Improvement of Image Reconstruction Techniques for μMRI

Submitted by
Simon Miller

Mathematics

To
The Honors College
Oakland University

In partial fulfillment of the
requirement to graduate from
The Honors College

Mentor: Dr. Yang Xia, Professor of Physics
Department of Physics
Oakland University

May 13, 2020

Abstract

In microscopic magnetic resonance imaging (μMRI), time-domain signal is sampled on a nuclear magnetic resonance (NMR) spectrometer, then transformed into the frequency domain as the image. The conventional technique of data sampling traverses on a uniform Cartesian grid, which can be reconstructed into images by the application of the Fast Fourier Transform (FFT). In specific cases, the data sampling uses a radial grid, which needs to be interpolated onto the standard Cartesian grid before transformation. Existing methods are competitive but could be improved. This paper explores alternative image reconstruction algorithms for the radially sampled time-domain data to minimize the image artifacts due to the reconstruction of the Cartesian grid.

I. Introduction

In μMRI, there are several approaches to acquiring the time-domain data when imaging an object. The standard approach, Cartesian sampling, utilizes equidistant sampling, which makes reconstruction of images from spectrometer samples a simple process due to the robustness of the FFT. Utilizing a radial sampling trajectory allows one to reduce the 'dead time,' which is the time delay before sampling, and to avoid the motion artifacts sometimes associated with Cartesian sampling of the time domain. However, utilizing radial sampling can also cause a loss of clarity during the image reconstruction process, so improving reconstruction methods would be beneficial.

By improving the image reconstruction quality given radially sampled time-domain data, one could potentially improve the analysis of medical images in MRI and computer tomography

(CT). This could be useful, as it could allow for earlier and more confident diagnoses of diseases such as osteoarthritis, and a better-documented view of the progression of these diseases.

A straightforward way to judge the quality of the image construction is to compare the cross-sections of images resulting from different reconstruction techniques, as one could potentially identify better reconstruction techniques and exclude poor reconstruction techniques from future consideration. Also, with information about spectrometer settings used during the data acquisition process and properties of the object being imaged, one could gain insight into which reconstruction technique is most appropriate if any trends exist.

II. Data Collection

The data used to test the methods developed was collected with Bruker AVANCE II/III-HD micro-imagers (Billerica, MA) with a 7T/9cm vertical-bore magnet. Samples imaged included several glass phantoms that contained doped water (0.1 and 1% CuSO4) in single or multiple tube setup. The MRI protocols used for data collection were set as the following: MSME sequence (in the single-slice single-echo mode) used a TR of 2000 ms, TE of ~ 8 ms, and a matrix size of 256x256 with 25 mm field of view; the 2D UTE sequence had a similar image size with a TR of 100 ms and a TE of 0.27 ms. Note that the MSME imaging sequence uses the Cartesian sampling, while the UTE imaging sequence uses the radial sampling. The gradient trajectory was pre-measured on a glass sphere of water.

III. Mathematical Background

When developing techniques to reconstruct images, it is often beneficial to understand the mathematical nature of the issue at hand. Given a set of polar samples containing information from the time-domain, one may reconstruct the image by transforming the data into the frequency-domain. One can consider the time-domain information as belonging to a function $M(k_x, k_y)$,

which is a function of the k-space coordinates, with $M(k_x, k_y) \in \mathbb{C}$. This can be transformed into the function $M(k_r, \theta)$ by allowing $k_x = k_r \cos\theta$ and $k_y = k_r \sin\theta$, where k_r is the distance in k-space between the sample and the origin and θ is the angle between a ray from the origin to the sample and the polar axis. This is a standard conversion between Cartesian and polar coordinates. The function $M(k_r, \theta)$ is sampled by the function $S(k_r, \theta)$, acting similarly to a delta function or the Dirac comb, for polar sampling. It allows for a finite set of discrete samples to be taken from the continuous information present in k-space.

In the problem discussed in this paper, a collection of radial projections rotated about the origin a finite number of times at an equispaced angle is considered. The values of k_r are the radial distances from the origin of k-space at which each sample is taken, denoted by the set R. The equispaced angles of each projection, θ, with angle $\Delta\theta$ between each radial projection, ranging from 0 to 2-$\Delta\theta$, belong to the set T. To reconstruct the image, one can assign values on an equispaced grid on a square circumscribed about the radial sample. To achieve this, the values of $M(k_r, \theta)S(k_r, \theta)$, for $k_r \in R$ and $\theta \in T$, are used to assign values to $M(k_x, k_y)$ for $k_x \in X$ and $k_y \in Y$, where X and Y are sets containing 256 discrete, equally spaced points along the

interval $-1 \times \max\{r : r \in R\}$ to $\max\{r : r \in R\}$. A visualization of the locations of a subset of the collected data and corresponding locations of the desired data locations are below.

