely beneficial. I am truly thankful for their willingness to dedicate their valuable time whenever I sought their guidance.

Furthermore, I would like to acknowledge the contributions of all the professors at the Department of Physics whose exceptional teaching has been instrumental in my personal growth. A special note of gratitude goes to Kate Valerius, whose unwavering assistance and motivation have been pivotal in keeping me focused and determined. Working alongside my colleagues at Hirschmugl Lab has been a truly enriching experience, and I am deeply appreciative of their collaborative efforts and camaraderie. To each of them, I extend my sincere thanks.

Lastly, I am profoundly fortunate to have received unwavering moral support from my cherished friends and family. Their constant belief in me has been a wellspring of inspiration, and I owe them an immense debt of gratitude.

It is the collective efforts and unwavering support of these remarkable individuals that have paved the way for the completion of this MS thesis. I am forever grateful for their contributions to my academic and personal growth.

1. Introduction

Atomic state transitions are features of visible light; however, molecular vibrations are associated with infrared (IR) radiation. In recent years, IR has been widely used in spectroscopic techniques. One of the ways of studying the molecular vibrations in IR is by considering the interactions across the wavelengths. Wilbur Kaye in the 1950s pioneered the use of IR spectroscopy when the near-infrared spectrum was tested. Later in the 1960's, Karl Norris popularized the use of IR spectroscopy in the analytical world [1]. However, the discovery of Fellgett's advantage (measurements where there exist an increased signal-to-noise ratios (SNR)) in early 1970's and Fourier transformation made utilization of IR spectroscopy possible [2]. Since then, Fourier-transform infrared (FTIR) spectroscopy has been an instrumental tool for IR spectroscopy [3], [4], [5] and IR spectroscopy is a crucial tool for studying the structure of a variety of objects.

In 1972, Godfrey Hounsfield and Allan Cormack invented the Computed Tomography (CT) scanner. CT is used for non-invasive 3D reconstruction of an object of interest. Using a combination of computer technology and X-rays, CT produces images of the inside of an object (often, parts of human body). This study focuses on developing the capability to accurately capture imaging data with FTIR-MS meant for downstream 3D reconstruction with CT. The aim is to measure dataset for CT or Limited Angle Computed Tomography (LACT) using FTIR-MS. A custom-made stage for tomography was necessary for precise and reproducible sample rotation and translation under the FTIR microscope and within a tight space of a cube of about 3.5 cm sides. The design of the stage (henceforth referred to as TomoStage) required 6 degrees of freedom (dof) with 3 translational and 3 rotational axes. These 6 dofs were necessary to allow for the complex set of motion necessary to measure data for both CT and LACT experiments.

Hirschmugl lab had previously published a study of generating 3D reconstruction of biological sample(s) from more coarse measurements of tomography data, limited by the precision of the gantry stage (made with parts from National Instrument) used for sample rotation [6]. This provided a foundation for developing FTIR-MS based CT reconstruction with more precise capabilities of sample rotation.

The internal-chemistry of a sample can be inferred from Fourier Transform Infrared (FTIR) micro-spectroscopy (MS) imaging, which help in biological studies such as diabatic retinopathy (DR) [7], [8].

It is well known that Hyper Spectral Imaging (HSI) provides a suitable amalgamation of both microscopy and spectroscopy for biological studies [9]. One such HSI technique, FTIR-MS can be used to study the 2D-spatial distribution of chemistry on a sample [10] by evaluating the refractive index (RI) distribution inside the sample [11]. The spectra for a thin film of homogenous RI distribution inside will look much different from that of a spherical object with the same homogenous RI owing to the EM normal modes inside the object [11]. Add to the mix, the effects of optics and the spectra could look further different owing to the Optical Transfer Function (**OTF**) of the microscope. The measured FTIR spectra of an object often consists of contributions from absorption, reflection and scattering [12], [10], [13], [14], [15]. Elastic scattering resulting from the spatial inhomogeneity introduces broad undulating background to spectra [16]. Scattering correction has been implemented for IR spectra to extract biologically relevant information [13], [14], [15]. When combined, FTIR micro-spectroscopy and Tomography could yield a non-destructive method to study the volumetric chemical composition within the sample [6]. However, Computed Tomography (**CT**) assumes minimal to no scattering for each of the 2D projection

images and a linear ray-path. Diffraction Tomography (**DT**) considers bending of light inside the sample for constructing the internal RI distribution [17].

The contribution of this research work is three-fold. First, a custom-made retractable platform (RP) was developed so that the equipment (TomoStage) can be mounted on and off the FTIR microscope as and when necessary, allowing for measurements with other kinds of sample mounting stage possible without imposing any operational constraints. There was an additional requirement of the RP to allow for smooth operation with minimal to zero maintenance over a prolonged period (spanning over years), despite accumulation of dust and dirt/grease. The design of the RP, procurement of parts and tooling was done in the Physics Department machining shop on campus. Once the RP was available, the first measurement with the TomoStage revealed that any sample mounted on the stage was observed to effectively move outside the field-of-view (FoV) and that not all the 3 rotational axes could be used for sample rotational purposes due to lack of precision or reproducibility.

The Hirschmugl Lab had a script (using Brucker's proprietary scripting technology) to operate the Bruker Hyperion 3000 microscope that allowed them to capture tomography and other datasets (TomoData) in an automated fashion. However, a system upgrade to the Bruker microscope by the manufacturer rendered the previous automation scripts unworkable. This effectively curbed all automated measurement capabilities that could be reliably performed. Thus, we created an in-house automation of all the lab-equipment using AutoIt to complete Lab Automation. AutoIt is a test automation framework which allows one to quickly prototype automation by using Graphical User Interface (GUI) or Command Line Interface (CLI) automation. After rigorous efforts to control various equipment, including the microscope, light-sources, network-attached-storage, cryogenic-LN2-pump, old tomography stage, etc., a lab automation suite was developed.

Studying the three axes of rotation and understanding the error that happens in a scale of microns to millimeters for a sample that is typically one-tenth of the thickness of human hair (~100 microns) was crucial. The challenge was to understand the motion of the object by observing the object only under the microscope, while the location of the object was not known a-priori and the only means of observing the object under the microscope was by trial and error.

Additionally, since the TomoStage was under warranty from the manufacturer, we could not afford to have made any modifications inside the equipment, even if we had the necessary skillset to work with piezoelectric motors. Software automation was the only way to fix the problem– any direct hardware modification was out of scope. The ab-initio home-grown Lab Automation suite, called AutoLab consists of multiple modules, each used for automating/controlling different aspects or equipment of the lab. The module written for measuring TomoDataset provides means to autocorrect the position of the sample mounted on the TomoStage after each rotation. The solution for this was engineered so that the initial setup can be completed between 30 – 45 minutes and then the automated FTIR measurements will continue running. For the rest of the FTIR measurement (which could run for over 12 hours), the automated module keeps correcting the position of the sample and then runs FTIR imaging. For diagnostics detailed logfiles are also created, which helps in case the experiment fails for some reason.

This was first achieved with micromeshes (resembling tennis rackets) measuring 300 microns across and using a 4x objective. Once the 4x objective showed promising results with the micromeshes, we measured a microloop (50-micron diameter) using visible light.

However, when FTIR images were compared against visual light (Vis) based measurements, the Vis images seemed to move in the plane of the images, whereas the IR images did not share any such characteristics. The only possible reason behind this was the Vis Camera was probably

shaking while measurements. On the other hand, when sinograms were constructed with the IR data, we observed that the sinograms were jagged and the jaggedness only suggested the projection images being non-centered about the axis of rotation. Thus, even though the Lab Automation driven positioning correction did yield about 1000 times better correction in positioning the sample, the quality of the jagged sinogram was yet not acceptable for downstream consumption by a CT algorithm and reconstruct the object of interest non-invasively.

Second, Post Image-acquisition Positioning Restoration (PIPR: pronounced as *piper*). Since visible light images were observed to move in the plane of the image, they could not be of any use for positioning correction. The establishment of an acceptable ground truth of what exact change will set the projections align about the same axis of rotation and smoothen the jagged sinograms as a result was necessary. One observation was that given all the images are offset such that the center of the microloop in all the projections coincides, it will effectively ensure a common axis of rotation passing through the microloop's center in all projections and through the plane of the image and typically through the microloop's holder. Using Feature Engineering, specifically, Principal Component Analysis (PCA), we tested this hypothesis by manually identifying the microloop centers in each projection and noting their coordinates. Finally, coinciding the projection images' microloop centers effectively placed them on the same axis of rotation.

The next logical step was to develop an automated method of identifying the microloop centers in the projections. To start with, a heuristic method of exploiting the circular or elliptical geometry of the microloop in a projection, based on the angle of projection was applied. This approach worked in almost 90% of the projections. But had its limitation when the cross-section of the microloop hardly resembled an ellipse. So, there was an improvement needed over the heuristic method. Third, prediction of which pixels in the image belonged to the microloop and not to the

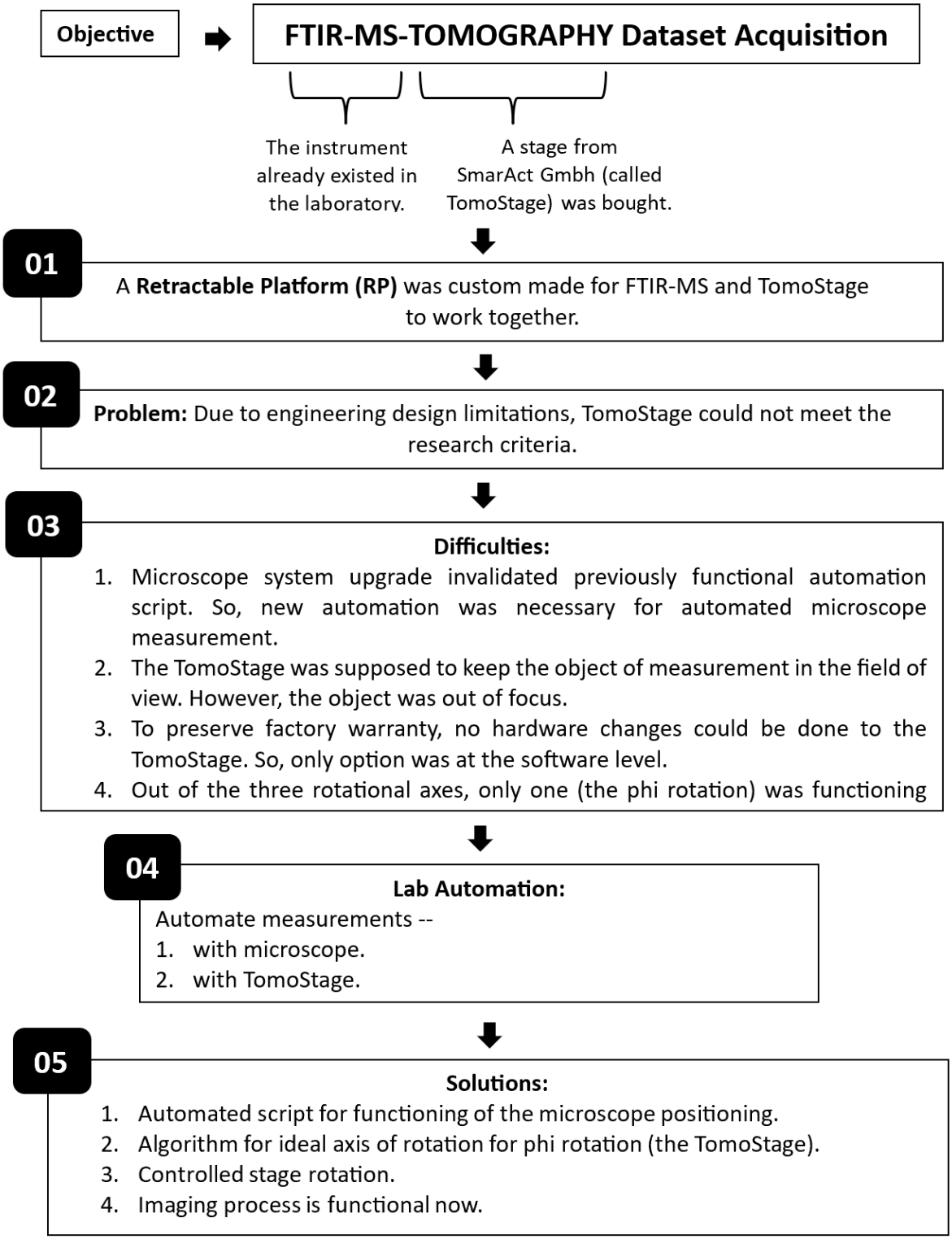
sample alone or the background. This is a classic scenario, where given an input X, one is trying to map it to another space with an output of Y. Deep Learning models typically are good at this kind of problem scenarios, given we have enough training data with a healthy proportion of unique variations in the features, so that the model can learn to generalize.

If the data to train the DL model already existed, then there was nothing to solve for as the perfect data is already available; or, if there is no data, then the DL model cannot be trained. Scenarios like this can often be circumvented by generating synthetic training data such that it closely resembles the real-world scenario. We used Computer Aided Design (CAD) to generate parametrized synthetic datasets of sample and microloop assemblies were derived. The datasets thus produced were used to train a UNet [18] based DL image segmentation model, which can predict the pixels in the image that belong to the loop portion of the microloop. Given these pixels' positions, a centroid of those positions will yield the position of the microloop's center in that projection.

To summarize, this thesis aims on developing the capability to accurately capture imaging data for tomography with FTIR-MS, correcting the positioning problem primarily in three steps – (i) lab automation assisted using real-time positioning correction, (ii), feature engineering and object image mask generation and (iii) Deep Learning based solution to identify microloop's center.

A flowchart is provided in the next two pages summarizing the salient points of this thesis.

1.1. Summary Flowchart



06

Observations:

1. The sinogram from the images had jitter.
2. The micro loop was not stable in visible light. However, it was stable under IR imaging.

07

Challenges:

1. Images often had challenges such as, diffraction, scattering, out of focus pixels.
2. Jitters in sinogram needed correction before the dataset could be used for tomographic reconstruction.

08

Ground Truth Determination:

1. A manual process of finding the center of the micro loop was adopted.
2. All points were labelled to find the micro loop center (MLC).
3. Automation to find the MLC.
4. Single tomo data projection dimension was: 128 x 128 x 770. There were 770 wavelengths for the IR measurements.

09

Feature Engineering:

1. Method of "edge" detection → **Principal Component Analysis (PCA)** was used to find the edge of the assembly (micro loop + sample).
    a. Data dimension reduced from 128 x 128 x 770 to 128 x 128.

10

Synthetic Dataset Generation for Deep Learning (DL):

1. 2D image datasets produced with parametric CAD generated synthetic 3D microloop and 3D sample.
2. Variations in microloop structure, sample location and sample shape/size were used to enrich the synthetic dataset.
3. Image augmentations were used to further enrich the training dataset for DL.
4. The UNet based DL model was only trained on synthetic data and had no exposure to the original IR Tomo Data measured in the lab.

11

Microloop Center Detection:

1. **Heuristic Solution:** Center detection by exploiting the microloop's geometry.
2. **Deep Learning (DL):** Center detection using image segmentation on synthetic data.

1.2. Why FTIR Micro-Spectro-Tomography?

Fourier Transform Infrared (FTIR) micro-spectroscopy is an emerging technique for the biochemical analysis of tissues and cellular materials [19]. FTIR spectroscopy serves as an analytical method employed for the identification of functional groups within both organic and inorganic compounds. This is achieved by assessing their absorption of infrared radiation across a spectrum of wavelengths, as outlined in studies by Smith [20] and Margaris [21].

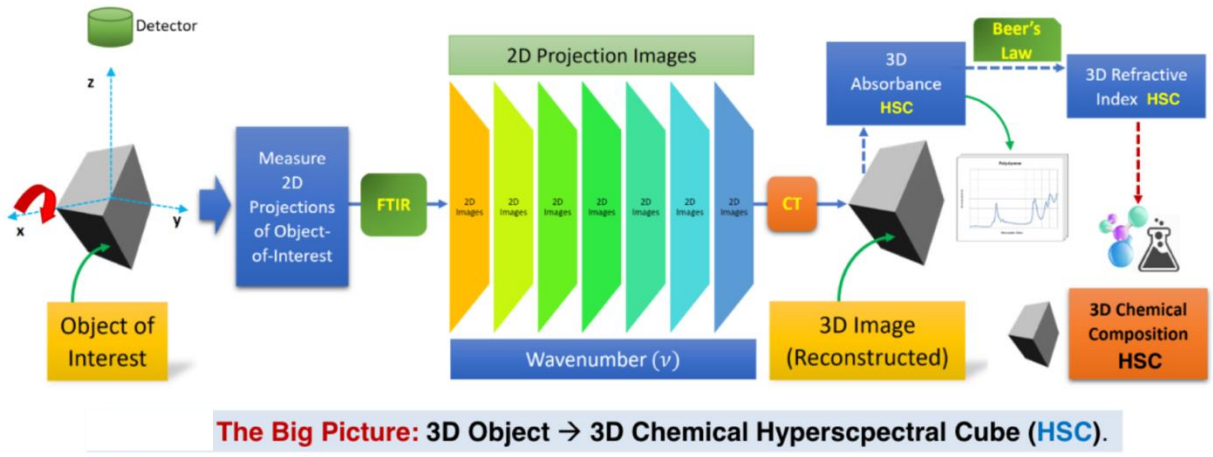


***Figure 1**: The Big Picture - Why FTIR Micro-Spectro-Tomography?*

High energy electromagnetic (EM) exposure, such as from the use of if X-rays could damage biological tissue, as the EM-wave travels through the tissue [22, p. 5]. An alternative to using such high energy EM wave, is to use relatively low energy infrared light source. The relatively low energy of IR makes it less harmful for biological tissue [23]. While this is a benefit of using IR light source to image a biological sample, there is no free lunch: since most biological cells are typically $\sim\ 1\ -\ 100\ \mu m$ [24]. Since the wavelength of the IR light source (mid-Infrared) that we

use in our experiments ranges between 2.5 microns to 20 microns, and that is also comparable to the size of the biological cells, this causes light to bend: which is also known as dispersion.

Refractive Index (RI) is typically a complex quantity, whose real part is associated with refraction and hence scattering. On the other hand, the imaginary part of RI is related to any attenuation as the EM wave travels through a medium. Thus, the imaginary part is related to the absorption in a medium [4, pp. 382-435].

The energy band of IR electro-magnetic wave excites the vibrational modes of the molecules. For FTIR to work the infrared absorption must manifest through the change in the electric dipole moment of the molecule. This is called the selection rule of infrared spectroscopy [3, p. 5]. The interactions of infrared radiation with matter may be understood in terms of changes in molecular dipoles associated with vibrations and rotations [3, p. 6]. Under the sinusoidal driving force from the EM wave, the molecules could undergo stretching or bending, which is typically known as vibration. IR Spectroscopy helps measure these vibrations and hence is also called vibrational spectroscopy [3, p. 8].

FTIR Micro-Spectro-Tomography uses multiple projections of the sample of interest (SoI), where each projection comprises of multiple images (2D data) of the SoI, measured using various infrared wavelengths. These 2D images are essentially absorbance spectra corresponding to a cross-sectional area under the microscope. A Computed Tomography (CT) algorithm then transforms (reconstructs) the 2D absorbance information into a 3D absorbance hyper spectral cube (HSC).

The Beer-Lambert law provides a relationship between the amount of absorption and the thickness of the sample [3, pp. 57-59]. It states that absorbance is directly proportional to the thickness and concentration of the sample.

$$A = \epsilon c l$$

Where, A = absorbance, c = molar concentration, l = path length of the sample (thickness), and ϵ = molar absorptivity.

The Beer-Lambert law could then be used to transform the 3D absorbance cube into a 3D complex refractive index cube. The 3D distribution of complex refractive index could help identify the type of biological material in a tissue.

As an example of how this method could be useful, say you have cancerous tissue embedded in an otherwise healthy tissue. The area where cancer has metastasized will have similar biological matter, but with higher concentration. For the same material (ϵ), and same sample thickness (l), a higher concentration should yield a higher absorption (and vice versa). However, if the type of material happens to be different, then the corresponding absorption peak will also move horizontally, in the absorbance spectra.

2. Lab Automation

Blacksmiths played an important role during the Middle Ages. They served as a major engineering source of talent for practicing and evolving critical technology, that ultimately decided if a certain kingdom had the necessary weapons or technology to establish its predominance or ward off intruders. They were forging the tools that would be used during peacetime or wartime. However, they had other set of tools to that enabled them to forge the end-product (weapons/tools) they produced for consumption by the public or the army.

Today, scientific progress rides on the advancement of various technologies and the use of software is pervasive in managing and building the technological infrastructure that in turn supports the core infrastructures. The role played over 1000 years ago by the blacksmiths are often bourn by engineers today and because software is controlling systems everywhere, software engineers also play a very important role today. For scientific laboratories, where resources are often limited and human attrition is high, the only way to scale up their capabilities could often depend on automating their processes. The reusable and scalable lab automation strategies that we have adopted could be open-sourced in future and shared with the scientific community for the greater good.

The laboratory measurements involve controlling multiple equipment, primarily including an FTIR Microscope, a tomography stage (SmarGon) and a liquid nitrogen (LN2) pump. The individual measurements could range from a few minutes to close to an hour, depending on the type of sample and the type of infrared (IR) source used. For tomography we may need 180 to 360

measurements, which could run for hours and sometime over multiple days. This clearly meant we needed automation for our lab measurements.

Significance and Benefits of Automation:

- Minimum human interference/involvement → **lower personnel cost**.
- Faster completion of tasks → **higher productivity**.
- Greater consistency → **better repeatability**.
- Structured oversight with output-log files → **more documented transparency**.
- Greater control → **technologically advanced**.

At present, we are using Globar as a broadband IR source. The imaging scheme for tomographic application involves measurements with both IR and visible light (**VIS**) for each angular-projection. The Bruker Hyperion 3000 FTIR Microscope uses a Focal Plane Array (**FPA**) detector and the operating temperature of the detector needs to be maintained at 84 K. The Norhof LN2 pump is used to inject liquid nitrogen into a tank onboard the microscope to keep the FPA at 84 K. Previously we used a National Instruments made tomography stage (***NStruct***) for tomography measurements. NStruct had manual setup for any translational motion and any subsequent setup was completely manual. It had a rotary stage mounted on top of the manual translational stages. Compared to NStruct, SmarGon is a rather accurate and advanced stage with both translational and rotational capabilities. There are three translational axes and three rotational axes onboard SmarGon. As a part of lab automation, each instrument is controlled by a dedicated module. A module is a group of custom functions written to control a specific instrument. For a list of lab-instruments refer to ***Table 2***.

Table 1: A Few of the Key Building Blocks of AutoIt for Rapid Prototyping of Workflow

Common Windows User Tasks	AutoIt Functions
• Do a mouse click (left, right)	**MouseClick**(), **ControlClick**()
• Type something in an editable space	**Send**(), **ControlSetText**()
• Cut, Copy and Paste something	**Send**(CTRL+X/C/V), **ClipGet**(), **ClipPut**(), **ControlGetText**()
• Select a menu-option with mouse or with command	**Send**(sequence of key strokes as you would to access the menu)
• Access different open windows (minimize, maximize, select, reposition/move a window)	**WinActivate**(window name), **WinMove**()

2.1. The Need for GUI Automation

All the instruments came along with their respective vendor-software and a user can operate these instruments via the graphical user interfaces (GUIs). The operating workflow for these instruments via their GUIs were already known and no other dedicated command-line interface was available. Thus, primarily any form of automation needed to happen on top of the GUIs.

2.2. The Choice of Primary Programing Language

The necessity of GUI automation in a Windows environment led to the use of a programing language called AutoIt [25]. A freeware BASIC-like scripting language designed for GUI automation and general-purpose scripting in a Windows environment, AutoIt [26], uses a combination of simulated keystrokes, mouse movements and window/control manipulation to facilitate automation in a reliable manner. **Table 1** provides a few examples of simple one-line functions for GUI automation using AutoIt.

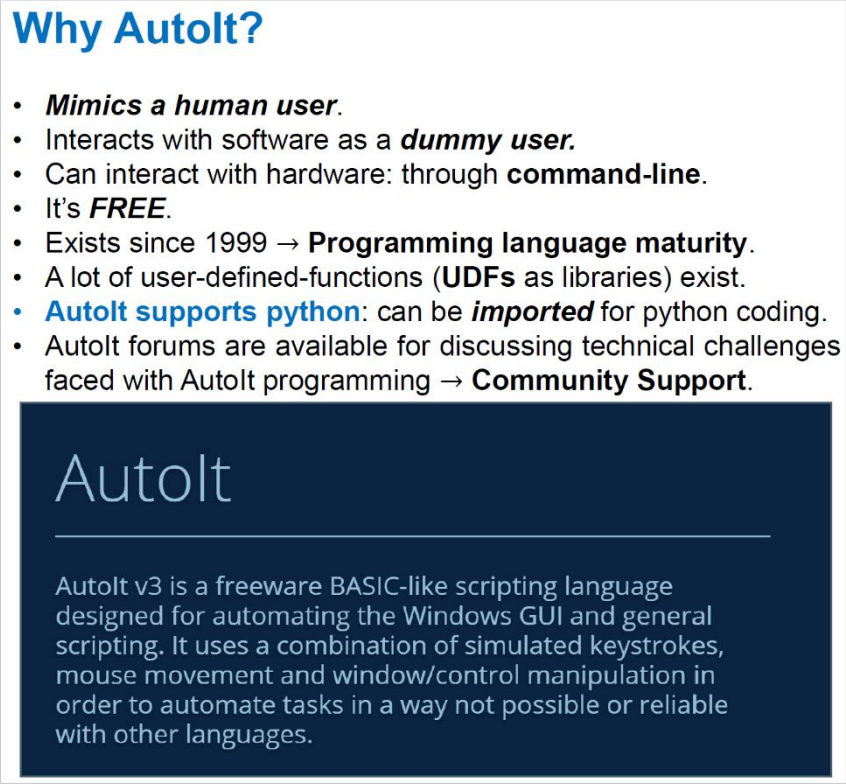


Figure 2: Why AutoIt?

A set of most important features that make AutoIt a favorable candidate for lab-automation in Windows OS.

AutoIt has been used to construct low-cost laboratory automation processes and it has been observed that it has low entry bar allowing new users of AutoIt to learn and apply it quickly [27].

2.3. Mode of Interaction Between AutoIt and Hardware

AutoIt can either directly interact with the hardware through some command line interface (CLI) or it could access the hardware via the vendor/manufacturer provided software GUI. A schematic diagram of this mechanism has been provided in **Figure 3**. The following table (***Table 2***) provides a list of the instrument/components along with their respective mode of interaction.