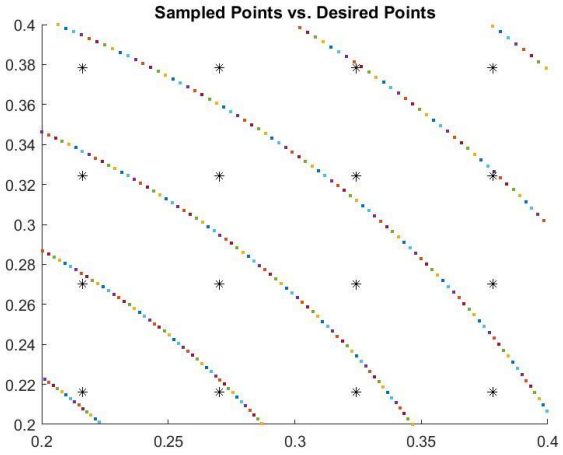

Figure 1: The subset of polar samples is seen in the fan-shaped pattern emanating from the origin. The subset of the locations of the desired data is located on the rectilinear grid, marked by the ''. This graphic was motivated by Pauly[1].*

In the technique proposed, this assignment process is completed by generating a Delaunay triangulation from the set of polar samples and using triangulation-based cubic interpolation to assign values to $M(k_x, k_y)$ for $k_x \in X_{fine}$ and $k_y \in Y_{fine}$, where X_{fine} and Y_{fine} are sets containing 4096 discrete, equally spaced points along the interval $-1 \times \max\{r : r \in R\}$ to $\max\{r : r \in R\}$. These values are then used for the final assignment, where the 289 points that form a square about (k_x, k_y) on the fine grid are used to assign a value to $M(k_x, k_y)$for $k_x \in X$ and $k_y \in Y$, the sets described above.

One can notice that this is the equidistant grid that is desired. In the basic problem of reconstructing a 256 x 256 image, once values have been assigned to the equispaced grid, one can apply the 2D FFT to obtain the image data, given by

$$f(x,y) = \frac{1}{256\times256}\sum_{k_x\in X}\sum_{k_y\in Y} F(k_x,k_y)\exp\left[2\pi i\left(\frac{k_x x}{256}+\frac{k_y y}{256}\right)\right]$$

with $x, y \in X = Y = \{1,2,\ldots,256\}$. This result will yield a 256 x 256 matrix which stores the pixel values for the image of interest, as the time-domain data is transformed into the frequency domain.

IV. Standard Gridding

Because of the existing simple, robust methods for image reconstruction using gridded time-domain data, a large portion of active research regarding the radial grid sampling focuses on interpolating the data onto a Cartesian grid. These approaches are termed gridding or nonuniform FFT (NUFFT) approaches. There are also efforts to apply methods from computed tomography (CT) to MRI. The core idea is to use a set of scattered k-space samples to assign values to a Cartesian grid in k-space. These NUFFT methods were first introduced by J. O'Sullivan[2]. Joint work by Jeffrey Fessler and Bradley Sutton[3] was successful in extending some of these ideas to MRI and further developing the powerful method.

In the first attempt at reconstructing an image from the radial samples, the idea of interpolating onto a Cartesian grid is borrowed from O'Sullivan and Fessler, but the method by which one obtains the gridded Cartesian data is a novel approach. The steps are outlined below.

Algorithm 1: Standard Gridding

Convert sample coordinates from radial to polar
Define ultrafine 4096×4096 grid to interpolate on to
Triangulation-based cubic interpolation on to 4096×4096 grid
Neighbor-based interpolation on to 256×256 grid
2D FFT and FFT Shift
Take modulus of each pixel
Apply binary mask

As an input to the algorithm in the case of this paper, when reconstructing a 256×256 image, one should expect 804 radial spokes, with evenly spaced angles between them, that begin at the origin of k-space. Each spoke is sampled unevenly 131 times along itself. Because the data

is collected in this way, a large number of data points are collected near the origin of k-space and fewer samples are collected near the edges. This makes interpolation onto the Cartesian grid at the edges of k-space more difficult for our gridding approach.