Table 2: Mode of Interaction Between AutoIt and Hardware

SL#	Instrument/Component	Mode of Interaction (CLI/GUI)
1	FTIR Microscope	GUI
2	IR Source	GUI
3	NStruct Tomography Stage	GUI
4	SmarGon Stage	GUI
5	Norhof LN2 Pump	GUI
6	ICCapure Visible Light Imaging Utility	GUI
7	LAN Control	CLI
8	Data Synchronization	CLI

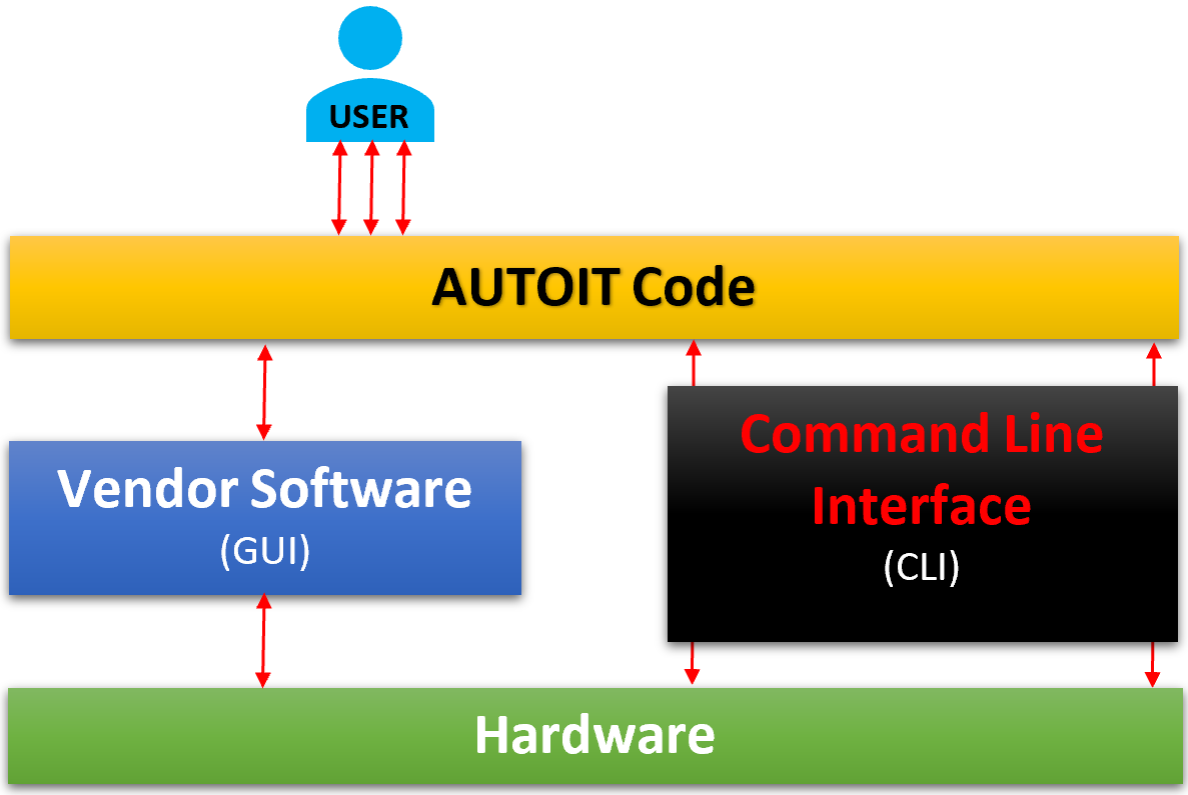


Figure 3: Lab Automation using AutoIt.

The hardware can be controlled by AutoIt either through a command line interface (CLI) or automation of a graphical user interface (GUI) provided by the vendor/manufacturer.

2.4. Modules

Several AutoIt modules were developed to support various kinds of operations. Each module provides some custom user defined functions (UDFs). The following table (**Table 3**) provides an alphabetically ordered list of the modules.

Summary: Total **16 Modules** with **8.5k lines of code** and a total of **232 functions**.

Table 3 shares a summary about the AutoIt modules created for enabling lab automation and rapid prototyping.

SL#	Module Name	File Name	Short Description	Number of Functions (NoF)	Lines of Code (LoC)
1	CMDUtils	CMDUtils.au3	Module with custom UDFs for CMD.exe.	2	64
2	DataSyncUtils	DataSyncUtils.au3	Module with for synchronizing data over Network Attached Storage (NAS).	1	151
3	FileCompressionUtils	FileCompressionUtils.au3	Module for handling compressed files (unzipping).	1	103
4	FTIRUtils	FTIRUtils.au3	Module to control .	52	1560
5	ICCaptureUtils	ICCaptureUtils.au3	Module to control visual image capture on the FTIR Microscope.	8	380
6	LabCredUtils	LabCredUtils.au3	Module to control authentication on Laboratory computer.	9	153
7	LANUtils	LANUtils.au3	Module to control Local Area Network (LAN) settings.	7	246
8	Math_LinearRegressionUtils	Math_LinearRegressionUtils.au3	Module for Linear Regression.	11	190
9	Math_TrigUtils	Math_TrigUtils.au3	Module to provide Trigonometric functions with array operations.	26	823
10	NorhofUtils	NorhofUtils.au3	Module to control NorHolf cryogenic liquid nitrogen pump.	20	890
11	NStructUtils	NStructUtils.au3	Module to control NStruct stage for tomography.	8	340
12	OPUSUtils2	OPUSUtils2.au3	Module to control imaging of infrared (IR) images on the FTIR microscope.	42	1775
13	OPUSUtils_BrukerSwitch	OPUSUtils_BrukerSwitch.au3	Module to control switching between imaging sources.	4	182
14	SmarGonUtils	SmarGonUtils.au3	Module to control SmarGon stage for tomography with advanced motion.	25	970
15	XYStageUtils	XYStageUtils.au3	Module to control the 2D motion of the detachable XY-stage that can be mounted on the FTIR microscope.	9	332
16	TomoGlobalVars	TomoGlobalVars.au3	Module to capture variables' state during an experiment and enable communication between various modules.	6	347

Table 3: List of AutoIt Modules for Lab Automation

3. Automated Tomography Measurements

With Hardware Positioning Correction

Current Operational Status:

The TomoStage (SmarGon) poses some significant challenges for rotational motion in keeping the sample steady under the microscope. We are currently calibrating the stage to algorithmically predict the position of the sample in space during rotation and bring it back under the focal point of the microscope for imaging.

3.1. Equipment Details

Equipment: SmarAct Stage (SmarGon) for Tomography/Laminography.

Manufacturer: SmarAct Gmbh.

Received: Jan-2018

3.2. Purpose of the Stage

Acquire FTIR-MS measurements for CT or LACT.

The intended use of the stage is to control translation and rotation of the sample with micron level precision for translation and milli-degree precision for rotation about certain axes. Such precise movements will enable accurate measurements in the FTIR micro-spectroscopy system to yield precision driven Tomography and future developments of Laminography.

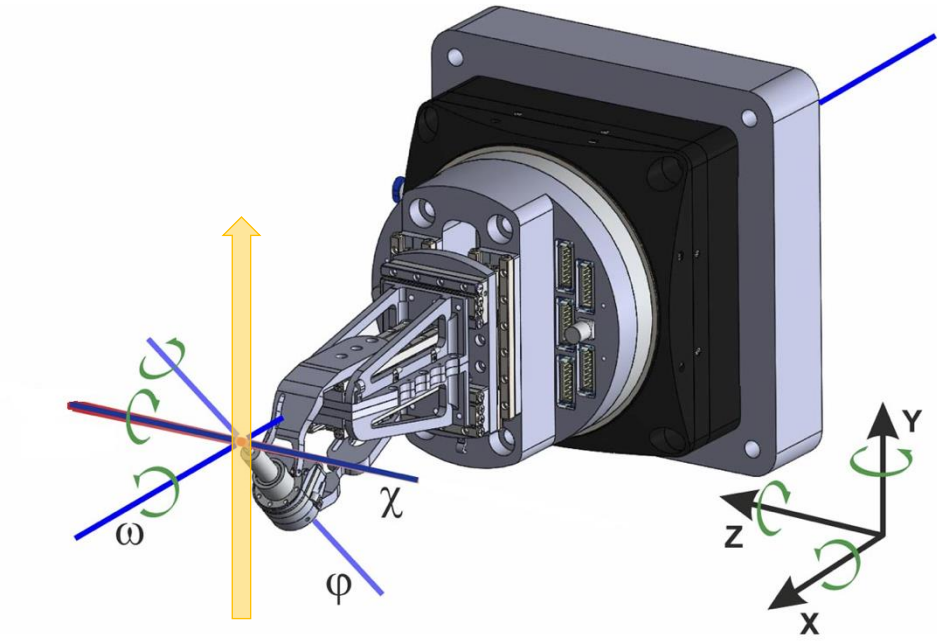


***Figure 4**: SmarGon axes for translation (x, y, z) and rotation (phi, chi, omega).*

The IR/visible light beam is directed along the y-axis. The sphere of confusion (SoC) for the ω-rotation is the least (1 μm) compare to χ-rotation (7 μm) and ϕ-rotation (10 μm). The least count ability of each of the three rotational axes is 1 milli degree. The accuracy for translational motion along all the three axes (x,y,z) is 1 nm. The yellow arrow shows the direction of IR beam.

3.3. Description of the Stage

The TomoStage (SmarGon) is equipped with piezo-controlled nanometer precision X, Y, Z motion. It also has three rotational motors with their precision ranging between 1 μm to 10 μm sphere-of-confusion:

- Omega ($\boldsymbol{\omega}$) Rotation: rotation about X-axis (most precise motion in rotational motors).

- Chi (χ) Rotation: rotation about Z-axis.
- Phi (ϕ) Rotation: rotation about an axis as shown in **Figure 4**.

3.4. Equipment Delivery Delay from Vendor/Manufacturer

The customized TomoStage (SmarGon) was ordered in Jan-2017 and according to the vendor bid, the delivery was expected by May 2017. The equipment was finally delivered in January 2018 owing to delay caused by the vendor/manufacturer.

3.5. Establishing Operational Compatibility in Lab

The stage (SmarGon) was not compatible for direct integration with the existing equipment in the lab. As it turned out, the vendor had manufactured SmarGon with certain assumptions about the operational height difference between the microscope and the optical table acting as the universal base for all lab equipment. We needed a custom-made platform which could support SmarGon and

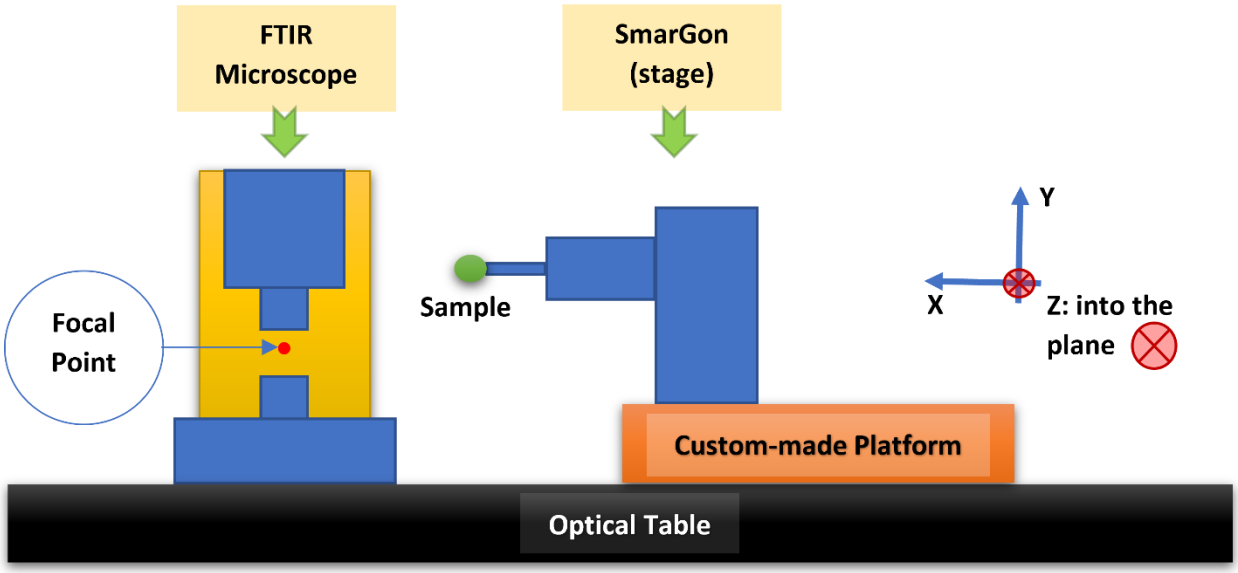


Figure 5: Schematic diagram of the microscope, the SmarGon stage and the custom-made platform

travel both in ***X*** and ***Z*** directions as shown in **Figure 5**. The custom-made platform was assembled

in the Physics Dept. machine-shop and was ready by the end of February 2018. The subsequent tests with SmarGon made it clear that using this equipment will require addressing some additional challenges.

3.6. Challenges faced with Operation of SmarGon

3.6.1. SmarGon does not have an absolute global translational reference frame

The entire assembly is mounted on a vertical platform attached to a tower. The vertical platform rotates about x-axis during omega-rotation. This makes the sample effectively translate to a different point in space under omega-rotation. Since, there is no global x, y, z translational capability available which is also uncoupled from omega-rotation, any translation using the onboard $x/y/z$ axes amount to moving in the relative frame of reference of the rotating stage.

3.6.2. Sample translates to outside Field-of-View during rotation

It was observed that both for the phi and omega rotation the sample rotates about the axis of rotation with some effective radius of rotation and hence moves outside the Field of View (FOV). The focal point is a point in 3D space under the FOV. Once the sample moves away from the focal point, it needs some compensatory translation to get back under the FOV at the focal point. The solutions for achieving this for phi and omega rotation are quite different from each other. As they come with certain benefits, rotation about phi and omega axes also present certain challenges and there are pros and cons to using one of them as opposed to the other. This will be explained in the subsequent sections.

3.7. Phi Rotation

The phi-rotation theoretically should be equivalent to the omega-rotation when there is no chi-rotation. Additionally, the phi-rotation is uncoupled from the onboard x, y, z translations. Thus, phi-rotation for no chi-rotation and no omega-rotation, effectively becomes a good test case that mimics the intended behavior of the stage for tomography. To test if rotation followed by compensatory translation calculated with algorithmic corrections could yield the desired rotation of the sample under the microscope, the sample was rotated with phi-rotation. It was observed that the sample moves through space during phi-rotation, instead of rotating about a point of interest inside the Field-of-View. We tested a proof-of-concept to address this problem:

1. Rotate about phi
2. Predict rotated position
3. Translate to the predicted position

This brought the sample back under the microscope for all the angular rotations with phi-axis. But it must also be noted that the accuracy of phi rotation is inferior to omega and chi rotation. We intend to use omega and chi separately, for Laminography and preferably omega alone for Tomography. This led us to our next logical step: we started implementing the same tactic for omega-rotation.

3.7.1. Key lessons from studying Phi-rotation

1. Phi-axis with Omega=0 and Chi=0.
2. Rotation axis is same as x-axis.
3. When sample is mounted using MiTeGen [28] mount, an arbitrary radius vector exists from rotation axis to the point of observation.

4. Rotation effectively translates the sample in space and away from the focal point. So, Rotation followed by compensating Translation can be used to bring the sample back under microscope focal point.
5. We want to Image the sample with its center being at the focal point. This necessitates the knowledge about the thickness of the sample and the focal position of the top-surface (Phi=0) and bottom-surface (Phi=180).
6. Predict the position of the sample in 3D space and then translate to that point before Imaging.

3.7.1. Methods

The phi-rotation being independent from the translation (x, y, z), absolute movements during ϕ-rotation are possible using SmarGon. This effectively reduces the number of steps required for tomography dataset. Also, the relationship between top, bottom and middle layer of the sample are different for ϕ-rotation compared to that for ω-rotation. Refer to ***Figure 6*** for this section.

Method Outline

1. Acquire position of point $\boldsymbol{P_+(\theta = 0°)}$ and $\boldsymbol{P_-(\theta = 180°)}$ as shown in. Here "position" means the 6-coordinates' position of a specific point in the order: A, B, C, X, Y, Z.
2. Note 1:
    a. Rotation axes: A = Omega, B = Phi, C = Chi.
    b. Translation axes: X, Y, Z.
    c. For tomography, set A = 0 and C = 0.
    d. Thus for $P_+(\theta)$ will have the form: (A0 B? C0 X? Y? Z?).
    e. The user needs to provide only for the coordinates with an accompanying "?".

3. Note 2:
    a. For quadrants Q1 and Q4 only the upper surface is visible to the detectors.
    b. For quadrants Q2 and Q3 only the lower surface is visible to the detectors.
    c. The object is essentially rotated about its center (t distance away from both surfaces).
    d. The microscope is focused at the top surface at point-$\boldsymbol{A_+}$ ($\boldsymbol{A_+ = P_+(\theta = 0°)}$) to begin with, which essentially traverses a circular trajectory of radius $\boldsymbol{t}$ around the center of rotation, and passes through the point-$\boldsymbol{C_-}$ ($\boldsymbol{C_- = P_-(\theta = 180°)}$) when the object gets upside down.
4. Note 3:
    a. There are four sets of conjugate-pair of points: $\boldsymbol{(A_+, A_-)}$, $\boldsymbol{(B_+, B_-)}$, $\boldsymbol{(C_+, C_-)}$, $\boldsymbol{(D_+, D_-)}$.
    b. These points represent upper and lower surface points for $\theta = 0°, 90°, 180°, 270°$ respectively.
    c. These points are different from the axes labels: A (omega), B (Phi), C (Chi), X, Y, Z.

5. Measure the thickness ($\boldsymbol{2t}$) of the tomographic sample at $\theta = 90°\ and\ 270°$. Use the average of these two measured values of thickness as the measured thickness to use in subsequent calculations.
    a. $(2t)_{measured} = \frac{(2t)_{90°} + (2t)_{270°}}{2} = \boldsymbol{2t}$
6. Calculate position of the points on the other surfaces at both positions ($\theta = 0°, 180°$).
    a. For $\boldsymbol{P_-(\theta = 0°)}$**: Point-**$\boldsymbol{A_-}$
        - $\boldsymbol{Y_-(\theta = 0°) = Y_+(\theta = 0°) - 2t}$
        - $\boldsymbol{X_-(\theta = 0°) = X_-(\theta = 180°)}$
        - Correction for $\boldsymbol{Z_-(\theta = 0°)}$ is yet to be determined (WIP: Work-In-Progress). But this is not essential for current measurement requirements.
    b. For $\boldsymbol{P_+(\theta = 180°)}$**: Point-**$\boldsymbol{C_+}$
        - $\boldsymbol{Y_+(\theta = 180°) = Y_-(\theta = 180°) - 2t}$

- $\boldsymbol{X_+(\theta = 180°) = X_+(\theta = 0°)}$
- Correction for $\boldsymbol{Z_+(\theta = 180°)}$ is yet to be determined (WIP). But this is not essential for current measurement requirements.

7. Calculate the circular trajectory of the point midway through the sample's thickness.
8. If in future we require to predict the trajectory of the inner circle of all $\boldsymbol{P_-(\theta)}$ points (red line) or, the outer circle of all $\boldsymbol{P_+(\theta)}$ points (blue line), we will develop the algorithm accordingly. It is not required as of now.
9. Finally, the circular trajectory through the center of the sample (at a distance $\boldsymbol{t}$ from either surface) is calculated and SmarGon motion program code is generated for the user to test it manually.
10. The final predicted trajectory places the focal point at the midway through the sample's thickness. Any point thus tracked ensures that the sample is back under the field of view after each rotation. Since the point of interest on the sample is geometrically constrained to move with the sample, predicting the circular trajectory of that point alone elegantly solves the requirement of placing the sample under the field of view, while centering around the point-of-interest.
11. Once the user verifies that the trajectory prediction works reasonably well, the generated SmarGon code file can be used for unattended automated measurements.

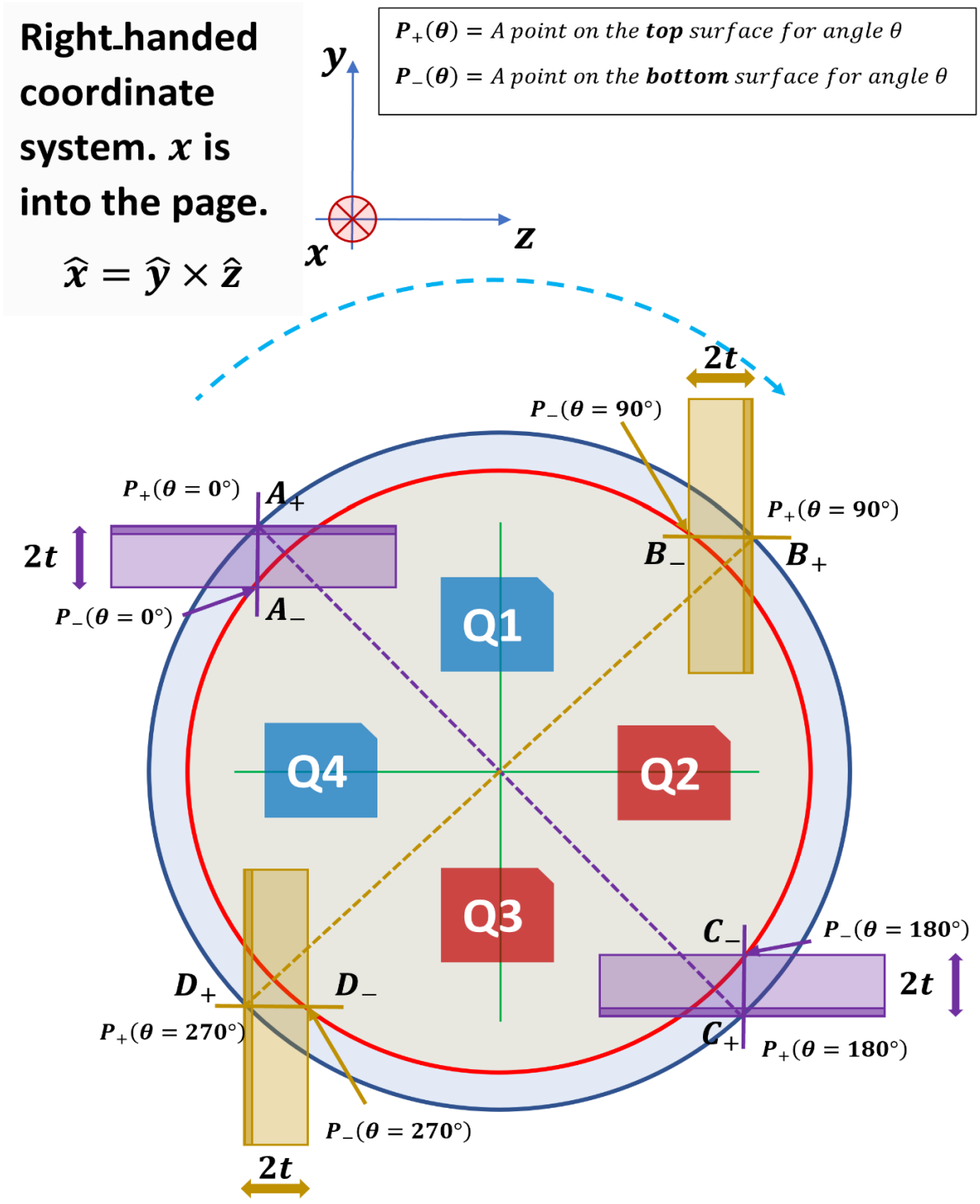


Figure 6: Phi Rotation Visualized.

The sample rotates from theta = 0 to 360 degrees. Trajectory: $ABCDA$. Outer surface is denoted by subscript "plus" (A_+, B_+, C_+, D_+) and inner surface by subscript "minus" (A_-, B_-, C_-, D_-). The points of interest for the outer and inner surfaces are: $P_+(\theta)$ and $P_-(\theta)$ respectively.

3.8. Omega Rotation

Since the omega rotation is the most precise rotation about the X axis, we decided to apply algorithmic prediction and subsequent correction of sample-position to omega rotation.

3.8.1. Challenges

The following challenges exist for ω-rotation:

3.8.1.1. Unknown Position of Omega Rotation-Axis

The omega rotation axis does not pass through (0,0,0). This creates a problem as the x, y, z readings during a rotation do not show the true position measured from the center of rotation. The solution was to determine the position of the omega-rotation-axis from multiple measurements.

3.8.1.2. Unknown Vector for Mounting the Sample

Once the sample is mounted, the effective position of it could be thought of a sum of several vectors each representing a distance vector constrained to move in a specific way along with the assembly. Multiple measurements of the same point under omega-rotation could be used to estimate the vector associated with mounting the sample. It should be noted that this final vector may most likely change from one sample to another and also for the same sample when it is mounted at different occasions.

3.8.1.3. Unknown amount of tilt between the microscope and the axis of omega-rotation

The weight distribution of the stage is not symmetric in terms of where it is placed on the custom-made mounting platform. This results in a variable amount of tilt every time the stage is brought into its operational position -- this tilt must be considered for all practical intended use of the stage.

In addition to this there could be some other rotation about the y and/or z axes in the absolute reference frame. At present we are trying to determine the amount of these tilts and prescribe a repeatable and practicable method to correct for such tilt.

3.8.1.4. It is more complicated than it sounds

Although we have mentioned the various challenges coming from omega-rotation as isolated events, in practice they are not. The user will only be able to measure the final effect of all these causes put together. So, we do not know a priori, which order of correction would be most effective. The solution will therefore be to use multiple strategies and test which yield better result than the others repeatedly irrespective of the sample mounted.

3.8.1.5. SmarGon develops some stress over the number of movements

SmarGon can operate in two ways: (**a**) *move to an absolute coordinate* and move the relevant motors simultaneously to reach the point; (**b**) *move incrementally* for the relevant motor/degree-of-freedom (**DoF**). It was observed that while working with omega rotation, approximately somewhere between 1000 and 1500 movements (incremental steps: translation/rotation) SmarGon often develops some stress and may move erroneously. At this point the only option left is to ***re-reference*** SmarGon and restart measurements. The problem with re-referencing SmarGon is that the measurements made afterwards most likely cannot be used in conjunction with the previous one. Currently we are trying to find a solution to this. Therefore, as of now, ϕ-rotation is the best available alternative for tomography.

3.8.2. Key lessons from studying Omega-rotation

1. Omega has more accurate rotation than Phi (manufacturer specification): smaller sphere-of-confusion. However, it develops stress over time for incremental steps and becomes prone to error.
2. The axis of rotation is not the x axis. It is parallel to x-axis.
3. Need a method to consider this unknown axis of rotation.
4. The only form of information available is visible light (VIS) measurement of the top or bottom surfaces under focus. This is because for pre-IR measurements, the only option available is to use visible light and setup the experiment.
5. User needs to define what is called to be “under focus” and verify the prediction with observations. Wrong focus means effectively wrong thickness of sample.
6. Omega rotation is coupled with the x/y/z translation. All translation under rotation is effectively in relative frame of reference. The absence of an absolute frame of reference poses the biggest challenge.
7. The arbitrary sample mounting vector must be included in the model predicting position.
8. Both predicted and measured data points should be used to extract information about the arbitrary sample mounting vector (ASMV).
9. Empirical correction can be applied for any other alignment issues between SmarGon and the microscope.

3.8.3. Methods

3.8.3.1. Relationship Between Top, Bottom and Middle Layers (Surfaces) of the Microloop

The objective is to rotate the microloop about its center (located on the plane-surface positioned midway between the top and the bottom surfaces) for tomographic projection measurements. However, the user can only observe the top or the bottom surfaces when $\omega = 0°$ and $\omega = 180°$ for $\phi = 0°$. This means the microloop is positioned such that the focal plane intersects with the top and the bottom surfaces respectively. In the following sections we prove the relationship between the top, bottom and the mid-layers in conjunction with the position of the focal plane (see **Figure *7***). The ability to control the position of the focal plane inside or on the surfaces of the microloop will be useful in interpreting how the phase of the final image is related to the position of the focal plane.

Define:

- **Thickness** of the microloop = $2t$
- **Top Surface** = surface of the microloop that is visible when $\omega = 0°$ and $\phi = 0°$.
- **Bottom Surface** = surface of the microloop that is visible when $\omega = 180°$ and $\phi = 0°$.
- **Middle Surface** = This surface is never visible as this one resides inside the microloop, midway between the Top and Bottom surfaces.

$Y_{\omega 0}^{+}$ = Y-position required to focus on the Top surface for $\omega = 0°$.

$Y_{\omega 0}^{-}$ = Y-position required to focus on the Bottom surface for $\omega = 0°$.

$Y_{\omega 0}^{0}$ = Y-position required to focus on a surface at a depth midway through the thickness for $\omega = 0°$.

Similarly,

$Y_{\omega \pi}^{+}$ = Y-position required to focus on the Top surface for $\omega = 180°$.

$Y^{-}_{\omega\pi}$ = Y-position required to focus on the Bottom surface for $\omega = 180°$.

$Y^{0}_{\omega\pi}$ = Y-position required to focus on a surface at a depth midway through the thickness for $\omega = 180°$.

Evaluation of Actual Center of Rotation for Omega Axis

It can be shown that for **Case-1**:

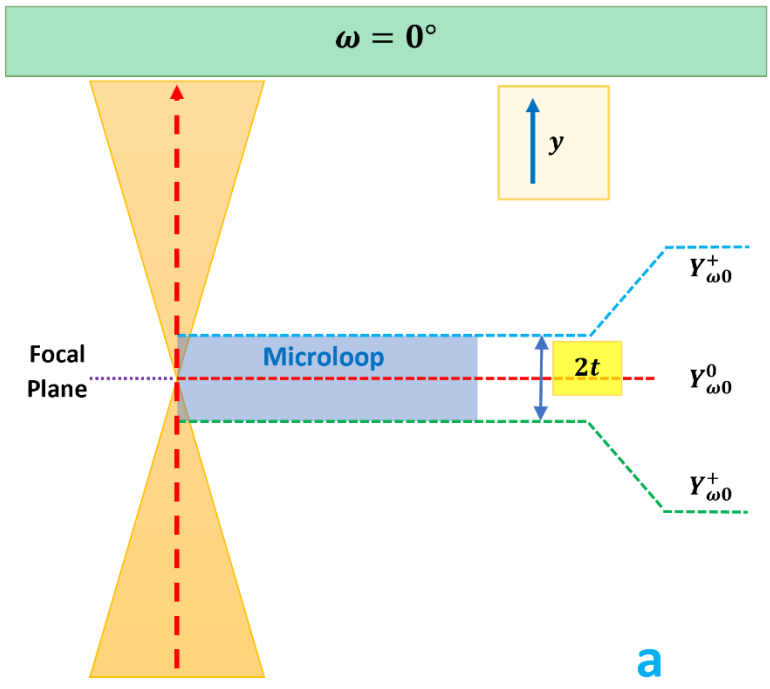
$$\boldsymbol{Y^{0}_{\omega\pi} = -\left[(Y_{CT} - 2Y_{CA}) + \left(Y^{0}_{\omega 0} - Y_{CT}\right)\right] = -Y^{0}_{\omega 0} + 2Y_{CA}}$$

Thus, $\boldsymbol{Y^{0}_{\omega\pi}}$ is independent of position of CT.

$$\boxed{\boldsymbol{Y^{0}_{\omega\pi} = -Y^{0}_{\omega 0} + 2Y_{CA}}} \qquad \textbf{Eq. C.01}$$

Using equations **Eq. 2.07a** and **2.08b**, **Eq. 2.05** can be recast into the following form to evaluate $\boldsymbol{Y_{CA}}$.

$$\boxed{\boldsymbol{Y_{CA} = \frac{(Y^{0}_{\omega 0} + Y^{0}_{\omega\pi})}{2} = \frac{(Y^{+}_{\omega 0} + Y^{-}_{\omega\pi})}{2}}} \qquad \textbf{Eq. C.02}$$



Note: To bring the top surface under focus from the mid-surface (-----), one needs to move the microloop **DOWN** by half its thickness (t). Thus, for the top surface (-----)

$$Y^+_{\omega 0} = Y^0_{\omega 0} - t$$

$$\Rightarrow Y^0_{\omega 0} = Y^+_{\omega 0} + t$$

Similarly, for bottom surface (-----),

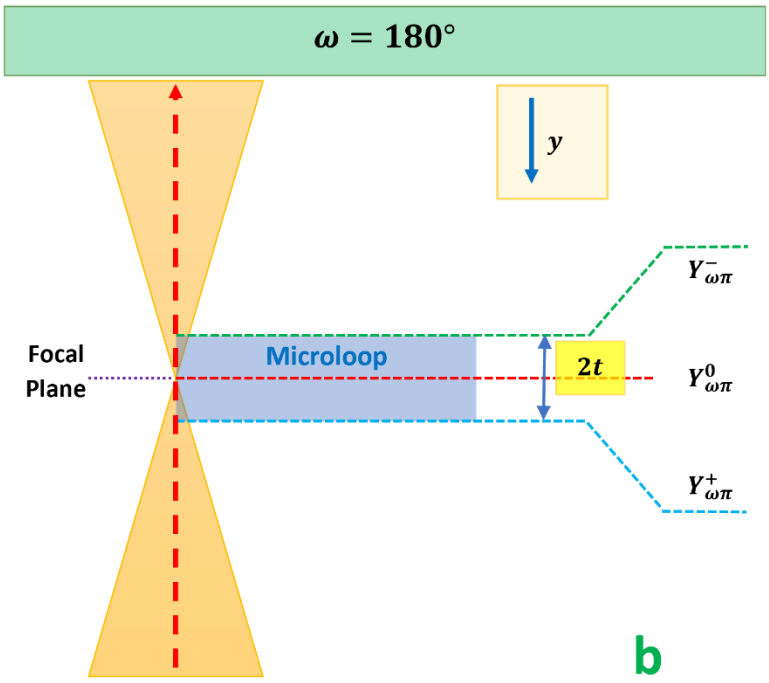
$$Y^-_{\omega 0} = Y^0_{\omega 0} + t$$

$$\Rightarrow Y^0_{\omega 0} = Y^-_{\omega 0} - t$$

We will use the following equations later:

$$Y^0_{\omega 0} = Y^+_{\omega 0} + t \quad \text{Eq. C.03a}$$

$$Y^0_{\omega 0} = Y^-_{\omega 0} - t \quad \text{Eq. C.03b}$$



Note: To bring the top surface under focus from the mid-surface (-----), one needs to move the microloop **UP** by half its thickness (t). Also, note that in this case the direction of y-axis has flipped after 180° rotation in ω.

Thus, for top surface (-----)

$$Y^+_{\omega \pi} = Y^0_{\omega \pi} - t$$

$$\Rightarrow Y^0_{\omega \pi} = Y^+_{\omega \pi} + t$$

Similarly, for bottom surface (-----),

$$Y^-_{\omega \pi} = Y^0_{\omega \pi} + t$$

$$\Rightarrow Y^0_{\omega \pi} = Y^-_{\omega \pi} - t$$

We will use the following equations later:

$$Y^0_{\omega \pi} = Y^+_{\omega \pi} + t \quad \text{Eq. C.04a}$$

$$Y^0_{\omega \pi} = Y^-_{\omega \pi} - t \quad \text{Eq. C.04b}$$

Figure 7: Relationship between top, bottom and middle surface for Omega-rotation.

At $\omega = \pi$ the axes flip upside down, the relationship between top, bottom and middle surfaces must consider this. (**a**) $\omega = 0$ and (**b**) $\omega = \pi$.

Primary Procedure

The following steps and formulas are **applicable for all four cases** mentioned earlier: **Case-1, 2, 3, 4**.

The objectives are:

1. Calculate the Y-position of actual center of omega rotation (CT).
2. Predict the required Y-coordinate of the Bottom Surface kept at focal plane.

For a MiTeGen microloop of aperture diameter $50\ \mu m$ and known thickness, $\boldsymbol{2t} = 10\ \mu m$ we will follow the steps below to accomplish Objectives 1 and 2 mentioned above.

The steps are:

1. Calculate the center of omega-rotation, $\boldsymbol{Y_{CA}}$ using **Eq. C.02**. Known values are $\boldsymbol{Y^{+}_{\omega 0}}$ and $\boldsymbol{Y^{-}_{\omega \pi}}$.

$$\boldsymbol{Y_{CA}} = \frac{(\boldsymbol{Y^{+}_{\omega 0}} + \boldsymbol{Y^{-}_{\omega \pi}})}{\boldsymbol{2}}$$

2. Calculate $\boldsymbol{Y^{0}_{\omega \pi}}$ by combining equations **C.02** and **C.03a** as follows:

$$\boldsymbol{Y^{0}_{\omega \pi}} = -(\boldsymbol{Y^{+}_{\omega 0}} + \boldsymbol{t}) + \boldsymbol{2Y_{CA}}$$

3. Calculate $\boldsymbol{Y^{-}_{\omega \pi}}$ using **Eq. C.04b**.

$$\boldsymbol{Y^{-}_{\omega \pi}} = \boldsymbol{Y^{0}_{\omega \pi}} + \boldsymbol{t}$$

4. Test if $\boldsymbol{Z_{CA}}$ is equal to:

$$\boldsymbol{Z_{CA}} = \frac{(\boldsymbol{Z^{+}_{\omega 0}} + \boldsymbol{Z^{-}_{\omega \pi}})}{\boldsymbol{2}}$$

5. Record the coordinate of the center of rotation:

$$\boldsymbol{X_{CA}} = \frac{(\boldsymbol{X^{+}_{\omega 0}} + \boldsymbol{X^{-}_{\omega \pi}})}{\boldsymbol{2}}$$

$$\boldsymbol{Y_{CA} = \frac{(Y^{+}_{\omega 0} + Y^{-}_{\omega \pi})}{2}}$$

$$\boldsymbol{Z_{CA} = \frac{(Z^{+}_{\omega 0} + Z^{-}_{\omega \pi})}{2}}$$

6. Calculate radius of ω-rotation:

$$\boldsymbol{R_{\omega 0} = \sqrt{(X^{+}_{\omega 0} - X_{CA})^2 + (Y^{+}_{\omega 0} - Y_{CA})^2 + (Z^{+}_{\omega 0} - Z_{CA})^2}}$$

$$\boldsymbol{R_{\omega \pi} = \sqrt{(X^{+}_{\omega \pi} - X_{CA})^2 + (Y^{+}_{\omega \pi} - Y_{CA})^2 + (Z^{+}_{\omega \pi} - Z_{CA})^2}}$$

7. Cross-check that the following relation holds:

$$\boldsymbol{R_{\omega} = R_{\omega \theta} = R_{\omega 0} = R_{\omega \pi}}$$

8. Calculate trajectory as:

    a. Rotation matrix for ω-rotation by angle θ:

$$\omega_{\theta} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\theta & -sin\theta \\ 0 & sin\theta & cos\theta \end{pmatrix}$$

    a. Define *CA* position:

$$P_{CA} = \begin{pmatrix} X_{CA} \\ Y_{CA} \\ Z_{CA} \end{pmatrix}$$

    b. Define $P_{absolute,\theta}$:

$$P_{absolute,0} = \begin{pmatrix} \boldsymbol{X^{0}_{\omega 0}} \\ \boldsymbol{Y^{0}_{\omega 0}} \\ \boldsymbol{Z^{0}_{\omega 0}} \end{pmatrix}$$

    c. Define *Relative* position:

$$P_{relative,0} = P_{absolute,0} - P_{CA}$$

$$\Rightarrow P_{relative,0} = \begin{pmatrix} \boldsymbol{X^{0}_{\omega 0} - X_{CA}} \\ \boldsymbol{Y^{0}_{\omega 0} - Y_{CA}} \\ \boldsymbol{Z^{0}_{\omega 0} - Z_{CA}} \end{pmatrix}$$

d. Calculate absolute position in space to predict trajectory. Note here we calculate the trajectory of the point defined by the intersection of the planar middle surface and the straight line that join $P_{absolute,\theta=0}$ and $P_{absolute,\theta=\pi}$. This needs correction for the direction flip

$$P_{absolute,\theta} = P_{CA} + \omega_\theta P_{relative,0}$$

e. The ω-rotation must be done as follows:

- Do not use ω-rotation along with xyz-translation.
- First goto P_{CA}.
- Execute ω-rotation.
- Translate to the expected xyz-coordinates. Now you are at $P_{absolute,\theta}$.

f. **NOTE**: The motion of the stage to bring the top (or, *bottom*) surface at the focal plane is:

- Opposite to the direction from the central plane to the desired surface for ω-rotation.
- Along the direction from the central plane to the desired surface for ϕ-rotation.

9. Let V be the arbitrary mounting vector from the $(X_{CA}, 0,0)$ to the point of observation on the sample when the SmarGon coordinates are: $\omega = 0, \phi = 0, \chi = 0, x = X_{CA}, y = 0, z = 0$. In future, we will refer to these coordinates as (A, B, C, X, Y, Z). After a rotation of $\omega = \theta$, the arbitrary vector V becomes $V' = \omega_\theta V$. The change is: $dV = V' - V = V(1 - \omega_\theta)$. dV is estimated from: $dV = (measured\ P_{absolute,\theta}) - (predicted\ P_{absolute,\theta})$. Although this method reduces the error to a few microns, the overall time required to

setup using omega is overwhelmingly long (2-3 hours). Therefore, we are using Phi-rotation as of now.

3.9. The Elegant Solution: Phi-Rotation

As stated before, phi-rotation at present offers a better and more reliable solution compared to omega-rotation for tomography. This is primarily because:

1. Phi-rotation is uncoupled from the translational motion of TomoStage (SmarGon).
2. Any point that needs to be tracked during rotation only needs two inputs:
    a. SmarGon Coordinates of the point on top/outer surface at $\phi = 0°$.
    b. SmarGon Coordinates of the corresponding point on bottom/inner surface at $\phi = 180°$. The point on the bottom surface is directly across the sample along its thickness, with respect to the point on the top surface.

The phi-rotation can be achieved so elegantly because as we predict and follow the trajectory of the point-of-interest, the rest of the sample also rotates while maintaining the geometric constraint relations of the sample's shape and size with respect to the point-of-interest. The entire setup can be achieved in about 30 minutes, making this a practicable option for laboratory implementation. In contrast, the method for omega-rotation requires at least 20 measurements followed by a lot more pre-processing. The overall procedure of omega-rotation prediction could take up to three hours – making it a less desirable option.

3.10. Results and Discussions

Here we present some images captured with Visible Light (Vis) and Infrared (IR) source.

3.10.1. Vis Images: Examples

The following four figures (**Figure 8**, **Figure 9**, **Figure 10**, **Figure 11**) show Vis Images for various projections.

Frame: 02 | Projection Angle: 10 degrees | Observation: In-focus, clean.

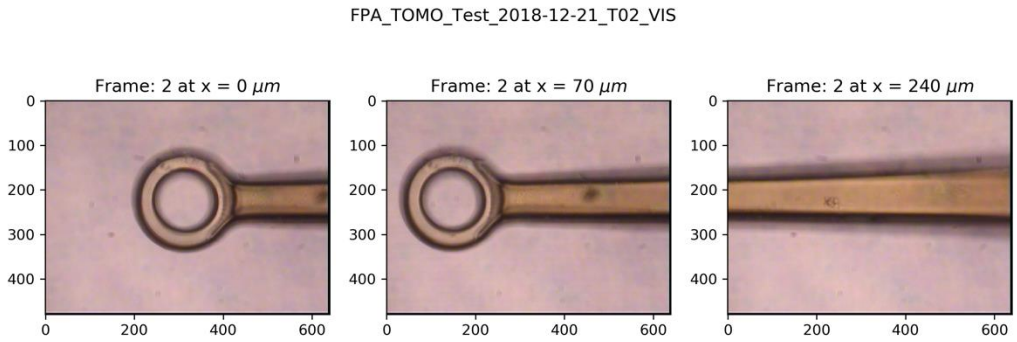


***Figure 8**: Visible light images of the microloop at different x positions for Frame-2*

Frame: 23 | Projection Angle: 115 degrees | Observation: Partly in-focus, partly blurry.

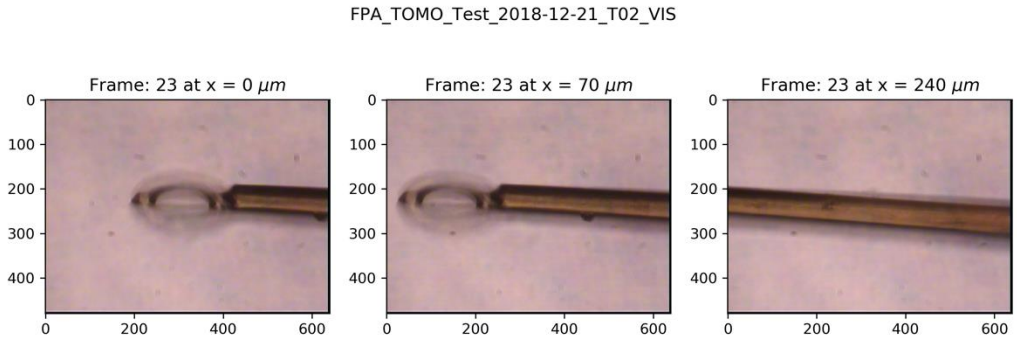


***Figure 9**: Visible light images of the microloop at different x positions for Frame-23*

If we look closely, the figures show some blurriness resulting from part of the microloop being outside the depth-of-focus while imaging with the visible light.

Frame: 45 | Projection Angle: 225 degrees | Observation: Partly in-focus, partly blurry.

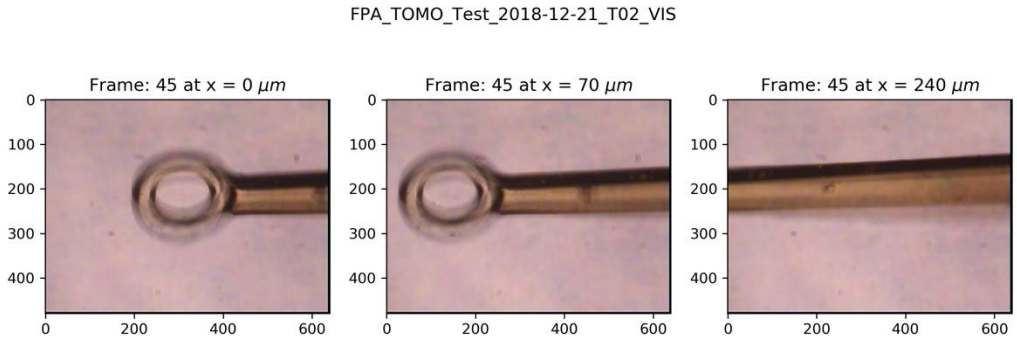


***Figure 10**: Visible light images of the microloop at different x positions for Frame-45*

Frame: 54 | Projection Angle: 270 degrees | Observation: Partly in-focus, partly blurry.

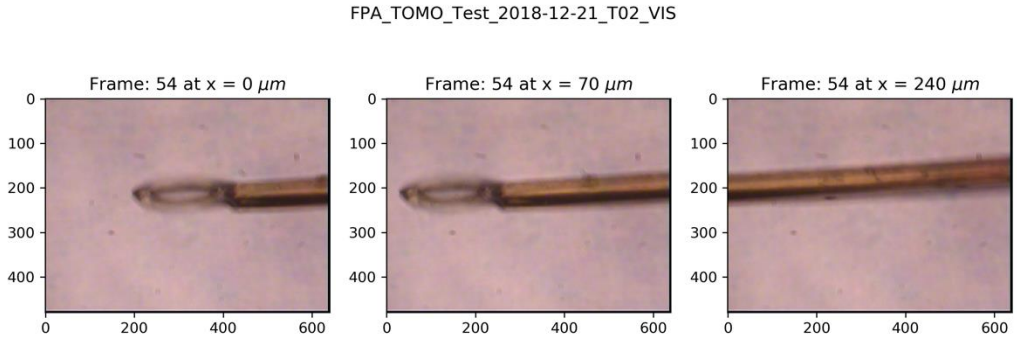


***Figure 11**: Visible light images of the microloop at different x positions for Frame-54*

3.10.2. Sinograms: IR Images

The following two sinograms (**Figure 12** and **Figure 13**) show that they are not smooth. The jaggedness in the sinograms is indicative of image displacement between subsequent projection measurements. This is a problem, as any Tomography routine will require the input dataset to be centered around the axis of rotation. We will explore possible alternative solutions in the subsequent chapters.

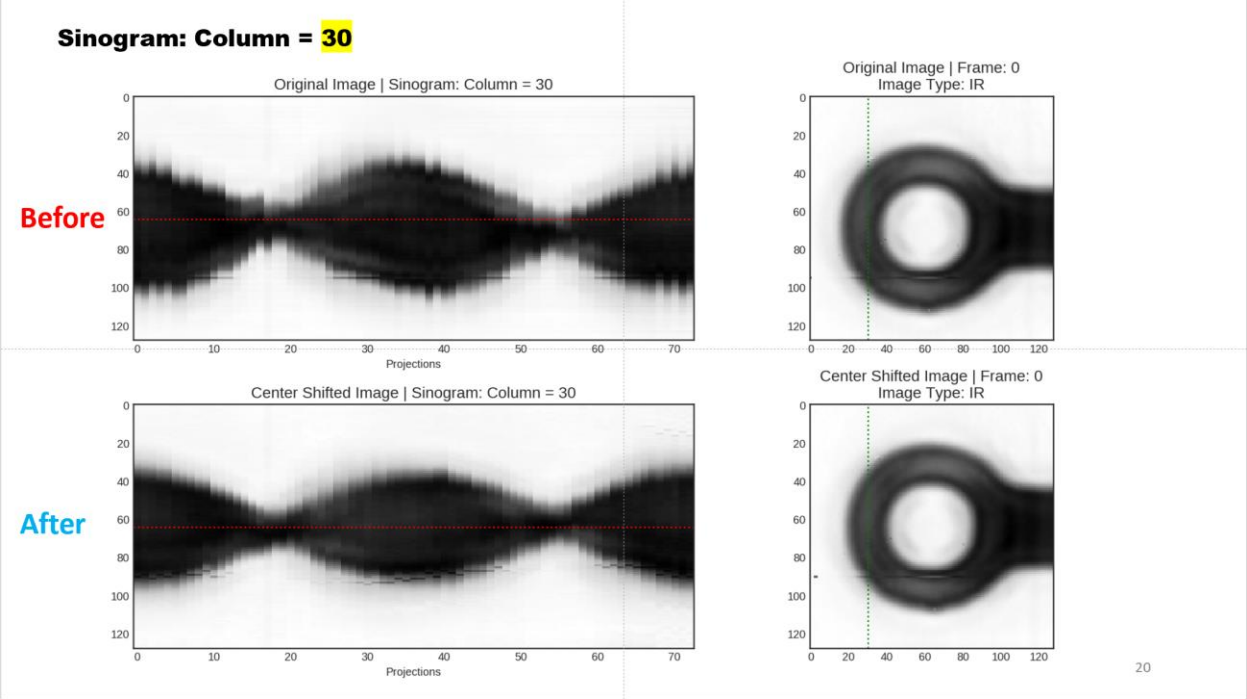


Figure 12: Singogram of IR Mircoloop Images at Column 30

The sinogram here has two parts: **Before** and **After** (**Figure 12**). The part ***Before*** pertains to the output of the automated tomography measurements discussed in this chapter. The part ***After*** shows the sinogram correction we have done and is described in the chapter next chapter. We would also like to note that this jaggedness is not a one-off behavior observed for some specific projections. It is rather effortlessly visible in all the projections. Here is the result using another sinogram

(**Figure 13**) constructed with a different column (60th). The previous one was made with the 30th column from all the projections.

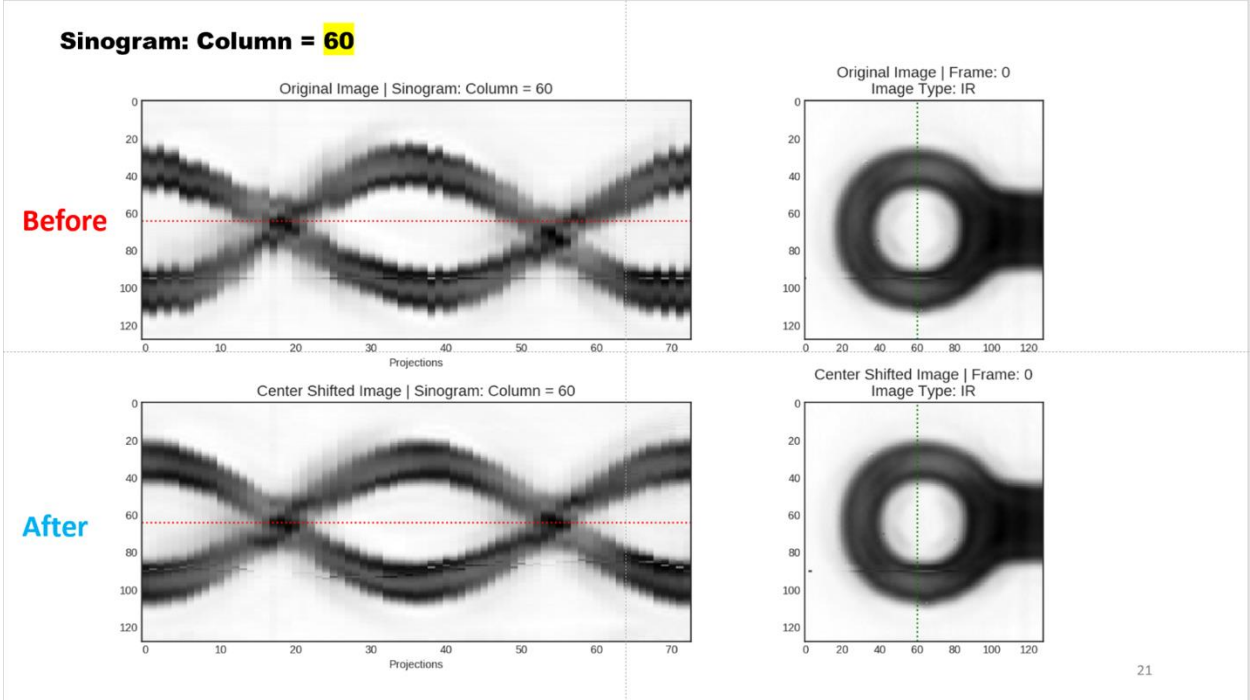


Figure 13: Singogram of IR Mircoloop Images at Column 60

In the next chapters we will introduce concepts and strategies used to smoothen the sinogram or devise a working strategy to improve the quality of the sinogram.