For a given scan, the sampling scheme used along a given radial projection is known, given by coordinate pairs (k_x, k_y) for 131 samples. This information helps obtain the distance from the origin for each sample along a spoke since $k_r = \sqrt{k_x^2 + k_y^2}$. It is also known how many radial projections there are, so a mesh-grid can be created. This results in (k_r, θ) coordinate information for each sample collected, which is easily transformed into (k_x, k_y) coordinate information because of the relations $k_x = k_r \cos\theta$ and $k_y = k_r \sin\theta$. This information is stored in a $402 \times 262 \times 3$ matrix, where a $1 \times 1 \times 3$ entry holds the complex value associated with a point in k-space, as well as its x and y coordinate information. For ease of understanding, one can scale the x and y coordinates such that the domain and range of k-space being considered are both $[0, 256]$, with k-space centered at $(128,128)$, as our final image should be 256×256. This domain and range were selected because the coordinates would easily correspond to indices of the desired final matrix. This step could be eliminated for efficiency but is computationally inexpensive when considering the total algorithm runtime.

After transforming the data so it can be easily processed, the samples are used to interpolate onto what will be referred to as an ultrafine grid. In this case, this is a grid with the same domain and range discussed above, but with a spacing of $1/N$ between each interpolated point, where N is a positive integer. $N = 16$ was used in the experiments and this proves to be effective.

Mathematically, this is equivalent to interpolating values of $M(k_x, k_y)$ for $k_x \in X_{fine}$ and $k_y \in Y_{fine}$, where X_{fine} and Y_{fine} are sets containing 4096 discrete, equally spaced points along the interval [0,256].

The interpolation is performed by first defining a mesh grid to represent the desired ultrafine grid serving as query points for our interpolation. The MATLAB[4] command *griddata* is used to interpolate from the set of scattered k-space data onto the ultrafine grid using the information from the processed input matrix (The MathWorks, Inc). While several interpolation methods are offered by *griddata*, the Delaunay triangulation-based cubic interpolation that fits the data using cubic splines performs the best. This interpolation technique is discussed at length in by Amidror[5]. The other options for interpolation methods included linear, nearest neighbor, and natural neighbor interpolation. It is important to note that the real and imaginary values are interpolated separately.

Because our original data was collected radially, there are some points outside of the convex hull in the domain and range $[0, 256]$ that do not have values after interpolation. Hence, cubic spines cannot be used to assign values at these points. To account for this, the values at these points are set equal to 0. This is sensible because the values at the edge of the convex hull approach 0.

Recall that the original samples belonged to a polar grid, while the interpolated values fall on a rectilinear grid. The locations of a subset of the interpolated values are located below.

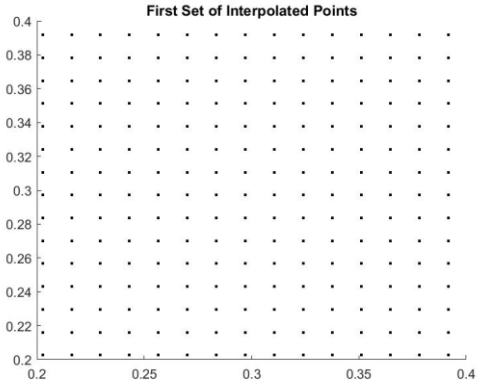

Figure 2: The complex data is interpolated onto the rectilinear grid using the data collected by the scanner on the polar grid. This is an unscaled k-space representation.

It is important to note the interpolation utilizing cubic splines is the most computationally-expensive step of the gridding algorithm. If this algorithm, or any of the algorithms discussed below which perform interpolation utilizing cubic splines, were deployed in a setting where reducing runtime was critical, it may be worth exploring a threaded implementation in which interpolation is performed in each quadrant of k-space simultaneously.

Next, a second interpolation from the uniform ultrafine grid onto the unit Cartesian grid with a domain and range of $[1, 256]$ and unit spacing is performed. Mathematically, this is equivalent to interpolating values of $M(k_x, k_y)$ for $k_x \in X,\ k_y \in Y$, where $X, Y = \{1,2, \dots, 256\}$. To do this, the neighboring points in a 1×1 area near the point on the unit Cartesian grid are selected, then a simple average of these points is taken. The locations of a subset of the interpolated data is depicted below.