4. Algorithmic Correction to Images Acquired using Phi-Rotation

4.1. Analysis of Images Captured with Visible Light Source

Although Phi-rotation has a higher sphere-of-confusion than Omega-rotation, currently it is the best option for tomography. The following (**Figure 14**) is a polyamide microloop with an aperture diameter of 50 μm and thickness 10 μm. The image was taken with visible light as visible light imaging is faster than IR imaging. We are exploring means to algorithmically correct the microloop projections in cases where the axis of rotation does not pass through the center of the microloop stick as well as the center of the aperture. The TomoStage (SmarGon) operation with Phi-rotation comes with a constraint: it can only be set such that the axis of rotation passes through the center of the aperture. In most cases the axis of rotation will not align with the axis of the stick.

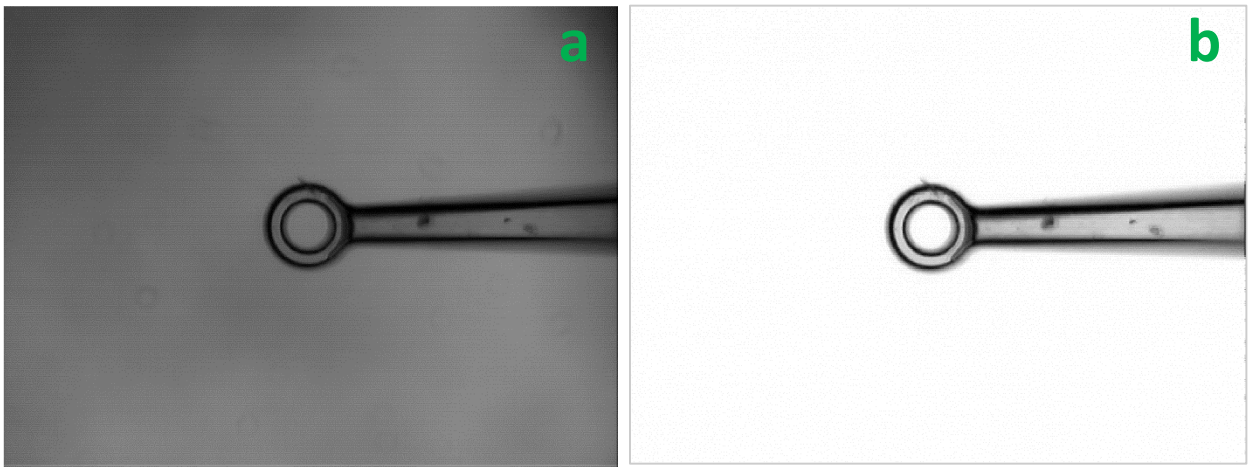


Figure 14: Microloop projection at 0° rotation.

Aperture diameter is 50 μm and thickness is 10 μm. The image was taken with visible light and a 15x Schwarzschild Objective. A mean background image was calculated from a total of 119 images without the microloop. To reduce background noise and increase SNR for the microloop for the images measured with visible light, each image was divided by the mean background image. (**a**) image without background suppression and (**b**) image with background suppression. Image dimension: 480x640 sq-pixels.

A total of 73 projections were collected, each 5 degrees apart. It's a standard practice to normally visualize the sinogram obtained from such projections. **Figure 15**(a) shows the vertical strip (*dashed-red-line*) picked from each projection and stacked horizontally to create the sinogram in **Figure 15**(b). The position of the vertical strip is column-270. When we look at all the projections it is observed that the axis of phi-rotation is not collinear with the symmetry-axis along the stick **Figure 16**(c) and **Figure 17**(a). The asymmetric sinogram (about the horizontal dashed-red-line) in **Figure 15**(b) confirms that the stick-axis was not collinear with the x-axis. We attempt to extract the angular offset between the stick's symmetry axis and the phi-axis by means of an algorithm, employing image processing and mathematical correction.

4.1.1. The Procedure

1. **Image Binarization**: Convert the projection images into binary images.
2. **Image Inversion**: Invert the binary images to have ones where the microloop is and zeros elsewhere.
3. **Target Area Selection**: Select a portion of the images so as to select a region from the stick that is present in all the images.
4. **For each projection**:
    a. Break this portion of the stick into vertical columns.
    b. Calculate mean (mean_1) and range (range_1) for pixels that have non-zero value.
    c. Use Least Square linear fit to the mean of the vertical strips for the Target Area on the Stick to estimate the equation of the stick-axis.
    d. Use the slope to rotate the image so that the axis now becomes horizontal.
    e. Calculate the mean (mean_2) and range (range_2) again.

f. Use the column position corresponding to the range-maxima at the center of the aperture to get the x-coordinate of the center.

g. Similarly determine the y-coordinate of the center of the aperture.

However, the center determined in different projections (post image rotation) keeps moving a few pixels from one image to another. We are currently working on devising a stable algorithm to ascertain the axis and the aperture-center simultaneously. The images shown here follow the following axes orientation (**Figure 15**):

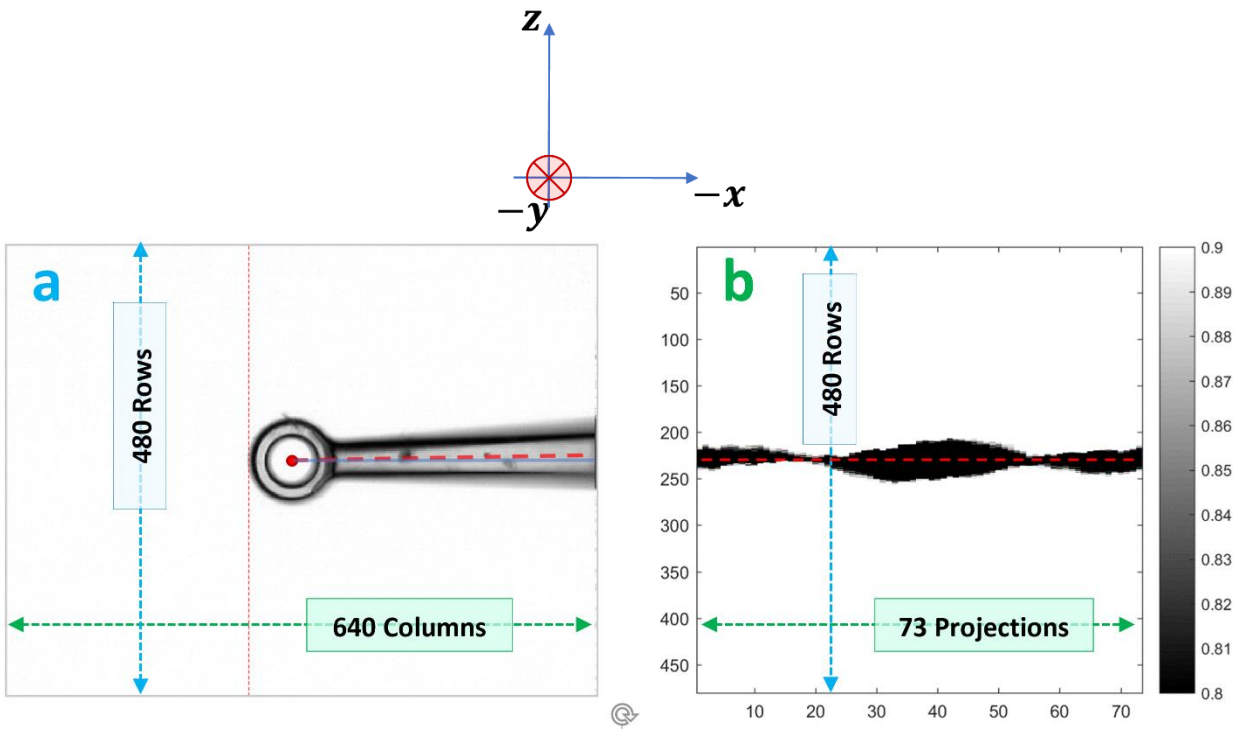


***Figure 15**: Microloop Sinogram.*

(a) The vertical strip (red dotted line at left edge of the microloop) depicts the column-position used for making the sinogram, **(b)** sinogram of the vertical strip at position column-270 across all 73 projections. The projections were measured 5 degrees apart. The Figure 8(a) is the same as Figure 7(b). The dashed red line in the sinogram shows that it is not symmetric above and below the center (dashed red line). The red-dot in figure-(a) is the aperture-center and the red dashed line in figure-(a) is the true axis of the microloop-stick. The semi-transparent solid blue line shows that even for *0°* rotation, the microloop axis was slightly at an angle with the ϕ-rotation axis (x-axis).

The following two figures (**Figure 16** and **Figure 17**) show that for non-horizontal orientations of the microloop stick-axis, the axis can be identified algorithmically and then the image can be rotated accordingly to make the stick-axis horizontal. Two microloop projections are chosen such that the loop-stick-axis is clearly at an angle with the horizontal x-axis. **Figure 16**(c) shows the identified orientation of the stick-axis as red dashed line for $\boldsymbol{\phi = 115°}$ and **Figure 17**(a) shows

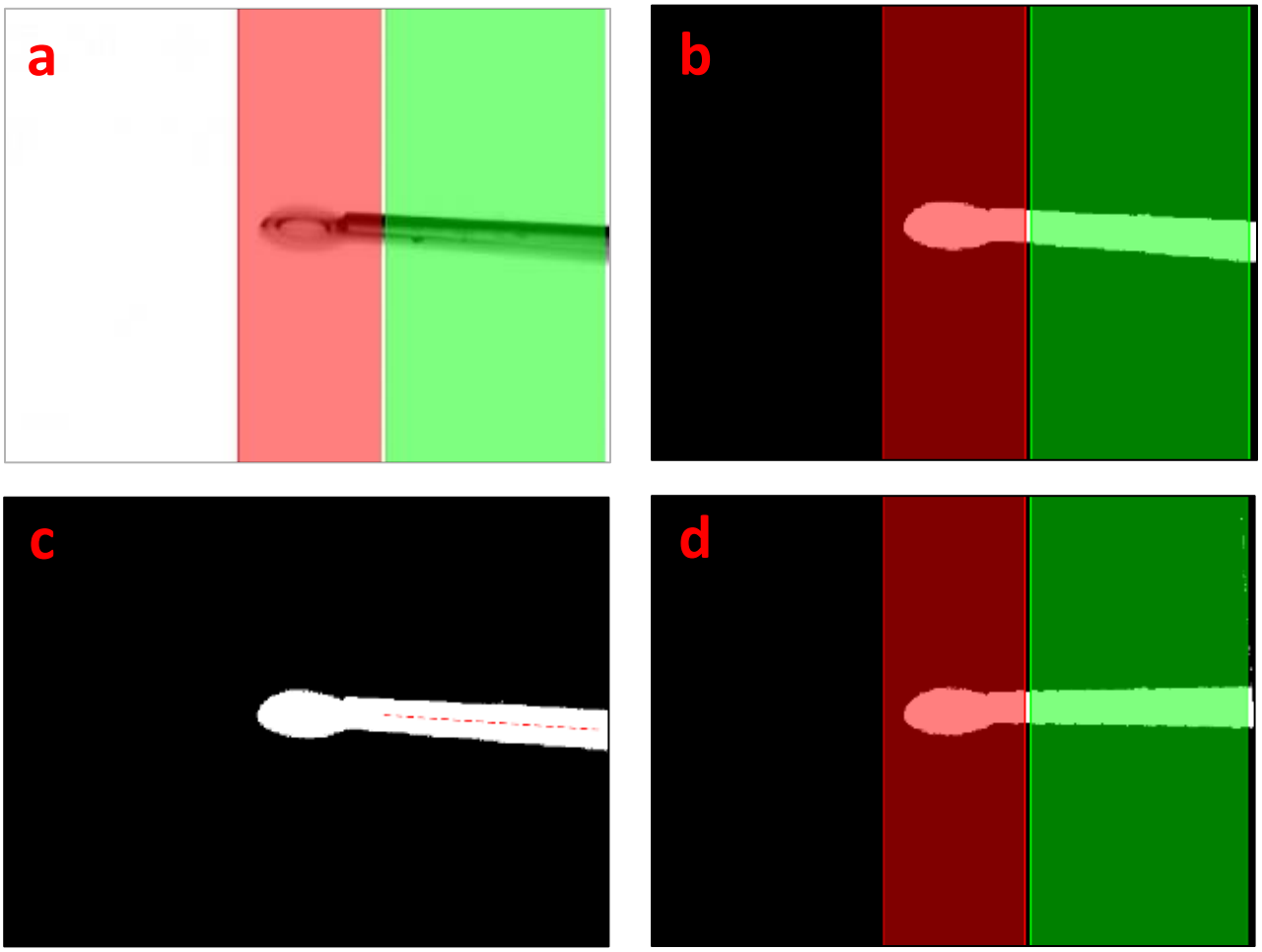


Figure 16: Identification of Microloop's Stick-Axis and Image Rotation at $\phi = 115°$.

The figures (a), (b), (c) and (d) show how the stick-axis of the microloop is identified and the image is rotated to create projections that effectively rotate about the stick-axis. (**a**) The red rectangle contains the loop region and the green rectangle contains the target section of the microloop stick. (**b**) Binarized image of the microloop in (a). (**c**) The red dashed line shows the identified stick-axis. (**d**) Axis-rotated image of (c).

the orientation of the stick-axis (*red dashed line*) for $\boldsymbol{\phi = 225°}$. Once the axis-orientations are identified, the images are rotated accordingly. However, the algorithm is not yet able to coincide

the center of the microloop from various projections after rotation. Currently, we are working on devising a statistically sound algorithm that can collocate and coincide the center of the microloop from all the projections.

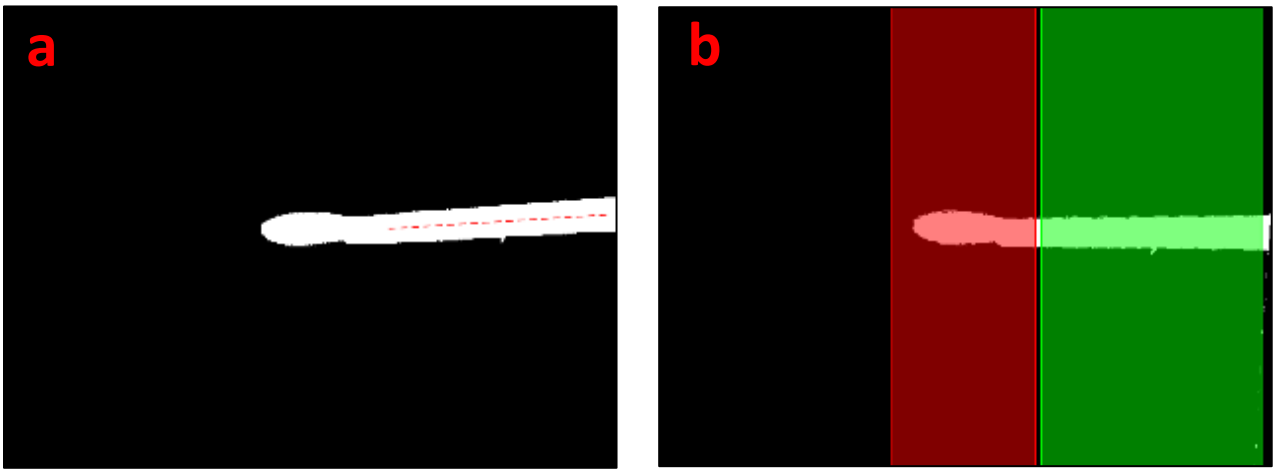


Figure 17: Identification of Microloop's Stick-Axis and Image Rotation at $\phi = 225°$.

(**a**) Microloop stick-axis identified: red dashed line. (**b**) Axis-rotated image with the *loop-region* and the *stick-region* highlighted by ***red*** and ***green*** semitransparent rectangles respectively.

4.2. Analysis of Images Captured with FTIR-MS

Since the sinograms were found to be jagged, we knew that the subsequent projection images were not aligned. However, we needed to know how much of restoring offset is necessary for each projection image, so that all the images effectively get aligned with the microloop center as a common center.

So, we manually labelled the centers (see **Figure 19** and **Figure 20**) on each of the 73 projections (each 5 degrees apart) with the location of their microloop centers. Since we had only one microloop per image, this was a relatively simple task. What we learned from the manual labelling exercise, is that for certain projections, closer to 90 or 270 degrees, it could get increasingly harder for one to precisely locate the position of the microloop center.

Thus, to get around such ambiguity, we created a python tool that will ask the user for left, right, top and bottom margins that contain the inner-aperture or the external surface of the microloop, within the image. The tool then essentially shows the guiding lines (green dotted lines) and the center of the image (red dotted lines). This helps the user iteratively determine the center of the microloop in a projection. Once a final steady value is achieved, the user can record that.

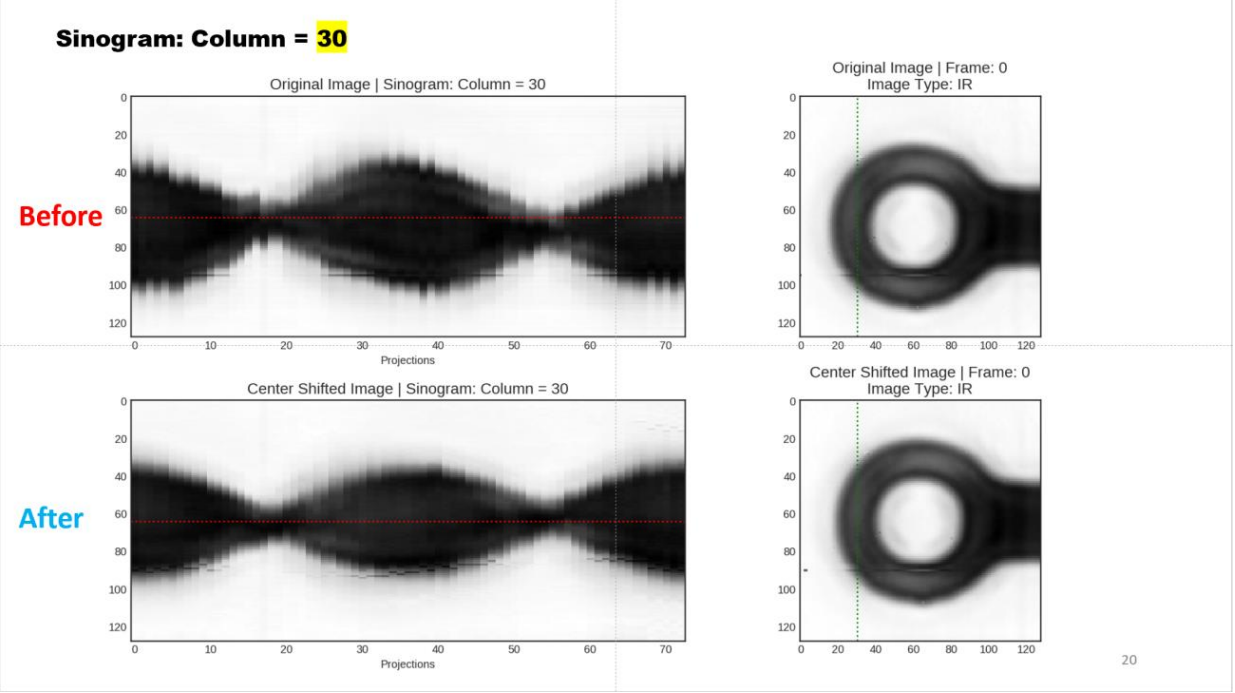


***Figure 18**: Manually correcting microloop sinogram.*

The following images (**Figure 19** and **Figure 20**) show how a trail center is used to guide the user to find the user determined center (*prediction*). Once we had these manually determined centers, we applied them to offset the projection images and tested the effect of these offsets on the projection images' sinogram. It was observed that the sinogram became considerably smoother as compared to before.

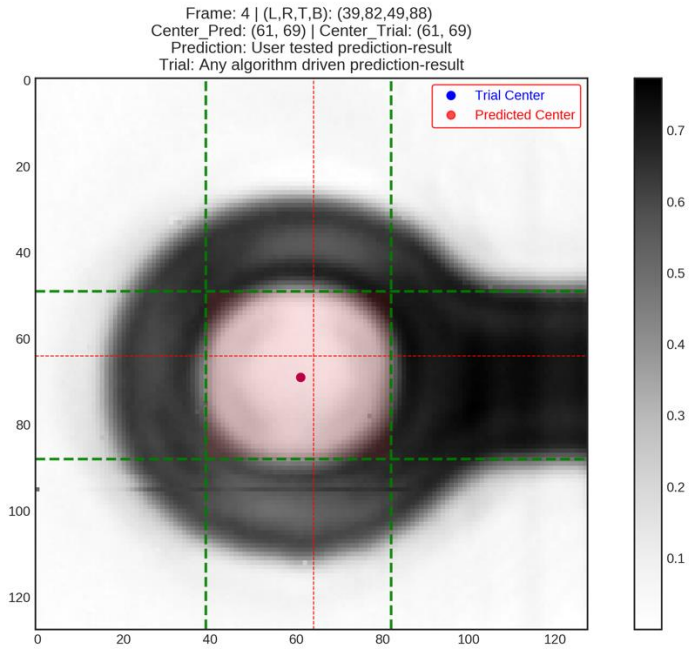


***Figure 19**: Manually identifying microloop center for Frame 4 (20 degrees projection).*

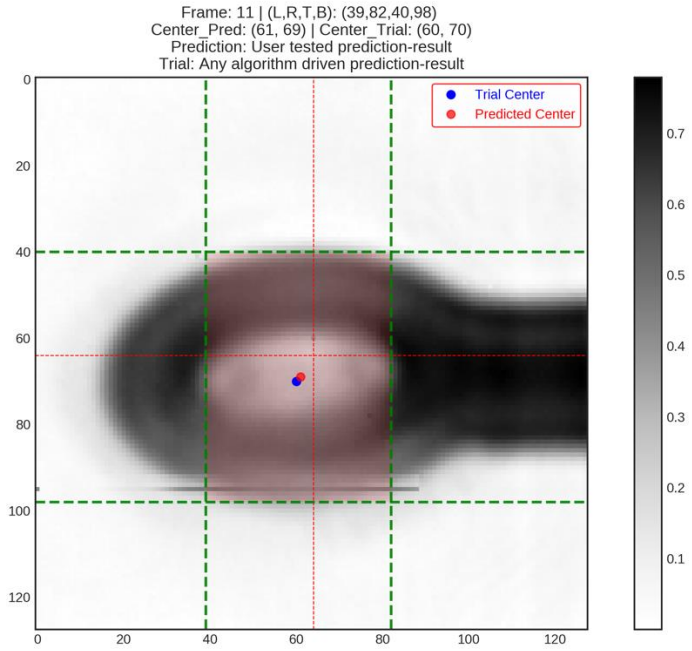


***Figure 20**: Manually identifying microloop center for Frame 11 (55 degrees projection).*

5. Edge Detection and Feature Engineering

The images acquired with FTIR-MS could be blurry around the edges due to diffraction, scattering, or could be simply out of focus. This could pose some challenges for fitting an ellipse, estimating the center of the microloop or using the images with deep learning. Simply put, we need a clean background subtracted image of the microloop and the sample in each of the projections.

Why Binarize?

Binarization of an image classifies every pixel into two classes: *black* or *white*. A binarized image is often called a mask. Given, we have a mask that determines which pixels in the image belong to the assembly of the sample and the microloop-holder, we can use it for downstream tasks of image-segmentation.

The FTIR Micro-Spectroscopy (MS) images we measured would typically have a dimension of 128x128x770 as Height x Width x Channels (or Layers). One of the challenges with the images was that they were not centered. This creates a jitter and displaces the image-slices on the sinogram for tomography. If we can precisely extract the assembly as a mask from each of the projections, then we could use these masks for identifying the loop section of the microloop-holder. This mask extraction effectively acts as a preprocessing step, enhancing features of the input images.

In our attempt at solving the object-positioning-uncertainty while measuring with FTIR MS for Tomography dataset, we realized a need for a method of binarization of the hyperspectral images (see **Figure 21**).

***Figure 21**: Schematic diagram of converting an HSI multilayer image into a binarized single layer image.*

One possible approach to this HIS binarization could be setting a pixel threshold by manual inspection. But a possible pitfall of this method could be frequent manual inspection of the validity of the binarization-threshold.

The binarization step was necessary for the input to the deep learning solution we developed for positioning-correction of the object-of-interest measured for FTIR MS Tomography. We preferred an alternative method that requires less human intervention and offers a rather standard approach to HSI binarization.

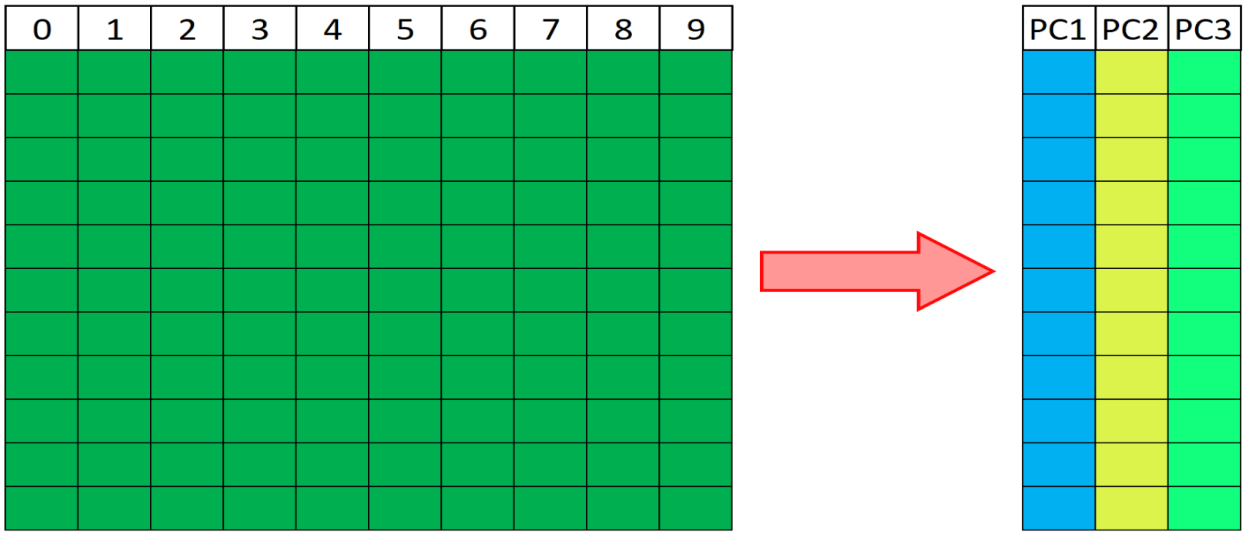


***Figure 22**: The notion of what PCA does to your data.*

PCA Based Binarization of Hyperspectral Images

Principal Component Analysis (PCA) is a method, typically used for dimensionality-reduction of data.

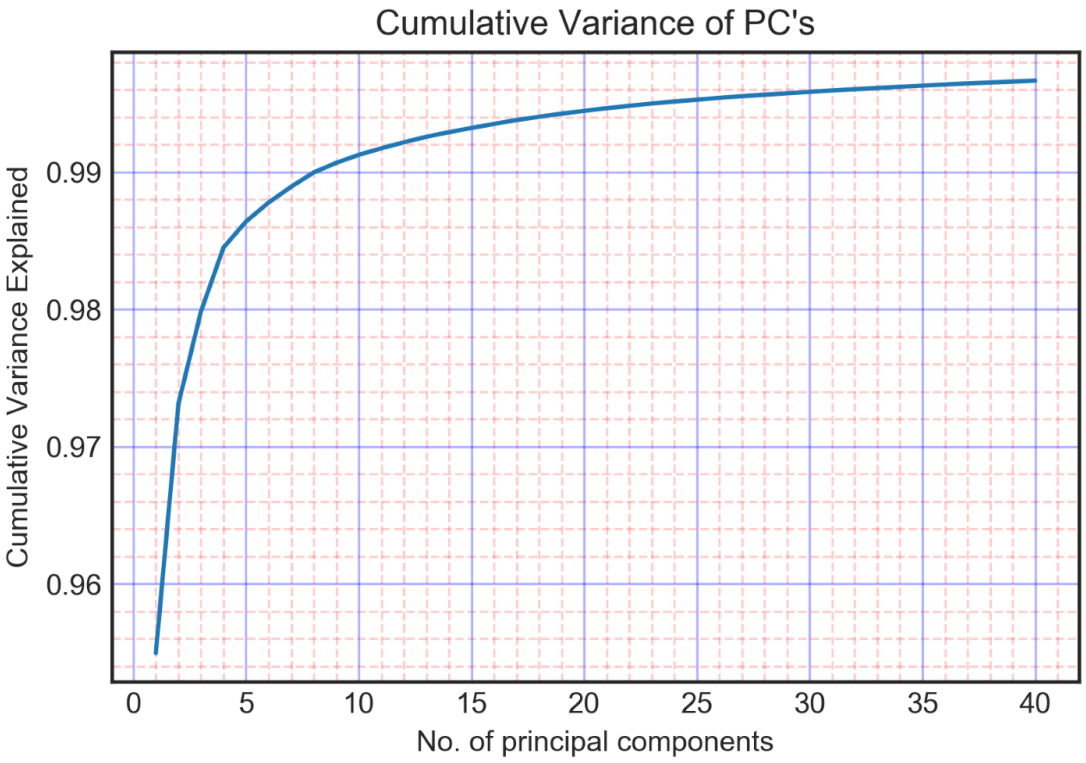


***Figure 23**: Using Cumulative Variance Explained as a metric to determine the number of Principal Components.*

We started with a total of 40 principal components on the data from the entire dataset of projection-0 of holder-only-data. The EVR (explained variance ratio) plot (see **Figure 23**) shows that inclusion of all the 40 components is as good as explaining the data with 99.75% information. The first PC alone explains 95.5%, the first three PCs together explain 98%, and the first eight PCs explain 99% of the data.

Choosing only the first PC for projection 0 degree, yields a surprisingly clear image (**Figure 24**) of the microloop, accentuating its features.

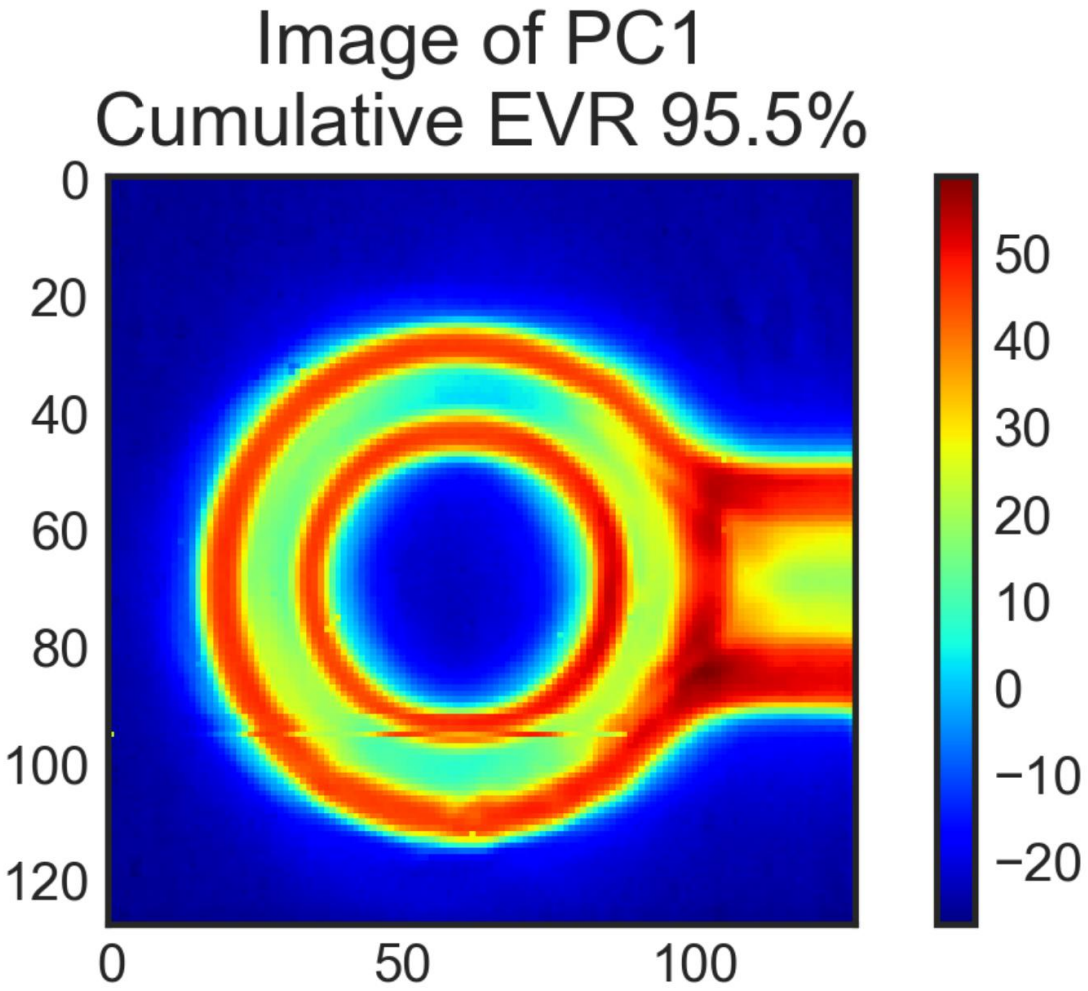


Figure 24: *PC1 for Projection 0 (0 deg)*

Figure 25 shows how all the first 40 PCs look like for the projection of 0 degree.

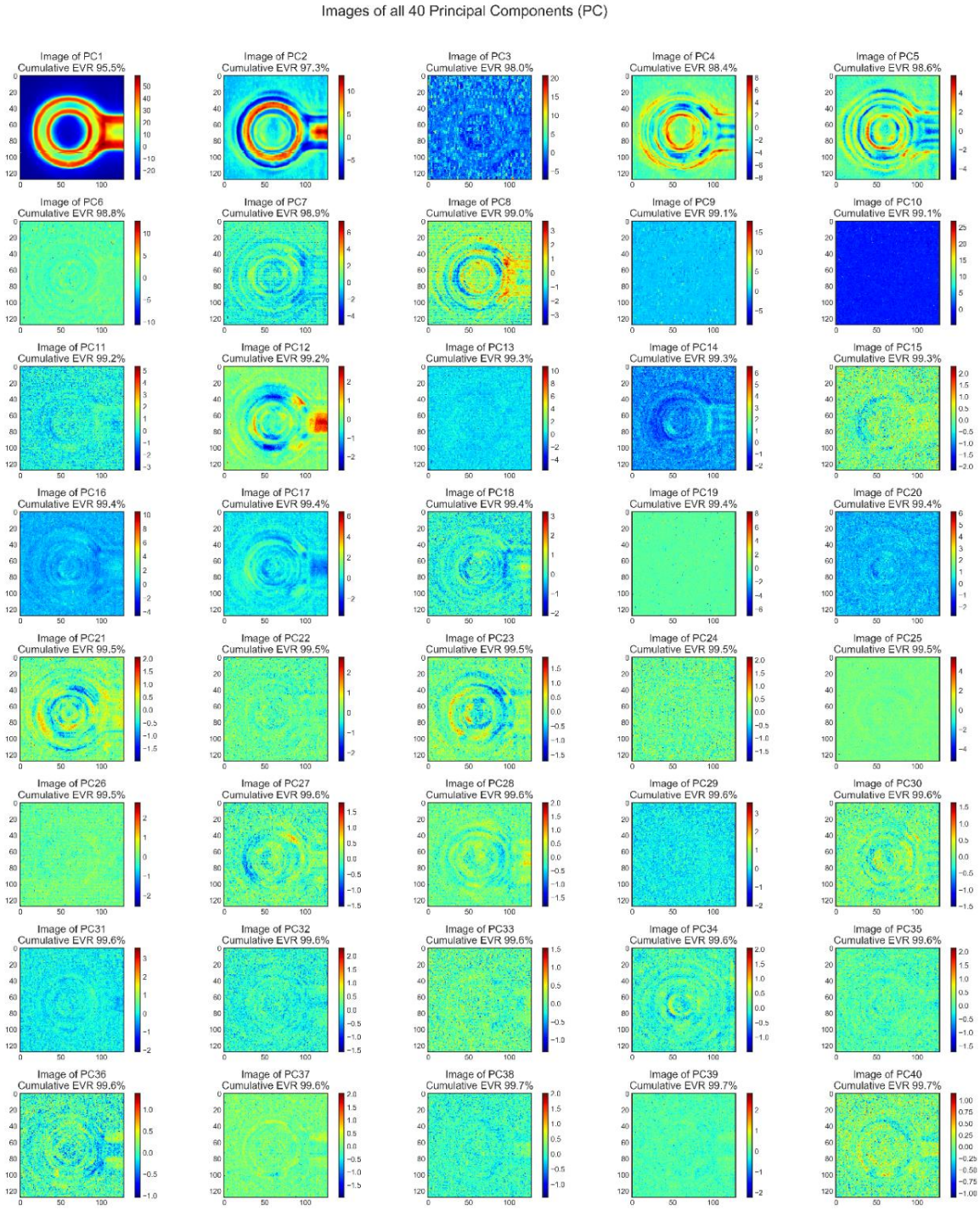


***Figure 25**: All 40 Principal Components (PCs) of the 0-degree microloop IR projection data.*

The following figure (**Figure 26**) compares the first principal component (PC) for projections 0 degree, 30 degrees and 90 degrees evaluated along the dimension of the wavelength. This effectively reduces the 128x128x770 dimensional dataset to 128x128x1 (as discussed in **Figure 21**).

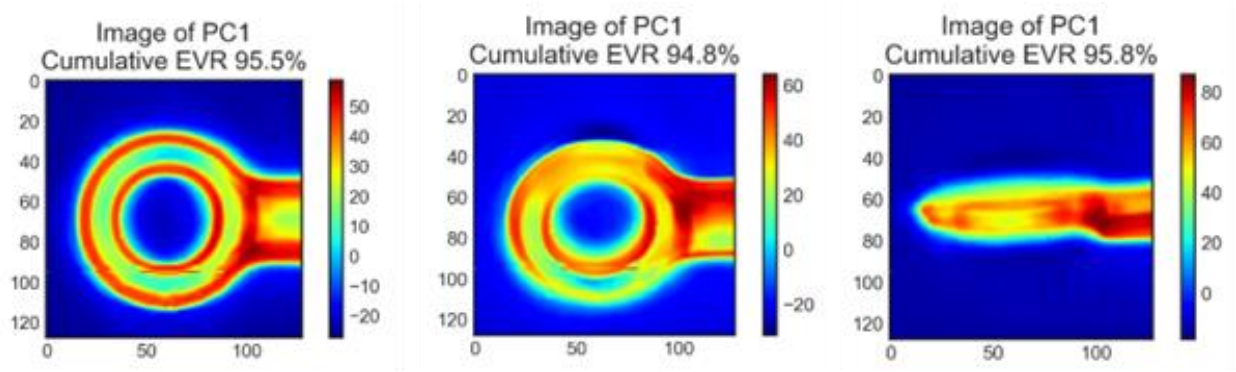


***Figure 26**: A comparison of PC1 of three projections of the FTIR microloop data.*

6. Positioning Correction with Deep Learning based Solution

Lab Automation and the Software Automated Realtime hardware positioning Correction (SARC) routine implemented previously in this study, made it possible to do two things:

1. Run automated, unattended FTIR-MS-Tomography imaging experiments. These experiments could last from anywhere between several hours to even a day.
2. Keep the object of interest within the field of view of 140x140 microns.

However, we found that the microloop holder was not steadily rotating about a rotation-axis passing through its center in all the projections. This made the sinograms exhibit some jitters instead of being smooth (see **Figure 27**).

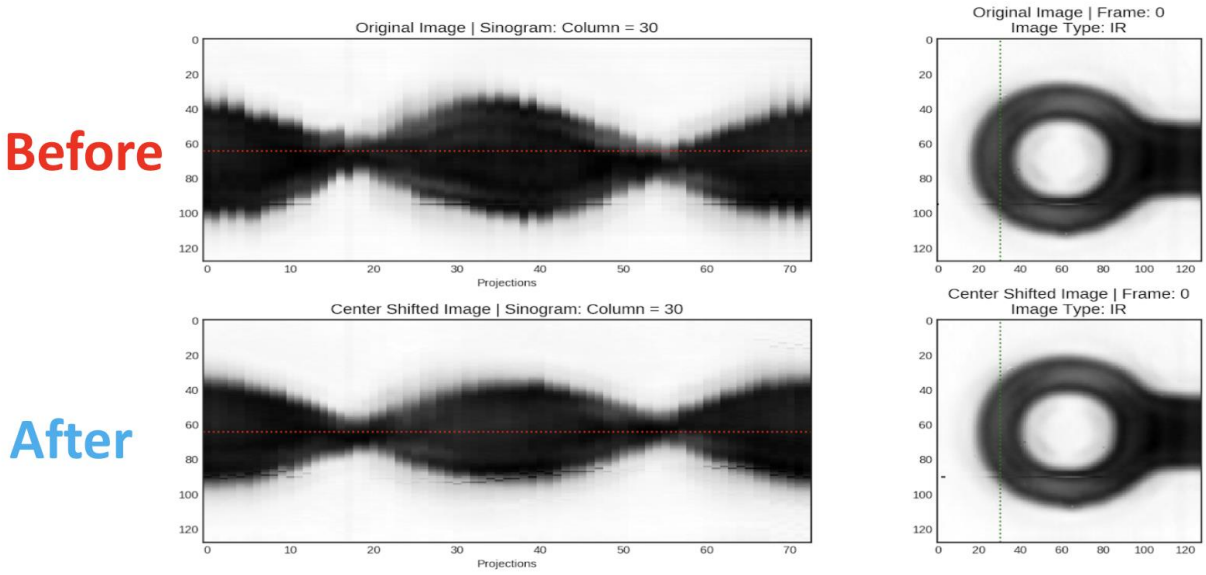


***Figure 27**: Sinogram correction with before and after correction perspective.*

Manual identification of the microloop centers followed by aligning the images so that their microloop centers coincide, provided support and proof for the hypothesis that this method has merit in smoothening the sinograms. Smooth sinograms are necessary for downstream 3D reconstruction of the sample with computed tomography (CT). Although application of CT is

outside the scope of this study, it is useful to mention why the *smooth-sinogram* correction is a necessary evaluation criterion before the FTIR-dataset could be used in CT.

6.1. Center Detection Algorithm: Heuristic (CDAH)

We first developed a heuristic algorithm that takes in a 128x128x770 dimensional FTIR-MS image and then calculates its wavenumber-filtered binarized image of dimension 128x128, followed by an algorithm that attempts to fit an ellipse to the outer edge of the microloop.

When applied to the microloop only situation (without any sample mounted), the heuristic algorithm was able to predict the microloop center reasonably well for 92% cases in the 73 projection images (spread 5 degrees apart).

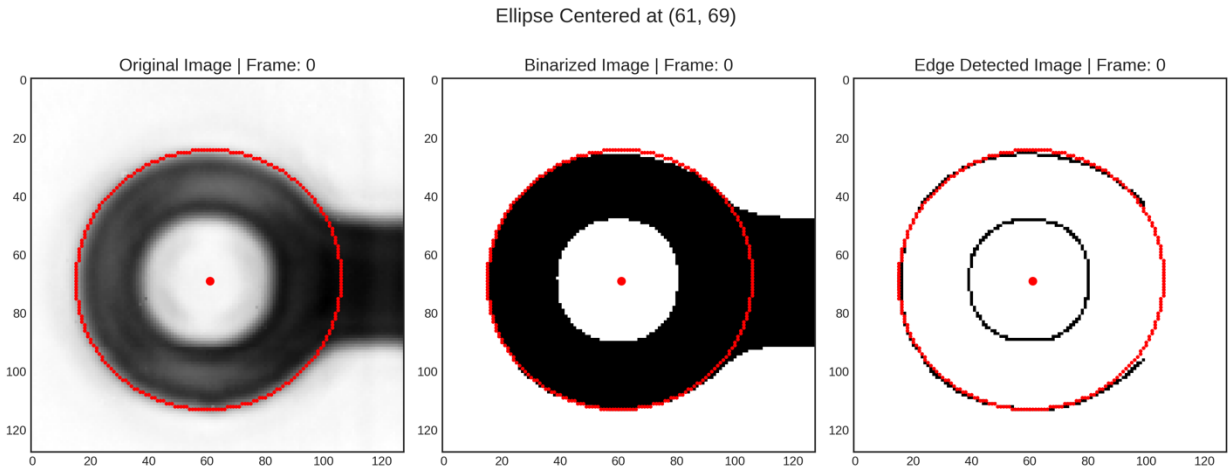


***Figure 28**: Center detection with heuristic algorithm that identifies the oval outline for Frame 0.*

Here we see that **Figure 28** (frame: 0) and the next two figures (**Figure 29** and **Figure 30**) for frames: 5, 9 respectively, show a reasonably well prediction of the location of the microloop center.

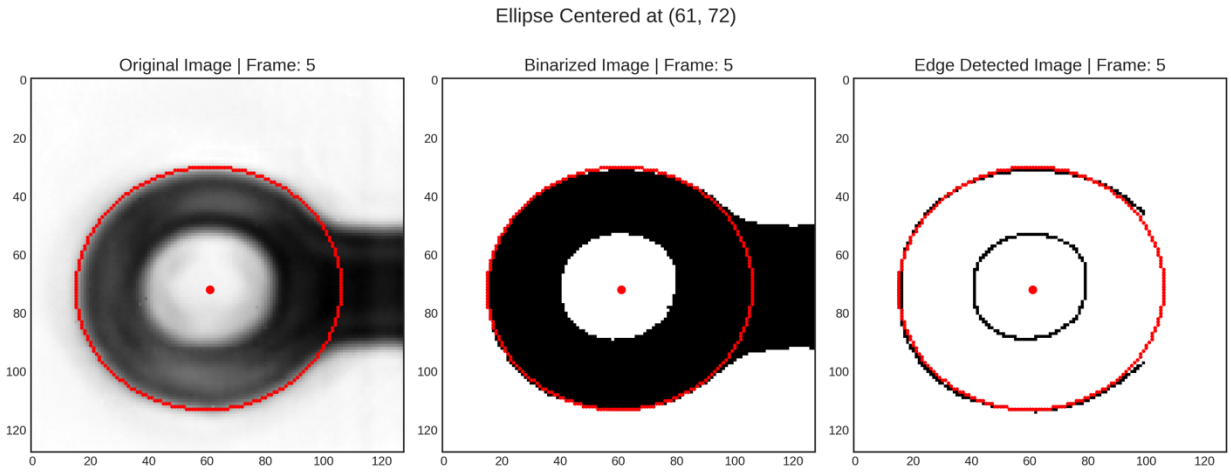


***Figure 29**: Center detection with heuristic algorithm that identifies the oval outline for Frame 5.*

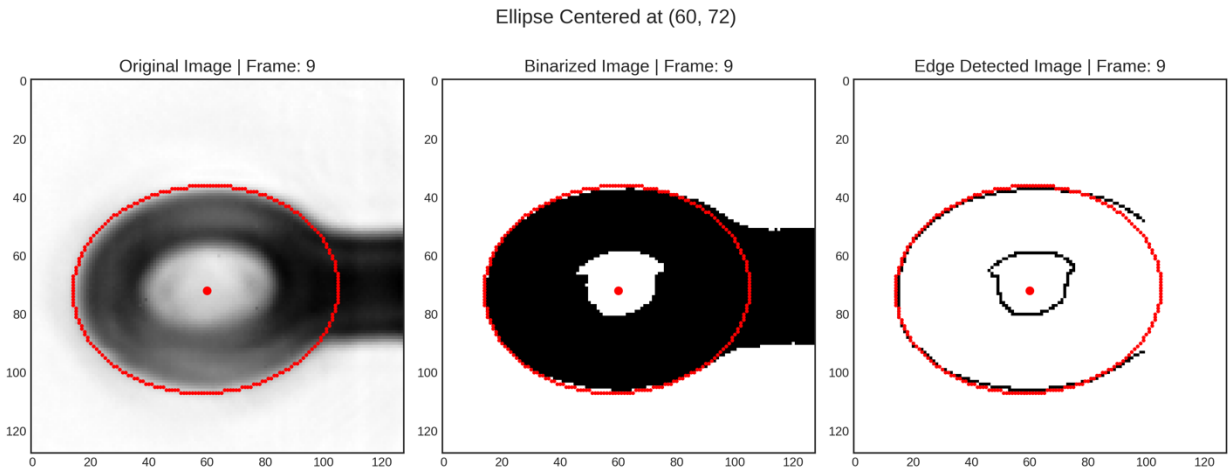


***Figure 30**: Center detection with heuristic algorithm that identifies the oval outline for Frame 9.*

The next two figures (**Figure 31** and **Figure 32**) show that even when the microloop is turned very close to 90 degrees, sometimes the center detection could work.

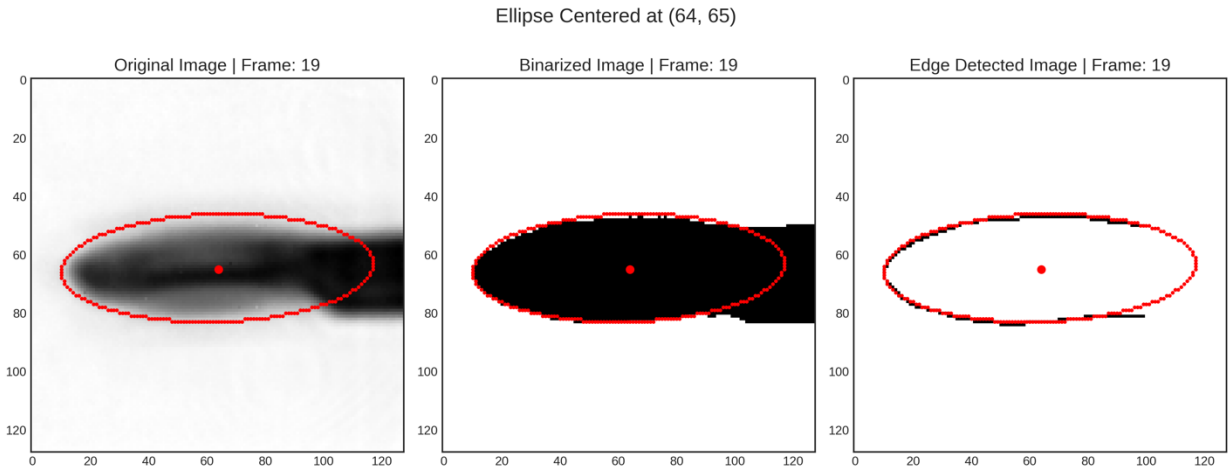


***Figure 31**: Center detection with heuristic algorithm that identifies the oval outline for Frame 19.*

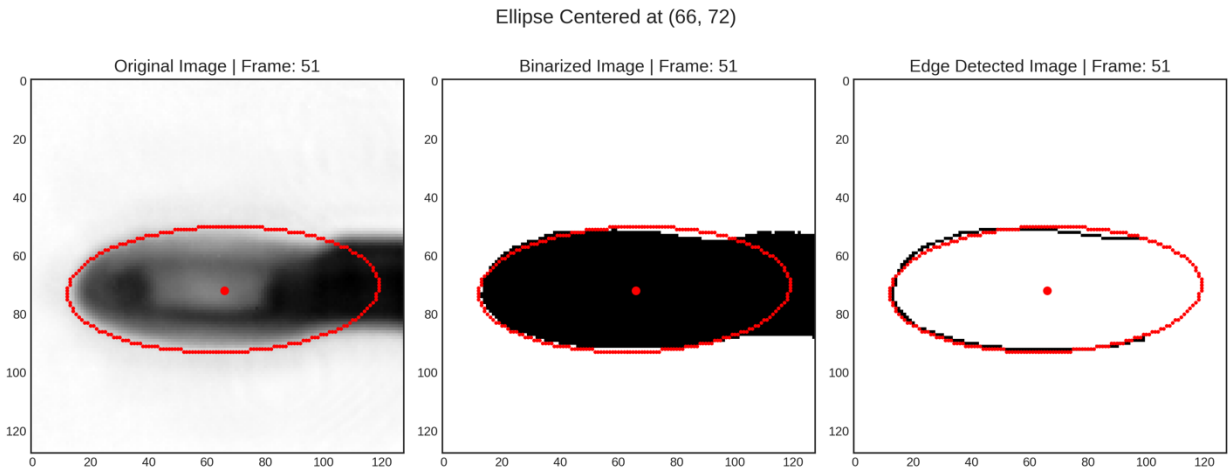


***Figure 32**: Center detection with heuristic algorithm that identifies the oval outline for Frame 51.*

But the last figure below (**Figure 33**) shows that in cases, where the projection angle in close to 90 or 270 degrees the center prediction may not work well. Thus, this is a good first step. However, it is clearly not enough to reliably get the microloop center alone.

Next, we will look into what improvements we can apply to identify the microloop center. Here, we would like to point out that this heuristic solution of fitting an ellipse works only when the image has the microloop alone and nothing else. However, in reality, the microloop will have some sample mounted on top of it (see **Figure 34**). This will break the symmetry of the microloop and in the absence of the circular outer-edge the heuristic algorithm will not work.