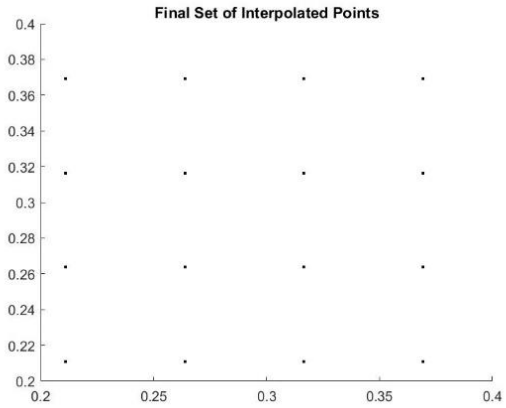

Figure 3: The next stage of interpolation reduces the number of data points on the rectilinear grid. This is the final set of values used for the algorithm. This is an unscaled k-space representation.

Moving the neighborhood, which is not optimized, produces varying results. Performing a Gaussian weighted-average centered about the location of interest did not improve the results.

The above is sufficient to interpolate data onto the desired unit Cartesian grid. Hence, the 2-dimensional FFT (2DFFT) of the data paired with performing an FFT shift is the next appropriate step:

$$f(x,y) = \frac{1}{256 \times 256} \sum_{k_x \in X} \sum_{k_y \in Y} F(k_x, k_y) \exp\left[2\pi i \left(\frac{k_x x}{256} + \frac{k_y y}{256}\right)\right],$$

$$x, y \in X = Y = \{1,2,\ldots,256\}.$$

Note that at this point, the matrix is a 256×256 complex matrix. Each entry is of the form $a + bi$, with $a, b \in \mathbb{R}$. One can then take the modulus of the value associated with each pixel, $\sqrt{a^2 + b^2}$, to obtain the image as a matrix, where each entry represents a grayscale pixel value. A binary mask is applied to the image to blackout the unknown portions of the image, as one can

only reliably image the circular area inscribed in the square of the 256×256 image due to the radial sampling scheme used.

Standard gridding successfully reconstructs the image, often performing as well as or better than the industry-approved radial sample reconstruction. However, when the object being imaged is large enough, there exist wraparound artifacts that could cause confusion during image analysis. This can be observed in the figure below.

Figure 4: Observe the artifact present in the image of the concentric tubes of liquid. The rings that wrap around the edges should not be present.

V. Gridding with Oversampling

While the standard gridding method produces the image reliably, one can observe wraparound artifacts that obscure the image near the edges. The effect is amplified as contrast in the image increases. This is what motivates the inclusion of the technique dubbed "oversampling" in the next stage of the development of the algorithm. This concept was discussed by Pauly[1]. The steps to the modified algorithm are outlined below.

Algorithm 2: Gridding with Oversampling

Convert sample coordinates from radial to polar
Define ultrafine 4096×4096 grid to interpolate on to
Triangulation-based cubic interpolation on to 4096×4096 grid
Neighbor-based interpolation on to 512×512 grid
2D FFT and FFT Shift
Take modulus of each pixel
Select central 256×256 submatrix
Apply binary mask

All steps of this algorithm are the same as the standard gridding algorithm, up to, but not including, the second interpolation. Instead of interpolating onto a unit 256×256 grid, interpolate onto a 512×512 grid with domain and range $[1, 256]$ and a step size of 0.5 between points. This is equivalent to interpolating values of $M(k_x, k_y)$ for $k_x \in X,\ k_y \in Y$, where $X, Y = \{1, 1.5, 2, \ldots, 255.5, 256\}$. To interpolate, use the same technique of taking the average of a certain set of surrounding neighbors, now those in the 0.25×0.25 area to the left and beneath the point being interpolated. This set of neighbors can still be refined. This interpolation is called oversampling because the same set of data as before is being considered, but more points are included in the final set of points. Utilizing a 512×512 grid proves to be effective in resolving the wraparound artifacts that sometimes present themselves. The locations of a subset of the interpolated data is depicted below.