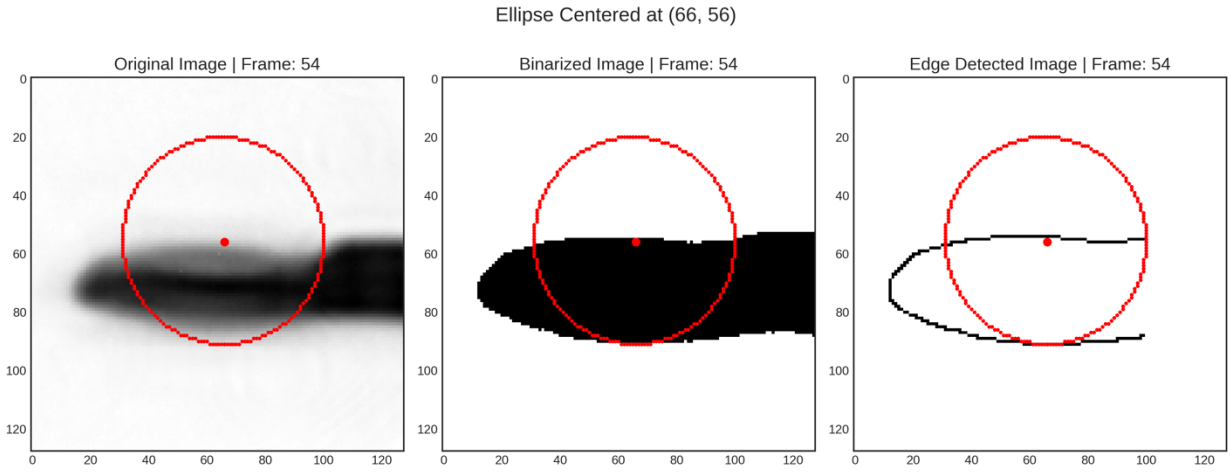


***Figure 33**: Center detection with heuristic algorithm that identifies the oval outline for Frame 54.*

6.2. Microloop Dimensions

We used MiTeGen [28] microloops and micromeshes in the experiments. The original microloop, most often used in the lab during the measurements, had an inner aperture radius ($\boldsymbol{r}$) of 50 microns, the breadth ($\boldsymbol{d}$) of the loop was 20 microns, and the thickness ($\boldsymbol{t}$) of the microloop was about 10 microns. The breadth-to-thickness ratio of the loop section was 2:1.

6.3. Synthetic Data Preparation (SDP)

Computer Aided Design (CAD) modelling was used to prepare 3D parametric models of the microloop holder and a 3D asymmetric sample (**Figure 34**).

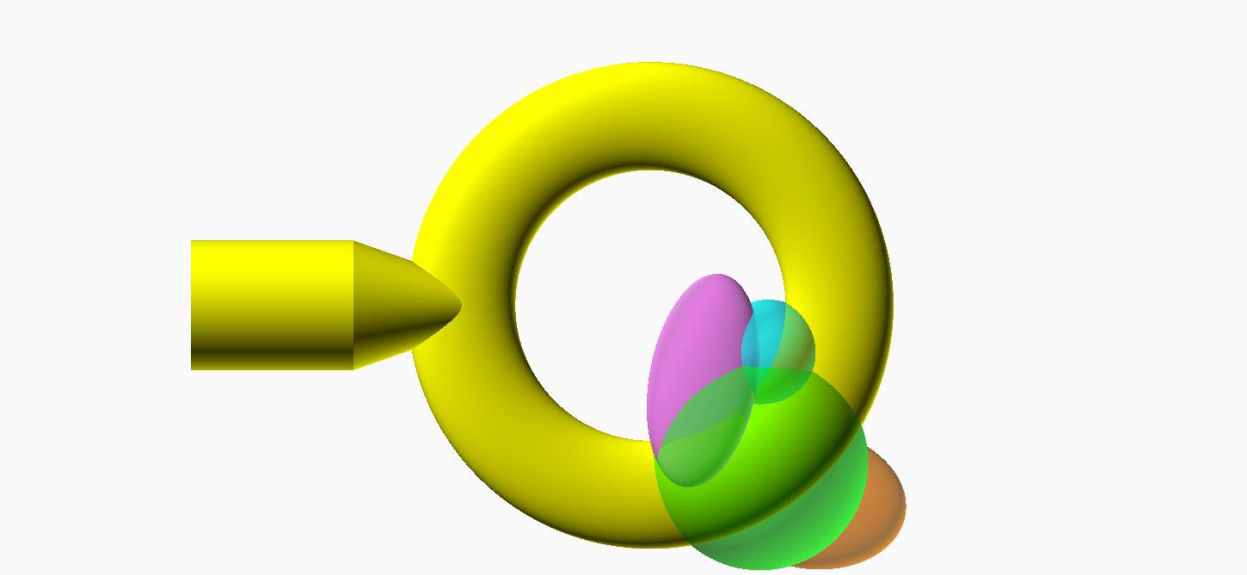


Figure 34: *Synthetic 3D Model of Microloop Holder and a Composite Sample*

Figure 35 shows a synthetically designed microloop holder. It depicts two of the three shape parameters (aperture radius and breadth) for the microloop's 3D CAD model. The thickness parameter of the microloop is the thickness of the loop section along the direction perpendicular to the plane of the image.

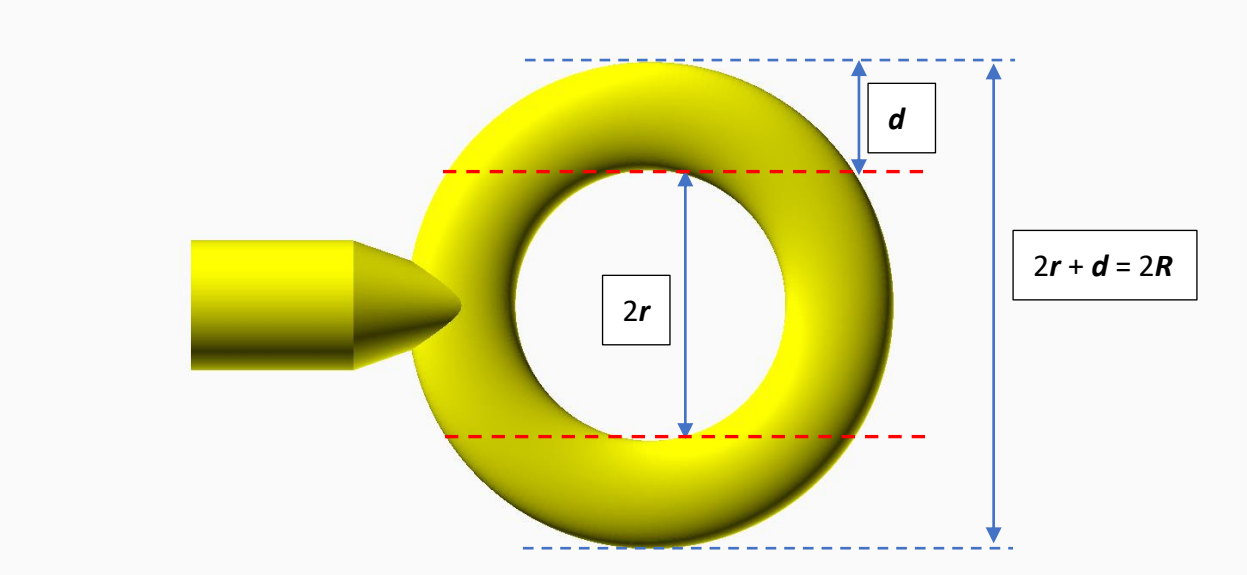


Figure 35: *Synthetic microloop schematic diagram.*

A synthetic sample was designed to have an asymmetric shape. However, to facilitate quick parametric modification of the sample's shape and structure, its constituent parts were chosen to be simple symmetric primitives (sphere, ellipse, cylinder, cube, etc.), generally used in CAD design. **Figure 36** shows a composite synthetic sample. The various substructures contributed by other smaller shapes have been painted with different colors to enhance visual clarity.

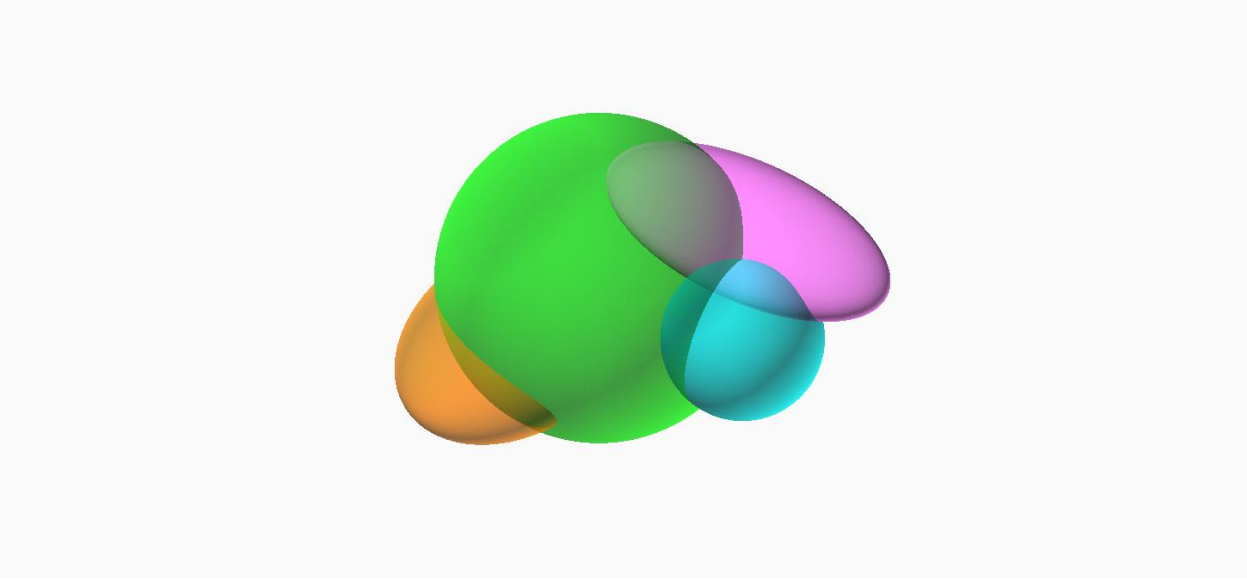


Figure 36: *Synthetic 3D model of a composite sample.*

The composite sample is made of simpler and symmetric 3D primitives. The composite model is designed to have asymmetry in its 3D shape, to create synthetic dataset that offers greater generalization.

6.3.1. Method: SDP

A parametrized 3D microloop model was created using OpenSCAD 3D CAD modeler [29]. The default values of the parameters for the 3D CAD model of the microloop were chosen to closely mimic the observed ratios of aperture-radius, loop-breadth and loop-thickness for the physical specimen of the 50 micron-aperture microloop from MiTeGen. However, the benefit of parameterization makes it possible to create other variations of the dimensions that define the microloop.

Similarly, a 3D parametrized composite sample (non-holder-material: NHM) was created using OpenSCAD [29] as a placeholder sample to place on the 3D model of the microloop. Synthetic projection.

6.3.1.1. 3D CAD Model Design: Degree of Freedom Considered

We considered the following set of changes in various dimensions of the assembly: microloop + sample.

The choice of using simulated data, comes with its own set of limitations and potential pitfalls. Simulated data could suffer from under representation of the possible dimensions in the data, unless taken care of. At the same time, trying to incorporate many degrees of freedom (**dof**) in the simulated data could explode into a problem of itself; and pose a new challenge – the curse of dimensionality. Thus, with the intension of keeping the data-generation space tractable and time-budged, we took a practical approach at allowing selectively some degrees of freedom to the simulated data.

- **Dof: Holder (Stick + Loop)**
    - scale
    - translation
    - rotation
- **Dof: Holder-Stick**
    - scale
    - length
    - shape
    - rotation (relative to the Holder-Loop)
    - translation (relative to the Holder-Loop)
- **Dof: Holder-Loop**
    - scale
    - radius of annular section

- thickness of annular section (perpendicular to the radial direction)
- rotation
- translation

- **Dof: Non-Holder Material (NHM: Sample)**
  - scale
  - shape
  - position of the loop
  - rotation
  - composition (could be made of multiple smaller shapes)
  - multiplicity (how many types and instances of samples to use on the same loop)
- **Dof: Projection**
  - projection angles
  - equal spacing: multiple values for angle-increment
  - inequal spacing: randomized behavior (reproducible: seeded random number generator)
  - motion of assembly (Holder + Sample)
  - in-plane rotation
  - in-plane translation

6.3.1.2. The Option of using Data Augmentation

It is a standard technique to often use data-augmentation for deep learning models, when the variability in the degrees-of-freedom in the training dataset is not sufficient or rather limited. There are various python libraries that offer data-augmentation capabilities for images torchvision [30], albumentations [31], kornia [32], tormentor, stainlib, etc. We used albumentations for image-data-augmentation purposes.

The choice of applying image augmentations also alleviates the burden of having to consider a large space of holder and sample assemblies with variations introduced in all the above mentioned dofs.

6.3.1.3. Why synthetic data?

From the previous steps as presented in this study, we knew that if we can somehow recover the specific pixels that belonged to the microloop in the projection images, it will allow us to predict the center of the microloop and then the projection images could be realigned.

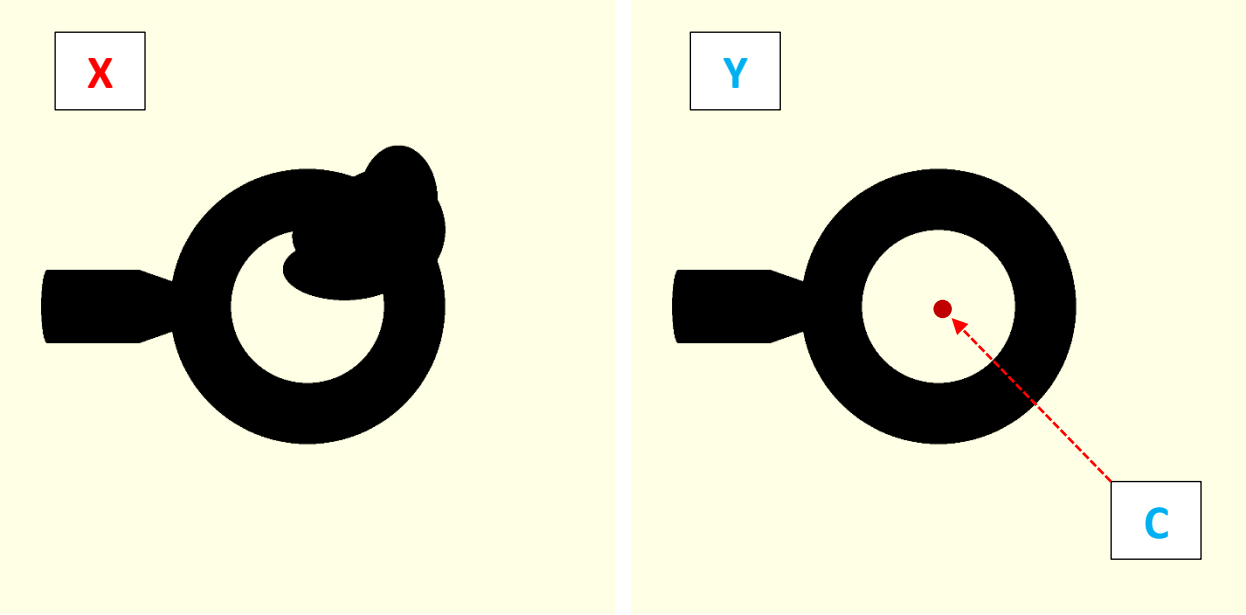


Figure 37: *Why Synthetic Data?*

Given a binarized input image (X), we need to extract the binarized output image (Y), with the center (C). A Deep Learning segmentation model could be used here to "learn" the representational space mapping between the input and expected out spaces.

A deep learning image segmentation model could be trained given we have sufficient amount of training data with meaningful variations for the model to learn from and generalize based on. But the challenge was that we did not have enough data to train a deep learning model. If we had the measured data from the lab, then the very problem we are trying to solve does not exist anymore. And if we did not have the data, then we could not train the DL model. There was a research paper that came out in 2021 by Microsoft researchers: “Fake it till you make it” [33], which also circumvented the problem of non-availability of rich data that could be used to train a DL model and identify features of faces. The researchers in [33] used synthetically generated human faces to

introduce a rich set of diversity and variations. Our research was also at a similar juncture, where we knew what we need, but did not have the data to train a DL model.

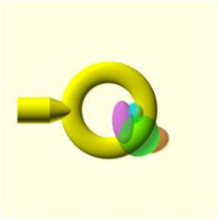
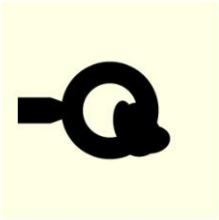
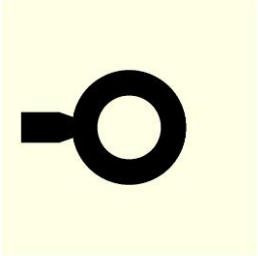
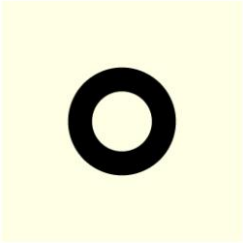
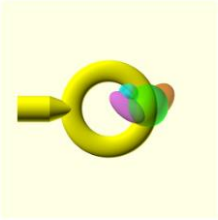
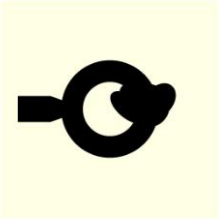
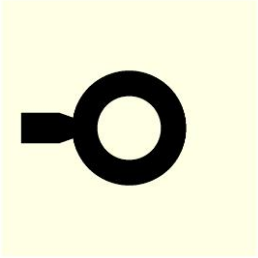
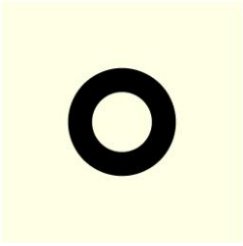
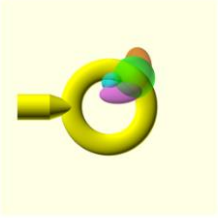
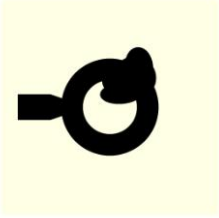
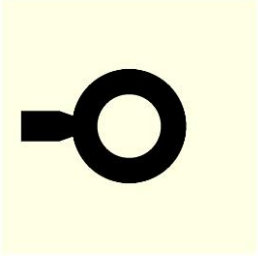
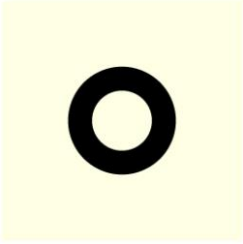
Location of Sample	Color: Holder + Sample	Black: Holder + Sample	Black: HolderOnly	Black: LoopOnly
01:30				
12:00				
10:30				

***Figure 38**: Synthetic data preparation modality.*

Figure 38 shows from left to right the colored assembly image, all-black assembly image, all-black holder-only (no sample) and all-black loop-only (no-holder-stick and no-sample) images.

Additionally, we have also moved the sample on the holder. Three locations are labelled with clock-positions (e.g., 12 o'clock), based on where the sample is placed with regard to the microloop center.

6.3.2. Classifying Projections as Easy and Hard

We have seen previously that the greater the area of the microloop's projection, the more likely it is to accurately predict for the microloop's shape of the loop and the pixels it occupies. Thus, while, for some of the projections (closer to 0 or 180 degrees), it is relatively easy to predict the microloop-occupied-pixels, for the other projections (closer to 90 or 279 degrees), it gets harder as we have increasingly diminishing number of pixels occupied by the microloop in those projections closer to 90 or 270 degrees.

We wanted to encode this information as a feature to the model and hence created a feature with two classes:

- **Hard**: projection angles between 35 degrees on either side of 90 and 270 degree.
- **Easy**: any other projection angle.

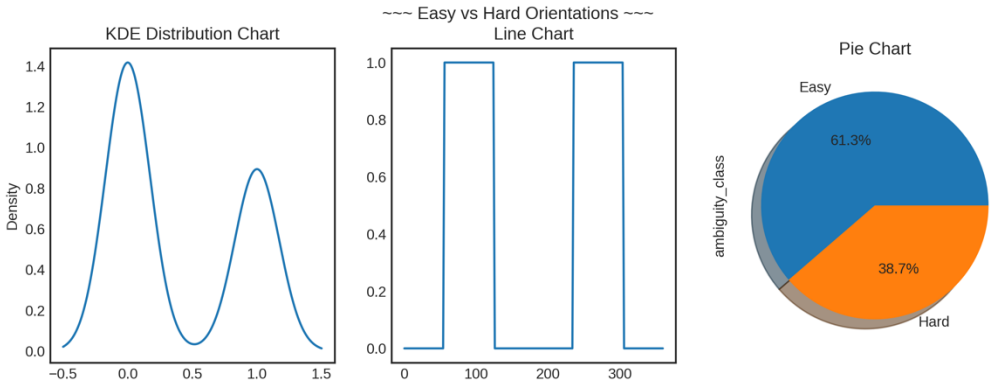


Figure 39: *Projection prediction difficulty level.*

The figure above (**Figure 39**) shows the distribution of 181 images from a single synthetic TomoDataset, each projection taken 2 degrees apart, has been classified as "easy" or "hard" depending on the angle of projection.

Next, we split these projection images into training and test splits (**Figure 40**), while still maintaining the same proportion of easy vs. hard images in the two splits. This will allow for the

model to learn to similar easy/hard distribution in training as to expect in test. This step lets us maintain similar data class-imbalance in across the training and the test datasets. The training dataset has 144 projections, and the test dataset has 37 projections.

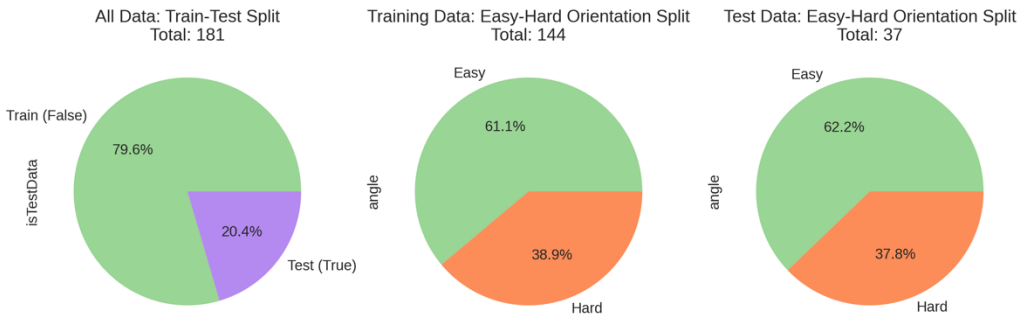


***Figure 40**: The Train-Test split and relative distribution of "easy" and "hard" projection images.*

6.4. UNet Architecture

The UNet architecture was first used in 2015 in a biomedical paper [18] for an image segmentation task. It comprises of two arms:

- An encoder, which performs compression of information from a high dimensional space to a latent space of low dimensional representation.
- A decoder, which performs the opposite of what the encoder does - decompression of information from the low dimensional latent space to the high dimensional output.

The classic UNet uses skip connections between layers. This technique allows it to avoid pitfalls like vanishing gradients.

The following image (**Figure 41**) shows the original UNet architecture from the 2015 paper [18].

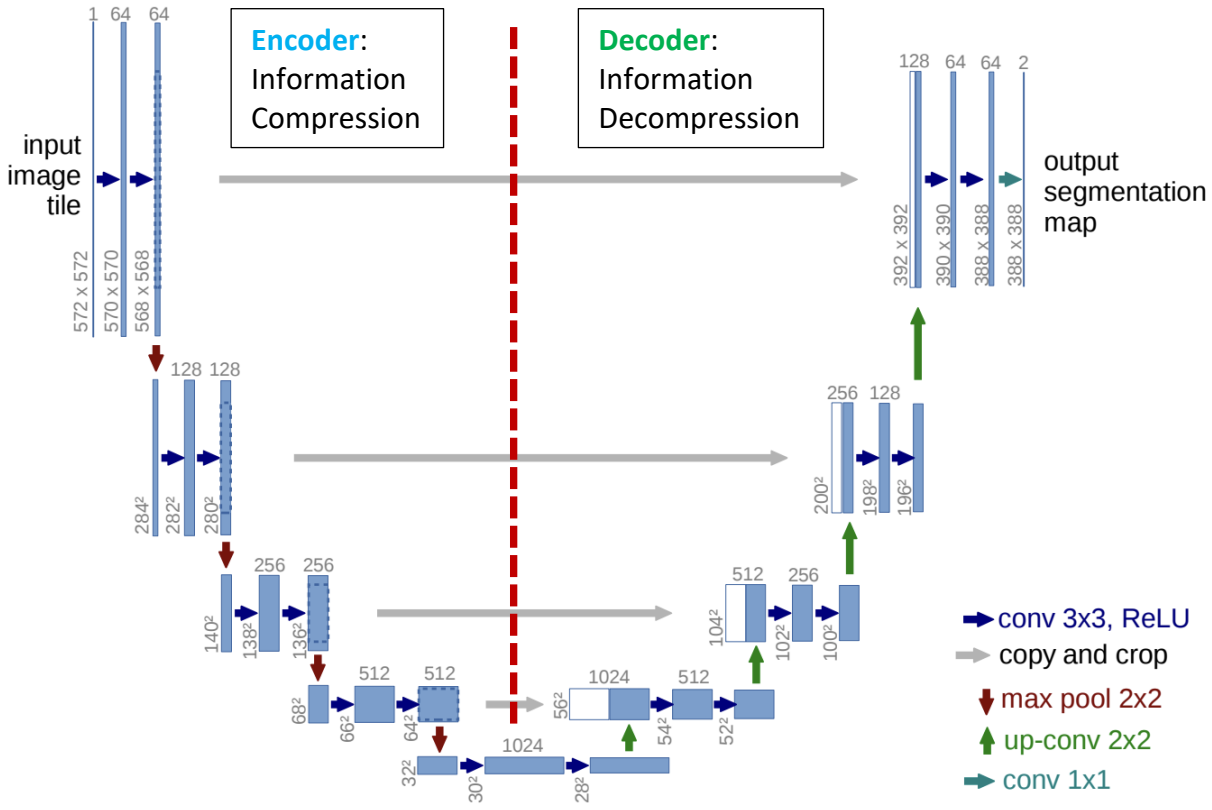


Figure 41: *Original UNet Architecture [18].*

The left half of the UNet architecture compresses information from the input image file and the right half decompresses it to the output segmentation map. The left arm steps down the dimensionality of the input in the 4 down-convolution blocks, that consist of 3x3 convolution, followed by ReLU activations and the compression also involves the 2x2 max-pool layer. The right half does the opposite of the left half and up-convolves the as well as combines the input from the incoming skip-connections.

6.4.1. Modifications to UNet

The role of the left half of the UNet is to compress information and extract the low-dimensional representation. Using the UNet as is, requires us to train the model weights of the entire model. We used a pretrained model instead for the left half. The pretrained model was trained with more generic set of images, from ImageNet dataset. Using its comprehending capability of images, helps training the modified UNet, only training the right half of the network. We chose to use a pretrained ResNet-34 model trained on ImageNet dataset.