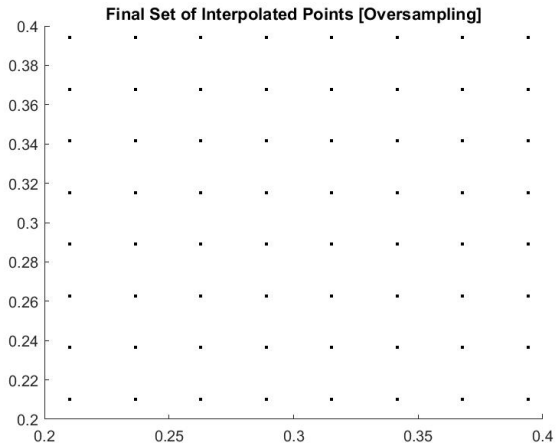

Figure 5: This is the final set of values used for the algorithm. Notice that there are 4-times as many points in the same region as the one depicted in the previous figure.

The next steps to the gridding algorithm with oversampling are similar to the standard gridding algorithm. A 2DFFT is applied to the data, followed by an FFT shift. The modulus of each pixel is taken, and the results are stored as image data. However, because the FFT is applied to a 512×512 matrix, the image produced is also 512×512, but the image of interest is 256×256. Hence, it is necessary to select the 256×256 submatrix centered in the 512×512, after which a binary mask can be applied. This concludes the process of the gridding algorithm with oversampling. The improvements it provides can be observed in the figure below.

Figure 6: Note that this image is produced using the same set of data as the image included in Figure 5. Interpolating onto a larger grid eliminates the wraparound artifact.

Gridding with oversampling successfully removes the wraparound artifacts present in standard gridding, at the cost of increasing runtime and a slightly worse ROI profile plot. In

some scenarios, this method also outperforms the industry-approved radial sample reconstruction.

VI. Gridding with Oversampling and Deapodization

In conventional NUFFT methods, a convolution kernel is used to interpolate onto the Cartesian grid from polar coordinates, resulting in apodization, a type of image imperfection, in the final image. To counteract this, many implement a process called deapodization to reverse these effects. This is the technique of dividing the image matrix values by the inverse FFT of the convolution kernel. It is, in essence, a spatial filter. While the standard gridding and gridding with oversampling methods do not use a convolution kernel for interpolation, there were imperfections in the profile plots of some images that looked similar to the effects of apodization noted in the literature. Specifically, the effects of apodization and its resolution with deapodization were discussed by Otazo[6]. This motivated us to include the additional step of deapodization in the next stage of algorithm development. The general steps are included below.

Algorithm 3: Gridding with Oversampling and Deapodization

Convert sample coordinates from radial to polar
Define ultrafine 4096×4096 grid to interpolate on to
Triangulation-based cubic interpolation on to 4096×4096 grid
Neighbor-based interpolation on to 512×512 grid
2D FFT and FFT Shift
Take modulus of each pixel
Construct and apply Gaussian spatial filter
Select central 256×256 submatrix
Apply binary mask

The steps of Gridding with Oversampling and Deapodization are the same as Gridding with Oversampling until the image is saved. Before saving the image, the Gridding with Oversampling and Deapodization algorithm applies deapodization, described below.

The first step of our deapodization process is to generate a set of 512 evenly spaced x values that span the frequency domain. Then these x values are used as inputs for the closed-form formula for the FFT of the Gaussian convolution kernel. Note that the algorithms created thus far as the topic of this paper did not apply convolution, so this choice in kernel is arbitrary. The Gaussian kernel's FFT has a closed-form formula, which is makes its implementation a simple process[7]. A parameter value of $\sigma = 0.25$ was selected for the final analysis based on a comparison of performances with values in the range $[0,1]$. This results in a 512×1 vector. Next, the transpose of the vector is taken and multiplied it against itself to produce a 512×512 matrix, which acts as the spatial filter discussed above. Divide each element of the image matrix by the corresponding element of the spatial filter constructed. Then, select the middle 256×256 elements of the result and stored this as the image matrix.

Gridding with oversampling and deapodization often performs as well as, or better than, gridding with oversampling, with a similar runtime. In most scenarios, it is our most competitive reconstruction method.

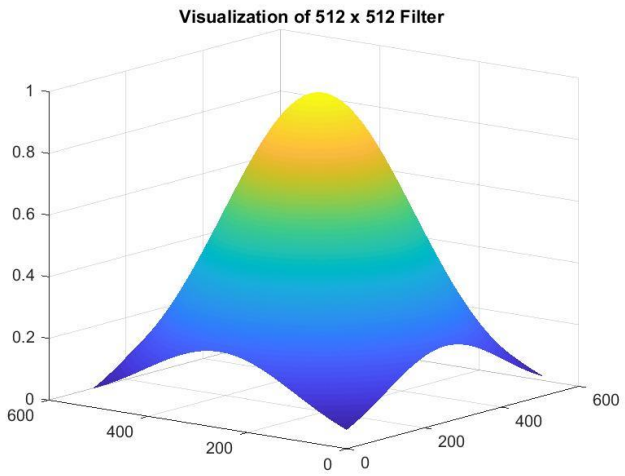

Figure 7: This is a visualization of the 512 x 512 filter discussed above. Values near the center of the image are changed less than those near the edges because values near the center are divided by a number closer to 1 than values at the edges.

VII. Standard Filtered Back Projection

Another common technique in CT (computed tomography) image reconstruction is filtered back projection (FBP). Instead of performing a 2-dimensional interpolation across the entirety of k-space, a 1-dimensional interpolation is used. Because the FFT is still used to transform the data from time-domain to frequency domain, it is necessary to interpolate the data and consider an equidistant set of points. To do this, it is necessary to transform the radial spokes which are sampled along such that they form a diameter through the center of k-space. Because techniques from CT and MRI are very similar, it is interesting to observe the performance of FBP on the μMRI data available. Previous inquiries[8] by the Rice University DSP group into back projection helped gain an understanding of filtered back projection as a reconstruction technique. The general steps are included below.

Algorithm 4: Filtered Back Projection

Interpolate each diameter set of samples to 256 equispaced points using a linear interpolation method
1D FFT and FFT Shift
Inverse radon transform of sinogram with linear interpolation and Ram-Lak filter
Apply binary mask

The steps to read in and transform the FID data are the same as those described in the gridding methods previously discussed. The differences in this method start after transforming the data into sets of diameter samples. Filtered back projection requires that projection data is available. To obtain projection data from the FID, it is necessary to take the FFT of evenly spaced data points. Because the sampling scheme used results in a dense set of samples near the middle of k-space, the FFT does not provide projection data. A 1-D NUFFT approach, similar to the technique used in typical gridding algorithms, could be helpful. Instead, a linear interpolation is performed onto 256 evenly spaced points along each diameter sampled to possess an

equidistant set of complex data. One can then take the 1D FFT of each diameter set of samples and shift it, followed by taking the modulus of each element, to obtain projection data along a given diameter of the image for evenly spaced angles from 0 to $\left(180 - \left(\frac{180}{402}\right)\right)$. The resulting information is called a sinogram.

Upon observation of the sinogram, one can observe several issues. First, there is an apparent discontinuity in the projection data in the middle of sinogram, as a large jump occurs when moving from one projection to the next. To resolve this, a series of operations are used to transform the sinogram into a continuous image with no jumps present. First, the first half of the projection data is switched with the second half of the projection data. Then, each projection that is in the second half of the projection data is flipped. This is sufficient to "piece the puzzle" that is the sinogram together. Additionally, one of the projections is much more intense than the rest. To resolve this, the values of this projection are assigned the values of its neighboring projection.

When utilizing FBP in CT there is an assumption that the information is collected in a continuous, periodic manner. The periodic assumption associated with the use of filtered back projection was also a concern of ours. Because the sample data appeared to not be exactly periodic, and, in practice, the data may not be, 360 degrees of projection data are used instead of the typical 180 degrees. However, the first 180 degrees of projection data are flipped and shifted appropriately to simulate the second 180 degrees of projection data, constructing a continuous sinogram. This allows us to improve the reconstruction without acquiring more information.

If these operations are not performed, streak artifacts are present in the reconstructed image along the edges of objects at 0 and 90 degrees. Yet when the operations are performed, sharpness is lost in the final image. It is recommended that these operations are used, but this is ultimately a user decision.

After transforming the projections, the resulting matrix, as well as the evenly spaced angles, are used as inputs into the *iradon* command from MATLAB's Image Processing toolbox, specifying linear interpolation, application of the Ram-Lak filter, and a desired image size of 256×256, to perform filtered back projection, "smearing" the projection data to obtain the final image.[9] The *iradon* command produces some pixel values which are less than zero. To account for this, these values are set equal to zero. The unknown portions of the image are also blacked out, using the same mask introduced in the standard gridding algorithm.

Below are the results of a reconstruction using gridding with oversampling and deapodization and filtered back projection.