We designed our model to accept input images of size: 640x640 and output predicted masks of the same size. The size of the input and output images was chosen based on the fact that the FTIR-MS measurements in Hirschmugl Lab have a dimension of 128x128x770. The binarized image of this single projection dataset will be 128x128. Choosing 640x640 is allowing us an effective magnification of 5x5 = 25 times. This design allows us greater room for prediction errors.

6.5. What is skip-connection?

The concept of skip-connection was used in the famous ResNet (Residual Network) paper [34] by researchers from Microsoft in 2015.

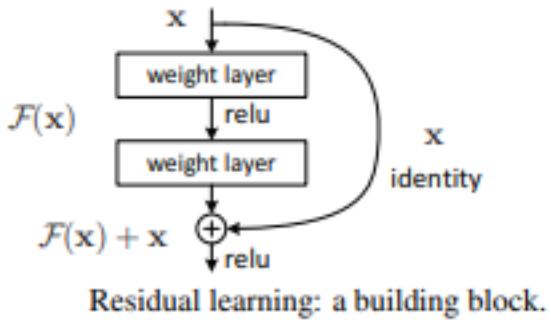


Figure 42: *Skip Connection Illustration.*

Source: ResNet paper [34]

It was shown that the skip-connections allowed using very deep neural networks without the problem of vanishing gradients. The usage of residual networks has now become a standard for models like UNet, where an un-altered signal (***x***) is added to the output of a subsequently altered signal, ***F(x)***, via passing through some layers: ***G(x) = F(x) + x***. The benefit is that even if the gradients calculated for the ***F(x)*** part become very miniscule during the backward-pass, the other part (***x***) lets gradient calculations for earlier layers possible.

6.6. Results and Discussions

6.6.1. Microloop Mask Prediction

The figure below (**Figure 43**) shows the results of microloop mask prediction, given the input image had both the microloop and the sample. The left column is the input image. The middle column is the ground truth (GT) mask for the microloop alone. The third column from the left has the DL model predict the likelihood that a pixel for belongs to the microloop.

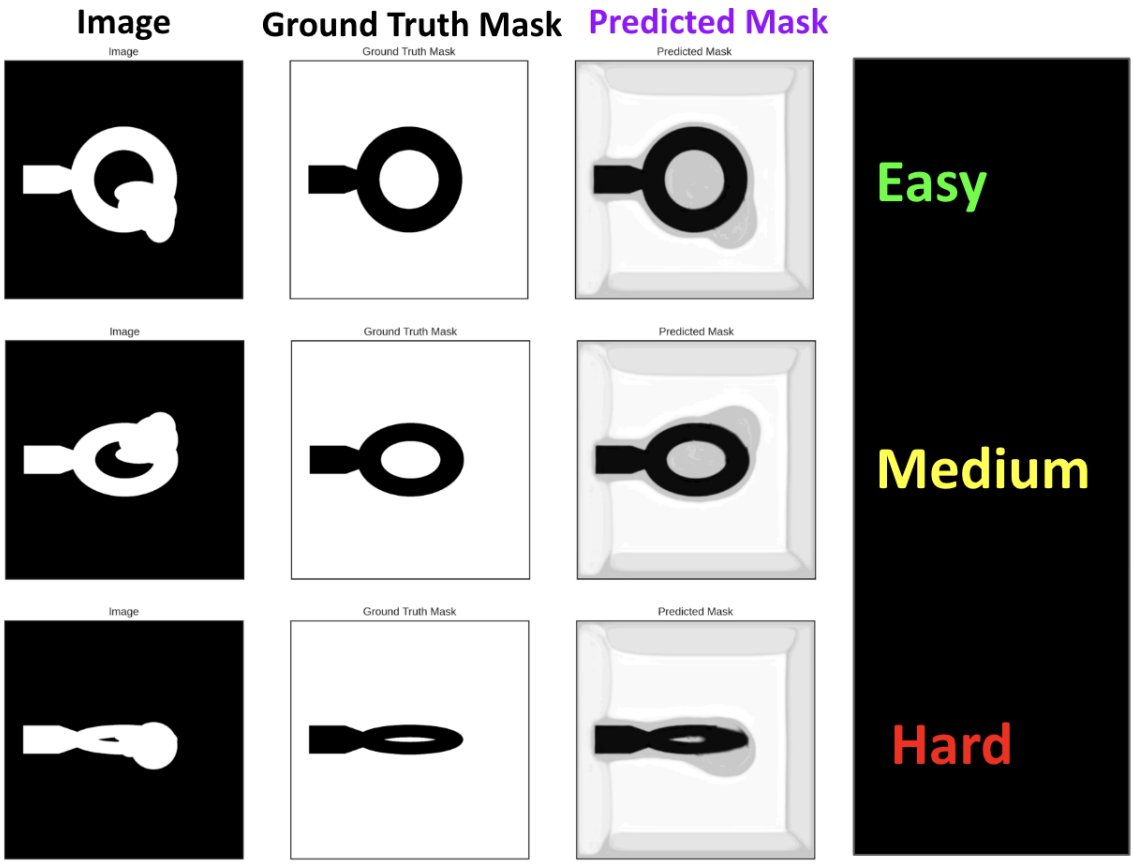


***Figure 43**: A comparison of the predicted and ground truth masks for the synthetic data.*

The dark pixels signify a very high prediction probability (>=95%) for a pixel classified as a part of the microloop. The other pixels that are not as dark as the obvious pixels from the microloop, belong to the sample or are artifacts of data-augmentation during training. However, since we are only concerned with the pixels from the microloop, we can disregard the gray pixels in the

predicted image. Another interesting aspect of this model prediction that comes up is that while for “Easy” and “Medium” difficulty images, the model does a descent job at predicting the microloop pixels, the “Hard” ones seem to give the model some challenge. Here the “Medium” difficulty is essentially a part of the “Easy” category, where the angle of projection is farther than 35 degrees from 0 or 180 degrees. It has been introduced here solely for purpose of presentation of gradual increase in the amount of difficulty of prediction. We have not used any “Medium” category during training the model.

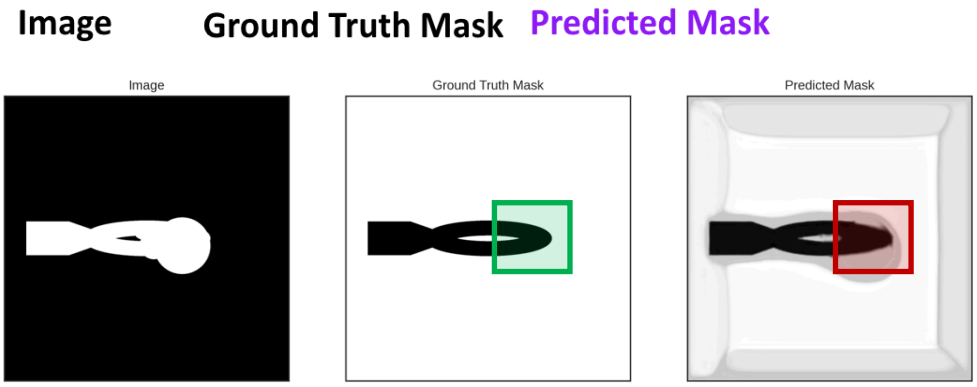


Figure 44: An example of holder recovery when original image is severely occluded by sample.

Figure 44 is an enlarged version of the “Hard” scenario from the previous figure. We can clearly see that the front half of the microloop was not accurately reproduced from the input image. It must be noted that the inflated degree of difficulty emerges from two separate sources:

- Angle of projection is close to 90 or 270 degrees, reducing the effective cross-section of the microloop in the image; less information leads to greater uncertainty.
- The sample occupies almost the entire front half of the microloop. Thus, the DL model effectively has very little information about the front half of the microloop in the input image.

So, while these two sources compound the degree of difficulty for the DL model, it does a decent job at recovering the microloop pixels reasonably.

The current deep learning model was applied to the test dataset with 37 synthetically generated projections spread across 0 to 360 degrees (as previously mentioned in section 6.3.2).

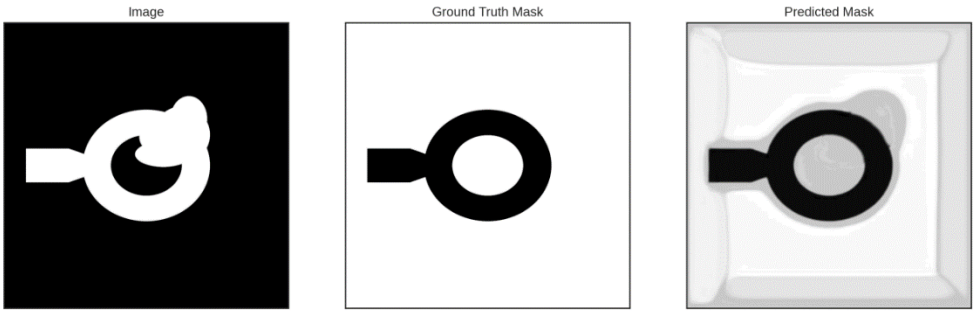


***Figure 45**: Deep learning model prediction for 0 deg projection*

Figure 45 shows (from left to right) the original image of the holder and the sample as an assembly, the ground truth mask of the holder-only pixels, and the deep learning model predicted pixel-specific likelihood of the pixels that belong only to the ground truth image.

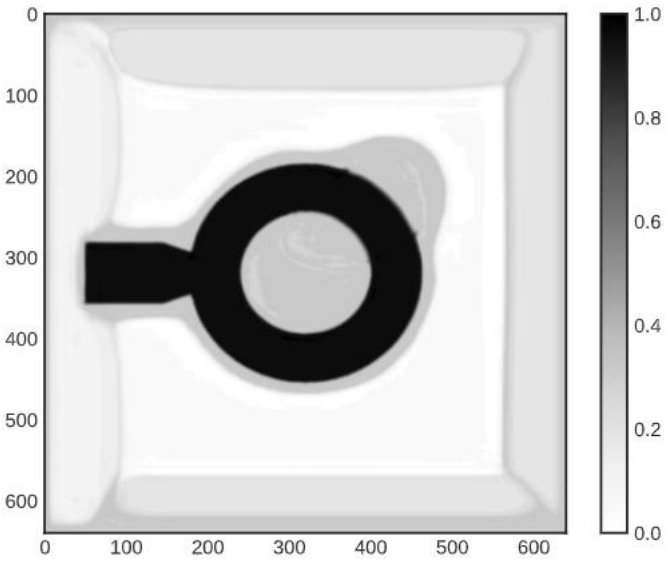


***Figure 46**: Intensity scaled predicted image for 0 deg projection*

An intensity scaled predicted image (**Figure 46**) shows that pixels with highest values could return a mask for the holder-only pixels. Prima facie it appears that a value of 0.95 could work as a good threshold for extracting the predicted mask for the target pixels.

The following figures (**Figure 47**, **Figure 48**, **Figure 49**, **Figure 50**, **Figure 51**) show how changing the threshold from 0.95 to 0.75 with decrements of 0.05 impacts the quality of prediction, when measured using Intersection over Union (IoU). Theoretically, IoU ranges between 0 and 1. The higher the value of IoU, the more accurate is the prediction.

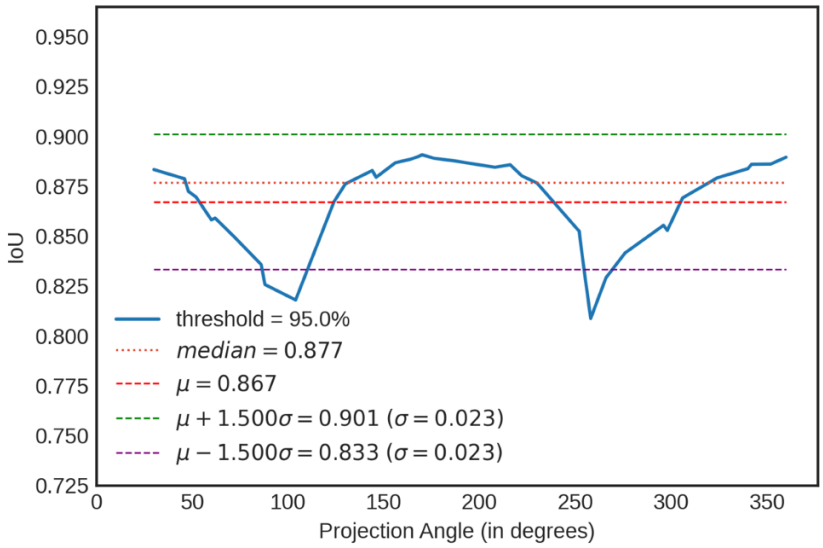


***Figure 47**: Predicted holder mask IoU for test dataset with threshold = 0.95*

The prediction with threshold value of 0.90 (Figure 48) shows an improvement for all the projections as opposed to what we saw for threshold of 0.95 (Figure 47).

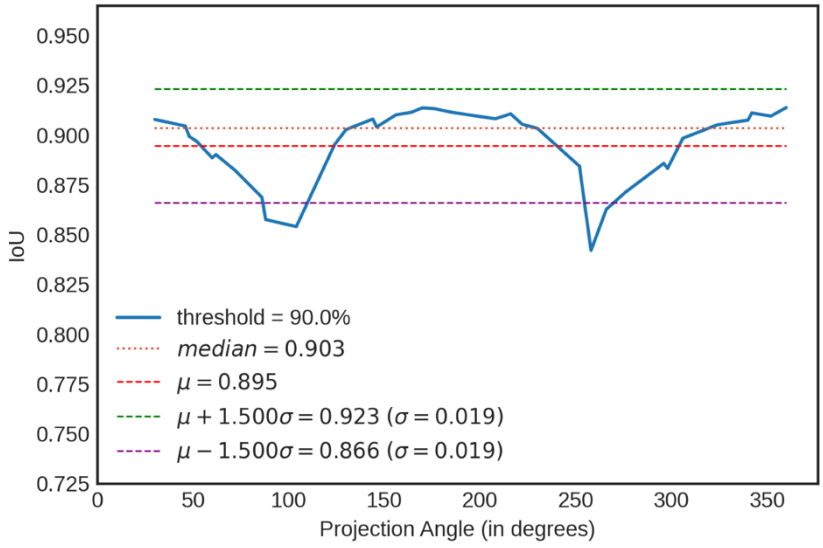


***Figure 48**: Predicted holder mask IoU for test dataset with threshold = 0.90*

Similarly, we see that threshold of 0.85 (Figure 49) yields better result for IoU when compared to thresholds of higher values (0.90 and 0.95).

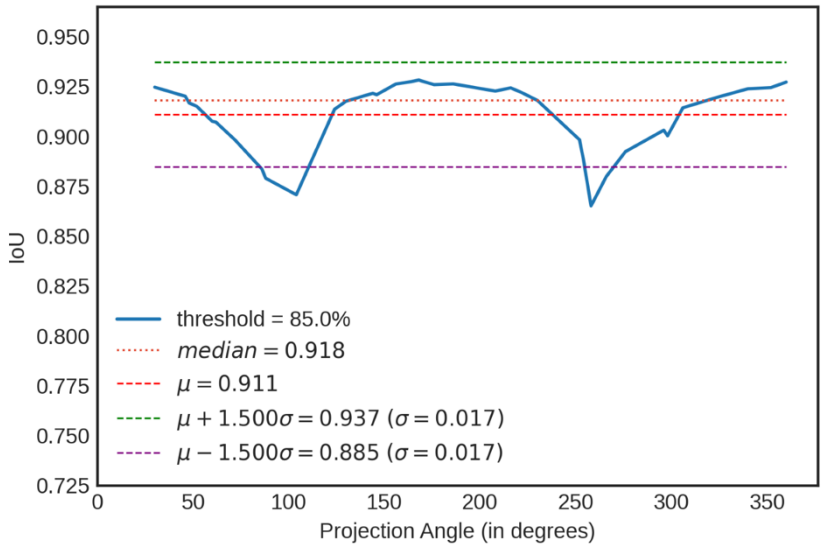


***Figure 49**: Predicted holder mask IoU for test dataset with threshold = 0.85*

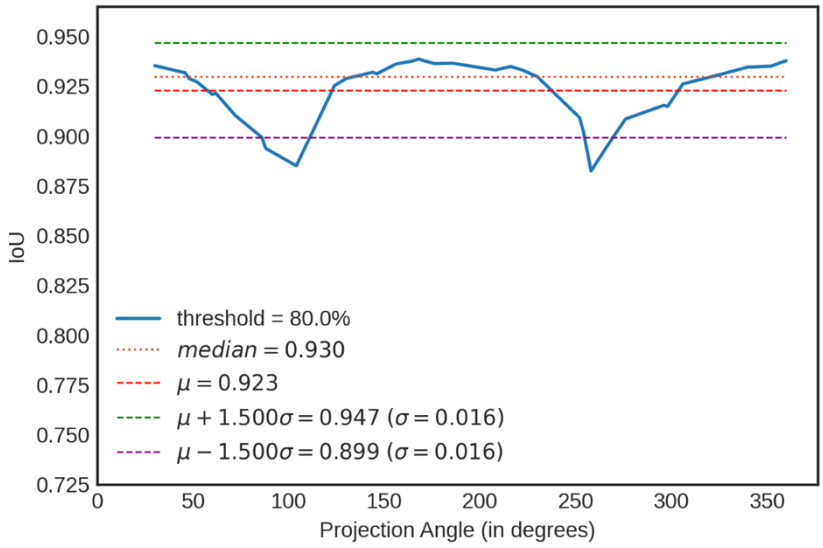


***Figure 50**: Predicted holder mask IoU for test dataset with threshold = 0.80*

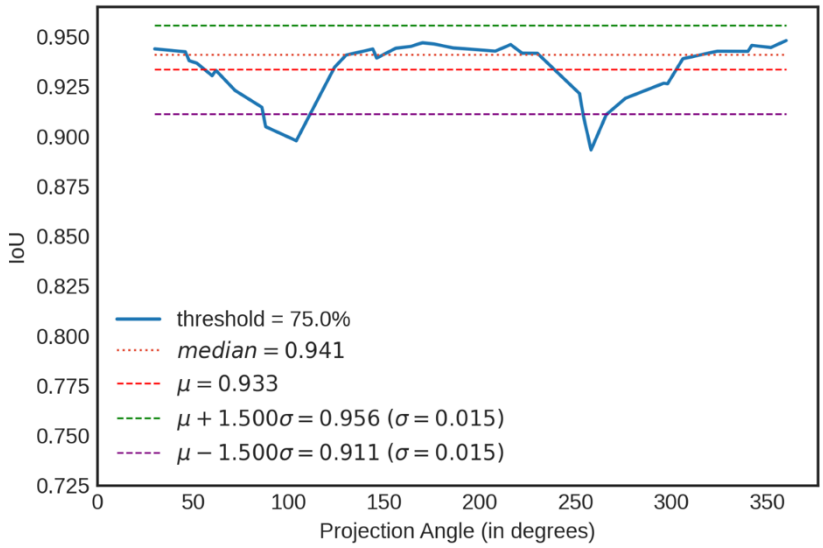


***Figure 51**: Predicted holder mask IoU for test dataset with threshold = 0.75*

It is quite evident that choosing a threshold of 0.75 on the DL model predicted image yields better prediction across the board than the best prediction with a threshold of 0.95 (see **Figure 47** and **Figure 51**). Also, this behavior is consistent as we move from 0.95 to 0.75 threshold. The drop in performance near the angles 90 and 270 degrees is expected as the signal cross-section (from the original image) drops for those angles of projections.

6.6.2. Future Scope of Possible Improvements

While the deep learning model does identify the microloop reasonably well, we saw that the model still has some limitations under certain scenarios:

- when angles close to 90 or 270 degrees and
- when the sample covers up a significant portion of the microloop.

6.6.2.1. How can we possibly improve the microloop center detection algorithm?

Although the heuristic method we discussed earlier, may not be useful anymore, the learning from the exercise could still be relevant. Given the binarized image of the sample and the microloop holder (we will refer to this as the **holder** from now on), we are interested in the following four tasks:

- **T1:** Predict the pixels that belong to the **holder**.
- **T2:** Predict the pixels that belong to the **loop** section of the holder.
- **T3:** Predict the **center** of the **loop**. (**Most Important**)
- **T4:** Predict the **outer edge** of the **loop**.

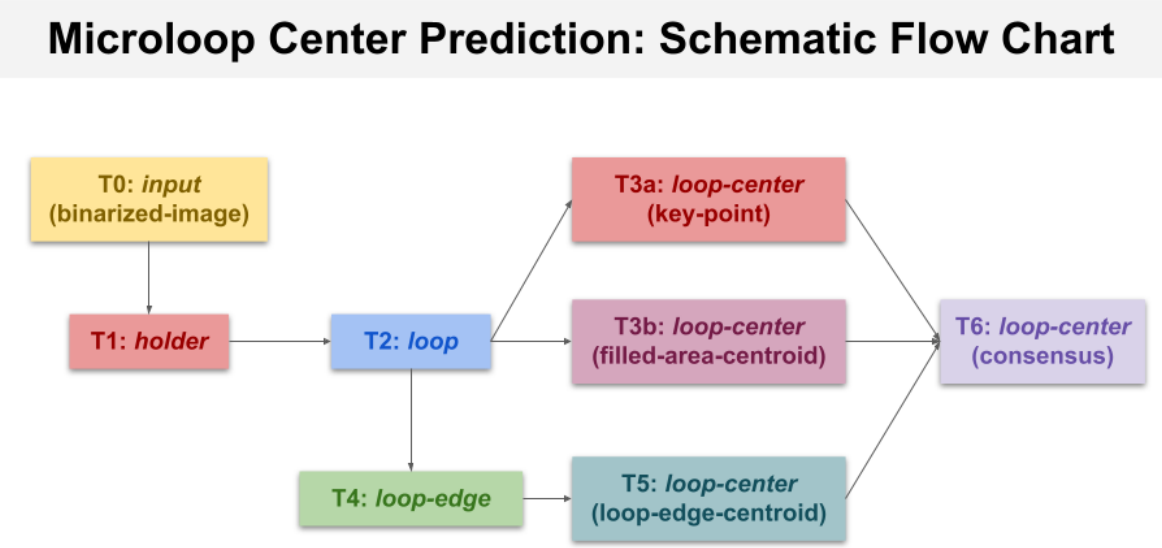


***Figure 52**: Prescribed schematic flow chart for future improvement of the microloop center prediction algorithm.*

Here we present a possible direction (**Figure 52**) of how predicting the microloop may not be robust enough. In the proposed pipeline above (**Figure 52**), the input image is first binarized (**T0**)

and then passed to a deep learning (DL) model that predicts the pixels with the holder (**T1**). Next the holder predicted binarized image is passed onto another model to predict the loop (**T2**). The loop-predicted binarized image is then fed to steps **T3a**, **T3b** and **T4**. While T3a comes from the same model that predicted T2, T3b is achieved by passing on T2's output to another DL model to produce a filled area with the loop. For T4, we use the heuristic method to determine the loop-edge (contour-detection) and at **T5** we evaluate the centroid of the loop-edge. Finally, we combine the three inputs T3a, T3b and T4 to arrive at the position for loop-center (**T6**).

This *consensus*-based approach makes sure to use the strengths of all three prediction mechanisms (T2-T3a-T6, T2-T3b-T6, T2-T4-T5-T6) and combines them to predict the center of the microloop.

6.6.3. Summary

The heuristic model worked well in about 90 percent of the cases when applied to 72 projection images (each 5 degrees apart) and predicted the microloop centers reasonably well as compared to manual detection of the centers. While it was clear that the heuristic model had limitations when applied to projections closer to 90 and 270 degrees, it undoubtedly offered a major advantage over the manual detection of the centers.

The manual detection of the 72 projections took around 2 days, and still it was not entirely accurate for the hard images (closer to 90 and 270) degree angles. So, despite its limitations on accuracy to about 10% images, the heuristic method was a step in the positive direction.

However, another fundamental limitation of the heuristic method was that it needed images of microloop holder alone and no sample with it. The heuristic method works only if the oval or circular surface geometry of the microloop is visible and not occluded by a sample mounted on

top of the holder. Thus, it would have been a serious impediment to applying even any automated heuristic model for image (sinogram) alignment if the images contained a sample in them.

The Deep Learning (DL) approach offers a solution here by restoring only the pixels of the microloop from the input images of microloop and the sample. Once the microloop only image is obtained, the heuristic model can be applied on the output of the DL model. The DL path thus helps in navigating around the input-limitation of the heuristic model.

However, it must be noted here that none of this addresses the prediction limitation we observed for the heuristic model in about 10 percent of the projections.

The previous section discusses how the DL solution could be improved by adopting a hybrid consensus model that combines:

(a) the predictions of purely heuristic method

(b) partly heuristic and partly deep learning driven methods

(c) wholly DL method.

Thus, in the current state of the models (heuristic and DL), we have gained some automated capabilities of predicting the microloop centers and predicting the pixels that solely belong to the microloop. This could be used to save time in labeling the microloop centers, via automated center-detection assistance to a human user looking forward to identifying the microloop centers in the IR projections. While the manual method of identifying centers could take 2 days for 72 projections and perhaps almost a week for a complete tomo-dataset of 180 projections, the automated method effectively brings that timeframe down to under an hour. Also, the PCA driven background subtraction transforms the problem from a multilayer (multi-channel) hyperspectral image problem to a single layer (single channel) binarized image problem. Thus, the method of

implementation of the automation also offers modular breakdown of intermediate steps and the intermediate solutions could be used in other use-cases even outside of the current pipeline.

Comparison of applying heuristic method on Ground Truth and Predicted Mask Images

The DL model compliments the limitations of the heuristic method of center detection. The figures **Figure 54**, **Figure 55** and **Figure 56** show the center prediction error when the same heuristic method is applied to the test-dataset, for both the ground truth (GT) image of the microloop holder and the DL model predicted mask of the GT. The relative horizontal error is the difference between the center detected from the GT and the predicted mask. Similarly, we evaluate the relative vertical error. The relative Euclidean distance error also shows that in the worst scenario the maximum error is under 5 pixels for the 640x640 pixels output image from the DL model. If we were to downsize this to 128x128 pixels, the individual errors in x (horizontal) and y (vertical) will be scaled down by a factor of 5. Thus, with the synthetic data, the DL model and heuristic method can keep the error in center prediction to under 1 pixel for an image size of 128x128 pixels.

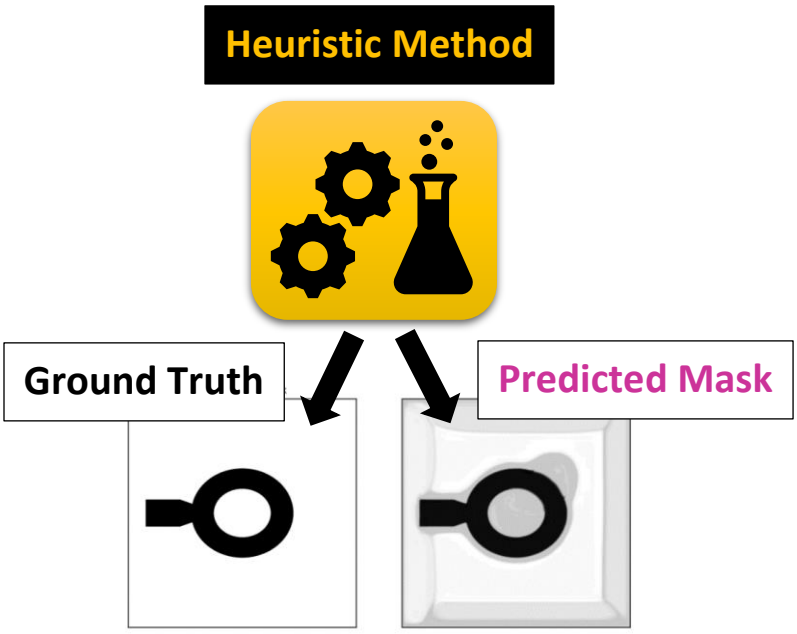


***Figure 53**: Heuristic method is applied to both Ground Truth and Predicted Mask*

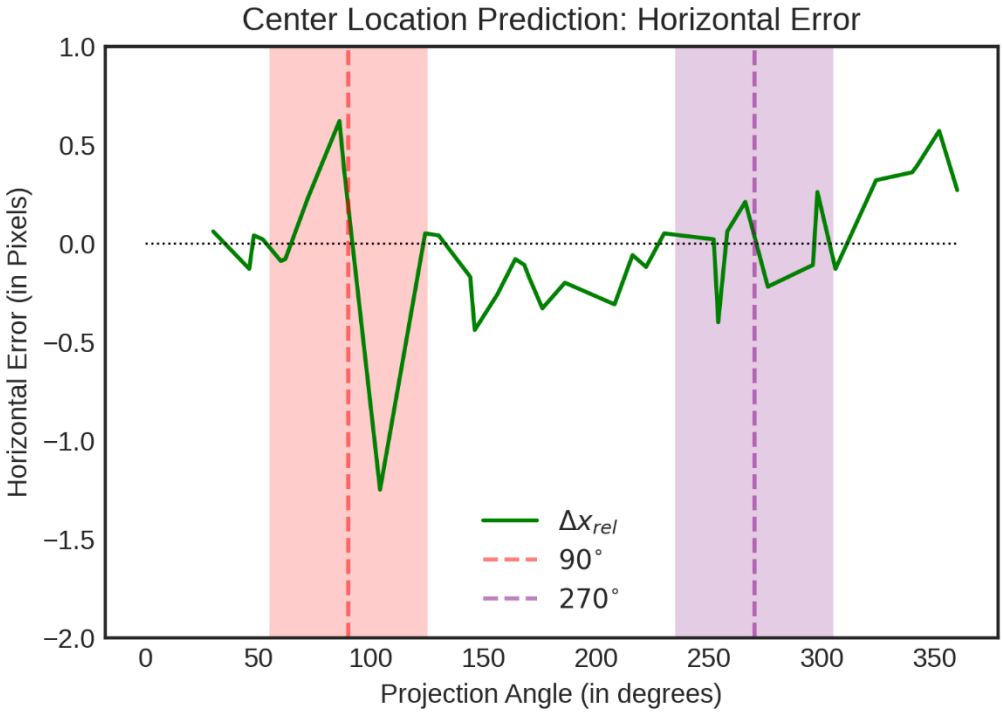


***Figure 54**: Center prediction error in x with DL model + Heuristic Method*

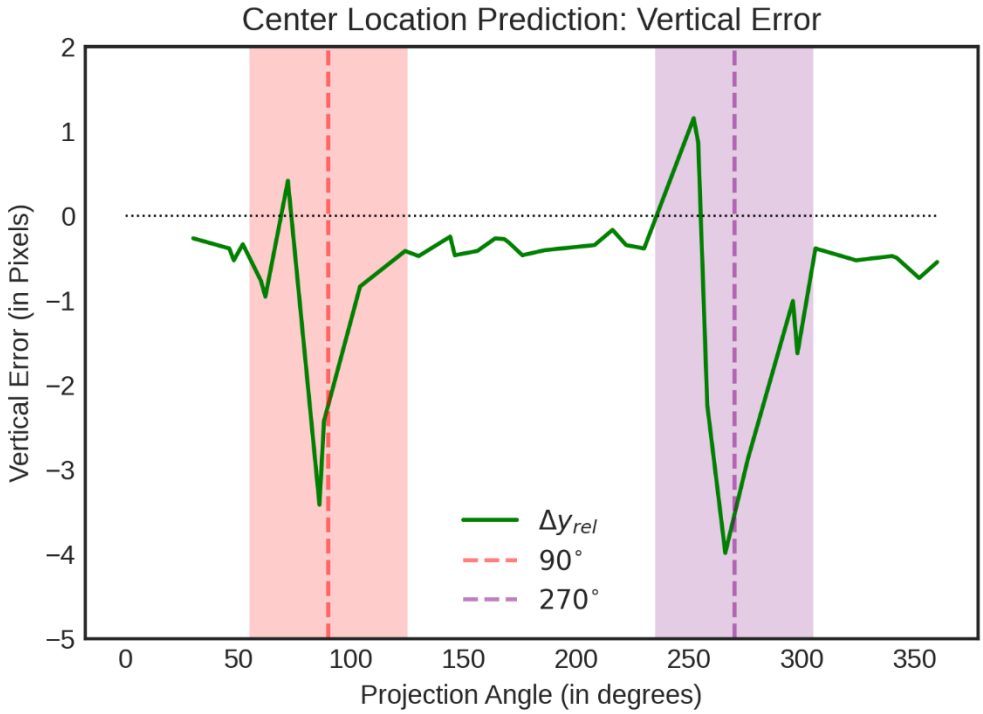


***Figure 55**: Center prediction error in y with DL model + Heuristic Method*

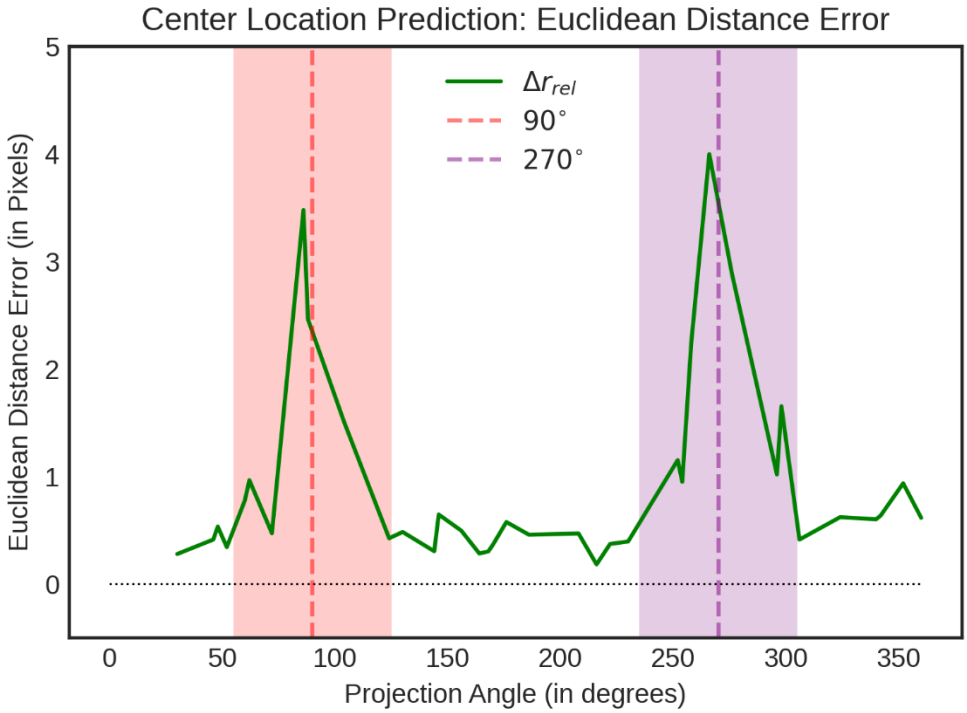


***Figure 56**: Center prediction error in r with DL model + Heuristic Method*

The shaded horizontal spans (90 ± 35 degrees and 270 ± 35 degrees) on the above three figures (**Figure 54**, **Figure 55** and **Figure 56**) show the range of angles we have identified as "hard" to predict the center even manually, owing to the smaller cross-section of the sample/holder and more noise in the images.

The relatively inferior predictions around the shaded regions are expected as the heuristic model was not performing well for those ranges of angles when applied on the original laboratory data earlier.

However, given the time taken for such predictions and the reproducibility achieved in the overall method, application of DL + heuristic method certainly instills optimism. Further research into applying other types of DL models and blending with other potential heuristic methods could be explored as part of future improvements of this study.

6.7. Deep Learning and Python Resources

Since both Deep learning and Machine Learning are vast and quickly growing fields, here are some resources to help the readers grasp some relevant concepts.

The application of Artificial Neural Networks (**ANN**) as part of Deep Learning (**DL**) or in a more general sense Machine Learning (**ML**) has been gaining traction in a wide array of fields [35], [36], [37], [38], [39], [40]. ANN has been implemented in ML/DL based studies involving Microscopy [41], [35], [38], Holography [42], [43]and Computed Tomography (**CT**) [44], [45], Limited Angle CT (**LACT**) [46], [45], [47], Diffraction Tomography (**DT**) [42], FTIR-MS [48]. While Computed Tomography (**CT**) assumes minimal to no scattering for each of the 2D projection images and a linear ray-path. Diffraction Tomography (**DT**) considers bending of light inside the sample for constructing the internal RI distribution [17].

Deep learning is a rather vast and fast-growing field. Often books and online resources tend to become obsolete as the field and the libraries keep updating in a very quick iteration. Even though I have used PyTorch framework [49], the documentation of TensorFlow is still quite helpful in grasping concepts [50].

For Image Segmentation, the TensorFlow tutorials are quite helpful [51]. For general conceptual understanding of deep learning, loss function, model architectures, strengths, weaknesses and limitations of DL and machine learning approaches refer to [37], [52], [53], [36], [54], [55], [39], [38], [46], [56], [57].

The UNet architecture is comprised of a compressing encoder arm and a decompressing decoder arm. To gain more clarity on Auto-Encoders and Generative Models in general, please refer to [58], [59], [60], [61], [62], [63], [64], [65].

The DL and ML application and the entire data processing was done using Python. Please refer to *Python Object-Oriented Programming* book [66] for the object-oriented programming knowledge required for using advanced ML/DL frameworks.

The book *Dive into Deep Learning* [67] is a well-balanced blend of theory and practical implementations using different DL frameworks, including PyTorch. Computer Vision (CV) related theory and concepts could be helpful for understanding and extending this work. Please refer to [68] and [69] for CV.

Machine Learning is a vast field. Dimension reduction is a subfield of ML. To learn more about hands on methods for such as PCA, please refer to [70]. The book, *Machine Learning with PyTorch and Scikit-Learn* [71] is a decent resource for learning about PyTorch (DL) and Scikit-Learn (ML).

Machine Learning related research progresses very quickly and a lot many more papers get published every year. This could often make it incredibly challenging in keeping up with the field. I have used three different tools/sites that helped with reducing noise while searching for papers and comparing/validating their claims: Papers-with-Code [72], Made-with-ML [73] and Connected Papers [74]. I would encourage the reader to explore these resources and use them where deemed fit.

7. Conclusion

IR imaging comes with its own set of limitations – diffraction, scattering, being some of the notorious ones. This leads to a region of ambiguity around the true edge of the object under study in the imaged data. PCA based background subtraction could yield an objective and reproducible method to identify the object from the homogenous background, solely based on information content. Manufacturing custom specified tomography stage with 6 degrees of freedom comes with its own set of engineering challenges and limitations. Use of all the rotational axes may not yield equal benefit and some may accrue detrimental effects when reproducibility is considered. Moreover, such custom-made stages, designed to operate in highly constrained spaces, may need other custom-made tools before acceptable lab use. Even when the stage can be used in the lab, it may require custom designed prototype software to facilitate proper functioning. AutoIt is a software-language that could be used to address such rapid prototyping of ad hoc demands presented by scientific experiments in a laboratory. Engineering limitations of such lab-automation, coupled with mechanical engineering limitations of such stages, could present further problems for precise positioning of the object of interest. While heuristic methods can be used as an option to further correct any irregularities or uncertainties in the imaged data's location offsets (introduced due to mechanical stress/limitations or uncertainties of the imaging system's stage), they may offer a good first approximation in improving the sinograms of the imaged datasets. Machine Learning solutions could pave the way when heuristic options have been exhausted. However, ML models need data to train on with well-known ground truth. If the training data has problems or bias, that will show up in the trained model. Synthetic datasets could alleviate the conundrum of solving the positioning problem even before the dataset acquisition has been perfected. The use of synthetic data was possible with hyperspectral images, as objective

background subtraction was owing to the use of PCA along the wavelength-dimension of the FTIR dataset. Finally, as a next improvement, we suggest that a combination of heuristic and ML based methods together may provide better prediction mechanism for identifying the center of the microloop and offsetting the shifted images.

References

[1] R. Osibanjo, R. Curtis and Z. Lai, "LibreTexts Chemistry," [Online]. Available: https://tinyurl.com/ir-spectroscopy-explained. [Accessed 12 Jul 2023].

[2] A. J. Schofield, "Scattering in Infrared Microspectroscopy, Theses and Dissertations.," University of Wisconsin - Milwaukee, Commons, 2019.

[3] B. H. Stuart, Infrared Spectroscopy: Fundamentals and Applications, John Wiley & Sons, Ltd, 2004.

[4] P. Griffiths and J. de Haseth, Fourier Transform Infrared Spectrometry, John Wiley & Sons, Ltd, 2007.

[5] G. Azarfar, "Light Scattering in Diffraction Limit Infrared Imaging," Theses and Dissertations. 2156, UWM Digital Commons, 2019.

[6] M. C. Martin, C. Dabat-Blondeau, M. Unger, J. Sedlmair, D. Y. Parkinson, H. A. Bechtel, B. Illman, J. M. Castro, M. Keiluweit, D. Buschke, B. Ogle, M. J. Nasse and C. J. Hirschmugl, "3D spectral imaging with synchrotron Fourier transform infrared spectro-microtomography.," *Nature Methods.*, vol. 10, no. 9, p. 861–864, 2013.

[7] E. Aboualizadeh, M. Ranji, C. M. Sorenson, R. Sepehr, N. Sheibani and C. J. Hirschmugl, "Retinal oxidative stress at the onset of diabetes determined by synchrotron FTIR widefield imaging: towards diabetes pathogenesis," *Analyst*, vol. 142, no. 7, pp. 1061-1072, 2017.

[8] A. E, S. CM, S. AJ, U. M, S. N and H. CJ, "Temporal diabetes-induced biochemical changes in distinctive layers of mouse retina.," *Scientific Reports*, vol. 8, no. 1, 2018.

[9] Q. Chen, C. Zhang, J. Zhao and Q. Ouyang, "Recent advances in emerging imaging techniques for non-destructive detection of food quality and safety," *Trends in Analytical Chemistry* , vol. 52, pp. 261-274, 2013.

[10] P. Bassan, H. J. Byrne, J. Lee, F. Bonnier, C. Clarke, P. Dumas, E. Gazi, M. D. Brown, N. W. Clarkedef and P. Gardnera, "Reflection contributions to the dispersion artefact in FTIR spectra of single biological cells," *Analyst*, vol. 134, pp. 1171-1175 , 2009.

[11] C. F, Bohren and D. R. Huffman, Absorption and Scattering of Light by Small Particles, Wiley, 1998.

[12] M. Romeo, B. Mohlenhoff and a. M. Diem, "Infrared micro-spectroscopy of human cells: Causes for the spectral variance of oral mucosa (buccal) cells," *Vibrational Spectroscopy*, vol. 42, no. 1, pp. 9-14, 2006.

[13] B. P, K. A, M. H, L. J, B. HJ, D. P, G. E, B. M, C. N and G. P., "Resonant Mie scattering (RMieS) correction of infrared spectra from highly scattering biological samples," *Analyst*, vol. 135, no. 2, pp. 268-277, 2010.

[14] P. Bassan, H. J. Byrne, F. Bonnier, J. Lee, P. Dumasc and P. Gardner, "Resonant Mie scattering in infrared spectroscopy of biological materials – understanding the ‘dispersion artefact’," *Analyst*, vol. 134, pp. 1586-1593 , 2009.

[15] B. P, K. A, M. H, L. J, J. E, L. N, D. P, B. M, C. N and G. P., "RMieS-EMSC correction for infrared spectra of biological cells: extension using full Mie theory and GPU computing.," *Journal of Biophotonics*, Vols. 8-9, pp. 609-620, 2010.

[16] M. Romeo and M. Diem, "Correction of dispersive line shape artifact observed in diffuse reflection infrared spectroscopy and absorption/reflection (transflection) infrared micro-spectroscopy," *Vibrational Spectroscopy*, vol. 38, no. 1-2, pp. 129-132, 2005.

[17] P. Müller, M. Schürmann and J. Guck, "The Theory of Diffraction Tomography," arXiv:1507.00466, 2016.

[18] O. Ronneberger, P. Fischer and T. Brox, *UNet: Convolutional Netweorks for Biomedical Image Segmentation*, 2015.

[19] V. Zohdi, D. R. Whelan, B. R. Wood, J. T. Pearson, K. R. Bambery and M. J. Black, "Importance of Tissue Preparation Methods in FTIR Micro-Spectroscopical Analysis of Biological Tissues: 'Traps for New Users'," *PLOS ONE*, 2015.

[20] B. C. Smith, Fundamentals of Fourier Transform Infrared Spectroscopy, 2nd edn., Boca Raton, FL: CRC Press., 2011.

[21] A. V. Margaris, "Fourier Transform Infrared Spectroscopy (FTIR): applications in archaeology.," in *In Smith, C. (ed.), Encyclopedia of Global Archaeology.*, New York: Springer, 2014, p. 2890–2893.

[22] J. Robert F. Cleveland and J. L. Ulcek, "Questions and Answers about Biological Effects and Potential Hazards of Radiofrequency Electromagnetic Fields," *Federal Communications Commission Office of Engineering & Technology*, 1999.

[23] T. SR and H. MR, "iological effects and medical applications of infrared radiation.," *Journal of Photochemistry and Photobiology B: Biology*, vol. 170, pp. 197-207, 2017.

[24] Nature.com, "Scitable by Nature Education," [Online]. Available: https://www.nature.com/scitable/topicpage/what-is-a-cell-14023083/. [Accessed August 2023].

[25] AutoIt, *AutoIt Homepage*, AutoIt.

[26] M. C. Carvalho, Practical Laboratory Automation: Made Easy with AutoIt, Wiley-VCH, 2016.

[27] N. Rupp, K. Peschke, M. Koppl, D. Drissner and T. Zuchner, "Establishment of low-cost laboratory automation processes using AutoIt," *SLAS Technology*, vol. 27, no. 2022, pp. 312-318, 2022.

[28] MiTeGen LLC, "MiTeGen," 10 Jul 2023. [Online]. Available: https://www.mitegen.com/learn/about-mitegen/.

[29] OpenSCAD, "OpenSCAD," 10 Jul 2023. [Online]. Available: www.openscad.org/index.html.

[30] Torchvision maintainers and contributors, *Torchvision: PyTorch's Computer Vision library*, GitHub repository, 2016.

[31] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin and A. A. Kalinin, *Albumentations: Fast and Flexible Image Augmentations*, vol. 11, 2020.

[32] E. Riba, D. Mishkin, D. Ponsa, E. Rublee and G. Bradski, "Kornia: an Open Source Differentiable Computer Vision Library for PyTorch," in *Winter Conference on Applications of Computer Vision*, 2020.

[33] E. Wood, T. Baltrušaitis, C. Hewitt, S. Dziadzio, T. J. Cashman and J. Shotton, "Fake It Till You Make It: Face Analysis in the Wild Using Synthetic Data Alone," in *ICCV*, 2021.

[34] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," *ArXiv*, 10 Dec 2015.

[35] F. Xing, Y. Xie, H. Su, F. Liu and L. Yang, "Deep Learning in Microscopy Image Analysis: A Survey," *IEEE Transactions on Neural Networks and Learning Systems,*, vol. 29, no. 10, pp. 4550-4568, 2017.

[36] A. Canziani, A. Paszke and E. Culurciello, "An Analysis of Deep Neural Network Models for Practical Applications," arXiv:1605.07678, 2016.

[37] F. Chollet, Deep Learning with Python, Manning, 2017.

[38] Y. Z. G. C. L. Y. Le Lu, Deep Learning and Convolutional Neural Networks for Medical Image Computing, Precision Medicine, High Performance and Large-Scale Datasets, Springer, 2017.

[39] J. Patterson and A. Gibson, Deep Learning: A Practitioner's Approach, O'Reilly Media, 2017.

[40] F. Deng, S. P. 2, X. Chen, Y. Shi, T. Yuan and S. Pu, "Hyperspectral Image Classification with Capsule Network Using Limited Training Samples," *Sensors (Basel),* vol. 18, no. 9, p. 3153, 2018.

[41] Y. Rivenson, Z. Gorocs, H. Gunaydin, Y. Zhang, H. Wang and A. Ozcan, "Deep Learning Microscopy," *Optica,* vol. 4, no. 11, pp. 1437-1443, 2017.

[42] T. Nguyen, V. Bui and G. Nehmetallah, 3D Optical Diffraction Tomography Using Deep Learning, (3D, AO, AIO, COSI, DH, IS, LACSEA, LS&C, MATH, pcAOP), OSA Technical Digest (Optica Publishing Group, 2018), 2018.

[43] Y. Rivenson, Y. Zhang, H. Günaydın, D. Teng and A. Ozcan, "Phase recovery and holographic image reconstruction using deep learning in neural networks.," *Light: Science & Applications,* vol. 7, no. 17141 , 2018.

[44] T. Wurfl, M. Hoffmann, V. Christlein, K. Breininger, Y. Huang, M. Unberath and A. K. Maier, "Deep Learning Computed Tomography: Learning Projection-Domain Weights From Image Domain in Limited Angle Problems," *IEEE Transactions on Medical Imaging,* vol. 37, no. 6, pp. 1454-1463, 2018.

[45] H. Zhang, L. Li, K. Qiao, L. Wang, B. Yan, L. Li and G. Hu, "Image Prediction for Limited-angle Tomography via Deep Learning with Convolutional Neural Network," arXiv:1607.08707, 2016.

[46] G. E., A. K. Hinton and S. D. Wang, *Transforming Auto-Encoders. In: Honkela, T., Duch, W., Girolami, M., Kaski, S. (eds) Artificial Neural Networks and Machine Learning – ICANN 2011. ICANN 2011.*, vol. 6791, Springer, Berlin, Heidelberg. .

[47] K. Hammernik, T. Würfl, T. Pock and A. Maier, "A Deep Learning Architecture for Limited-Angle Computed Tomography Reconstruction," in *In: Maier-Hein, geb. Fritzsche, K., Deserno, geb. Lehmann, T., Handels, H., Tolxdorff, T. (eds) Bildverarbeitung für die Medizin 2017. Informatik aktuell.* , Springer Vieweg, Berlin, Heidelberg., 2017.

[48] P. Lasch, M. Diem, W. Hänsch and D. Naumann, "Artificial neural networks as supervised techniques for FT-IR microspectroscopic imaging.," *Journal of Chemometrics: A Journal of the Chemometrics Society,* vol. 20, no. 5, pp. 209-220, 2006.

[49] PyTorch Foundation, *PyTorch,* PyTorch.

[50] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia and L. Rafal Jozefowicz, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," 2016, arXiv:1603.04467.

[51] Tensorflow, "Tensorflow Tutorials: Image Segmentation," [Online]. Available: https://www.tensorflow.org/tutorials/images/segmentation. [Accessed Oct 2022].

[52] F. Chollet, *Keras: The Python Deep Learning library,* 2018.

[53] F. Chollet, Deep Learning with Python, 2nd ed., Manning Publications Co., 2021.

[54] V. E. Balas, B. K. Mishra and R. Kumar, Handbook of Deep Learning in Biomedical Engineering - Techniques and Applications, Academic Press, 2020.

[55] G. Cybenko, "Approximation by superpositions of a sigmoidal function.," *Math. Control Signal Systems* , vol. 2, p. 303–314, 1989.

[56] J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," arXiv:1404.7828, 2014.

[57] K. Hornik, M. Stinchcombe and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks,* vol. 2, no. 5, pp. 359-366, 1989.

[58] C. Doersch, "Tutorial on Variational Autoencoders," arXiv:1606.05908 , 2016.

[59] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," arXiv:1312.6114, 2013.

[60] D. P. Kingma, D. J. Rezende, S. Mohamed and M. Welling, "Semi-Supervised Learning with Deep Generative Models," arXiv:1406.5298, 2014.

[61] E. Abbasnejad, A. Dick and A. Hengel, "Infinite Variational Autoencoder for Semi-Supervised Learning," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2016.

[62] I. Goodfellow, "Generative Adversarial Networks," arXiv:1701.00160, 2016.

[63] G. E. Hinton, A. Krizhevsky and S. D. Wang, "Transforming Auto-Encoders.," in *In: Honkela, T., Duch, W., Girolami, M., Kaski, S. (eds) Artificial Neural Networks and Machine Learning – ICANN 2011. ICANN 2011. Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg., 2011.

[64] Y. Pu, Z. Gan, R. Henao, X. Yuan, C. Li, A. Stevens and L. Carin, "Variational Autoencoder for Deep Learning of Images, Labels and Captions (Neural Information Processing Systems)," *Advances in Neural Information Processing Systems 29 (NIPS,* 2016.

[65] A. Radford, L. Metz and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," arXiv:1511.06434 , 2015.

[66] S. F. Lott and D. Phillips, Python Object-Oriented Programming, 4th ed., Packt, 2021.

[67] A. Zhang, Z. C. Lipton, M. Li and A. J. Smola, Dive into Deep Learning (D2L), D2L, 2023.

[68] A. R. Zamir, A. Hakeem, L. Van Gool, M. Shah and R. Szeliski, Advances in Computer Vision and Pattern Recognition: Large-Scale Visual Geo-Localization (book), et. al., Springer, 1st ed., Springer, 2016.

[69] R. Szeliski, Computer Vision Algorithms and Applications, 2nd ed., Springer, 2022.

[70] F. Pedregosa, G. Varoquaux and others, "Scikit-learn: Machine Learning in Python," *arXiv:1201.0490,* 2012.

[71] S. Raschka, V. Mirjalili and Y. (. Liu, Machine Learning with PyTorch and Scikit-Learn, Packt, 2022.

[72] Papers-with-Code, *Papers with Code: Image Segmentation*, https://paperswithcode.com.

[73] G. Modandas, *Made with ML*, madewithml.com.

[74] Connected Papers, *Connected Papers - UNet Paper (2015)*, connectedpapers.com.