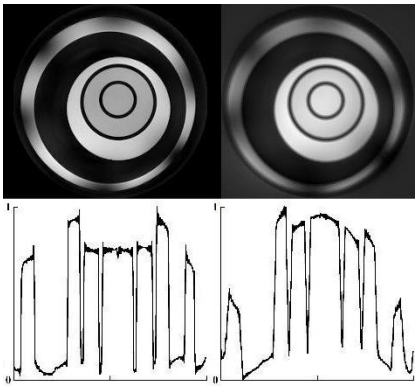

Figure 8: The FBP reconstruction has too much noise and exhibits inhomogeneity.

While the initial filtered back projection algorithm reconstructs the image, there appear to be systematic deficiencies that could be resolved. There is a high presence of noise in the image, exhibited by the lack of depth in areas of the image which separate each tube of liquid. Also, the intensity across uniform samples, as well as across the entire image, is nonuniform. The image appears to have a scaling issue that scales each pixel in the image with a scale factor drawn from a structure similar to a 3D Gaussian bell centered at the image for k-space data acquired with the Bruker AV III HD scanner. This is what motivates the next algorithm. Unless these issues can be resolved, one should utilize the gridding-based approaches.

VIII. FBP with Spatial Filtering

To correct for the issue associated with standard filtered back projection mentioned above, a new method that applies a spatial filter to the reconstruction obtained by standard filtered back projection was designed. To design our spatial filter, the information from a single tube AVIIIHD scan and fit a polynomial to the tube is used, as it is known that it should be approximately uniform. Next, this polynomial is evaluated at 256 equally spaced points to obtain a vector that approximates the distortion of the image through the center slice. Then, this vector is multiplied against its transpose and the absolute value of each element is taken to acquire a 256×256 matrix. Next, a few steps are performed that were experimentally found to improve the image reconstruction quality. The minimum value is added to each element in the matrix and all elements less than a certain value near a corner are set to that value. Apply the filter to the image output by the previous algorithm, before the binary mask is applied, then apply the binary mask. This filter is used for all AVIIIHD scans that acquire k-space data with the same trajectory. The necessary steps are included below.

Algorithm 5: Filtered Back Projection with Spatial Filtering

Interpolate each diameter set of samples to 256 equispaced points using a linear interpolation method
1D FFT and FFT Shift
Inverse radon transform of sinogram with linear interpolation and Ram-Lak filter
Design and apply the spatial filter using the technique discussed
Apply binary mask

In practice, this is often referred to as phantom-based inhomogeneity correction. The correct values of the phantom are known, so correcting for this is a simple process. One simply

needs to apply a filter that returns all values to a single uniform value, which was the goal of the process described above.

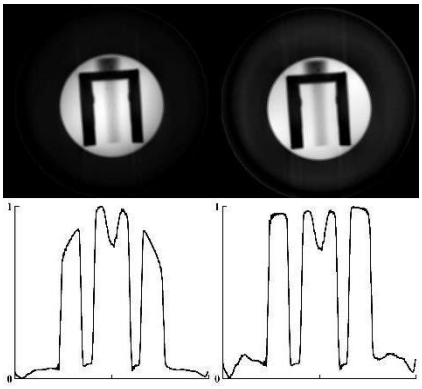

Figure 9: Spatial filtering resolves inhomogeneity, but noise level remains high.

Figure 9 shows the results of applying the spatial filter. The filter appears to be successful in resolving the issues with the bell-shaped inhomogeneity present in FBP reconstructions. Because the noise level in images is still high compared to gridding reconstructions, and because the technique is not relevant for images obtained from AV II spectrometers, it is excluded from analyses.

While the reconstruction quality is not the best when compared to the other techniques, it is important to note that filtered back projection runs significantly faster than the other methods. This is a result of the fact that only 1-D interpolation is performed, as opposed to the 2-D interpolations performed in the gridding approaches discussed above.

IX. Comparison of Results

The images included in the analysis include 256×256 images of single and multi-tube objects obtained from Bruker AV II and Bruker AV III HD spectrometers. 128×128 images are also available, but discussions of the methods focus on 256×256 images, so these results

have been omitted from the analysis. They are available upon inquiry. The analysis presented is also of a horizontal intensity profile through the approximate center of the image, but a vertical intensity profile is also available.

The reconstructed images are located in the appendix. First, in figures 9-11 the set of Bruker AV II images are presented, then, in Figures 12-14, the set of Bruker AV III HD images are presented. A different object imaged is included in Figure 15, but no Cartesian reconstruction is available. Within each collection of image profiles in Figures 9-14, from left to right, one can view the results of an industry-approved Cartesian sample reconstruction, an industry-approved radial sample reconstruction, the standard gridding, gridding with oversampling, and gridding with oversampling and deapodization algorithms. In Figure 14, from left to right, one can view the results of an industry-approved radial sample reconstruction, the standard gridding, gridding with oversampling, and gridding with oversampling and deapodization algorithms.

When interpreting the results, the industry-provided Cartesian reconstruction is considered the most reliable. The industry-provided radial reconstruction is imperfect due to an inability to produce sharp edges and, in AV III HD images, an intensity which peaks on the left and declines as pixels move rightward. In reconstructions produced by standard gridding, edges are represented well, but wraparound artifacts present themselves. Also, in AV III HD images there are issues with accurately portraying the uniformity of the liquid in each tube, as seen in the shallow increases and decreases in intensity near the edge of objects imaged. In reconstructions produced by gridding with oversampling, the same comments discussing standard gridding apply, but the issues with wraparound artifacts no longer persist. In AV III HD images, there appears to be inhomogeneity which assumes a parabolic shape, impacting the representation of the uniformity of solutions in each tube. In reconstructions produced by gridding with

oversampling and deapodization, the edges are represented relatively well and the uniformity of the solutions in each tube is better represented. All AV II images obtained from radial data struggle with representing the uniformity of the solution in each tube. The figure below outlines the differences between each method examined and the strengths and weaknesses associated with each method when reconstructing images from AV III HD data.

AV III HD			
Method	**Procedure**	**Strengths**	**Weaknesses**
Industry Technique	-	Limited Noise No Wraparound Artifact	Inhomogeneity Left to Right
Standard Gridding	Triangulation Interpolation Neighbor Interpolation	Limited Noise	Wraparound Artifact Inhomogeneity
Gridding with Oversampling	Triangulation Interpolation Neighbor Interpolation	Limited Noise No Wraparound Artifact	Limited Inhomogeneity
Gridding with Oversampling and Deapodization	Triangulation Interpolation Neighbor Interpolation Spatial Filter	Limited Noise No Wraparound Artifact	Limited Inhomogeneity

X. Conclusion

The stated goal of this paper was to improve the radial-sampled image reconstruction methods in μMRI and to identify a universally-best approach for reconstructing μMRI images from a set of radial time-domain samples. When using data from Bruker AV II spectrometers, the industry-implemented technique and gridding with oversampling tend to produce the best results. When using data from Bruker AV III HD spectrometers, gridding with oversampling and deapodization tends to produce the best results

Because of the mixed results observed, one is unable to claim a universally-best approach has been identified. The industry-implemented algorithm and both gridding with oversampling and gridding with oversampling and deapodization algorithms each produce competitive results under certain conditions. More specifically, when the field of view becomes too small, the

industry-implemented algorithm becomes the most reliable. Due to the uncertainty surrounding many objects being imaged, it is ultimately up to the user to apply their insights and determine the appropriate reconstruction technique, given they are working with a set of radial time-domain samples.

However, one can remove the interpolation-based FBP approaches from consideration, with the recommendation that a robust NUFFT is used instead of merely utilizing 1D linear interpolation. This may improve results. Nonetheless, applying techniques from CT to μMRI was ambitious and may not produce fruitful results in the future. Convolution and gridding approaches appear to be the better choice when working with μMRI data.

X. Future Work

There are several opportunities to improve the gridding-based approaches discussed in this paper, with respect to both runtime and image reconstruction quality. One major opportunity would be to explore a multithreaded approach to interpolation using the *griddata* command. Because the interpolation is based on triangulation, one could break the interpolation up by quadrants, or even further, to increase the speed with which the algorithm is capable of interpolating. One could also focus on selecting the optimal neighborhood to perform the second interpolation in the gridding-based approaches. There was little difference in image quality when a preliminary analysis was conducted, but this is an additional lever for optimization which could lead to improved results. Lastly, one could dedicate more time to applying NUFFT to the FBP approach, with the hope that it would result in a clearer sinogram. Future efforts should be focused on gridding-based approaches before more effort is applied to understanding the FBP approach.

XI. Acknowledgments

Special thanks to Farid Badar and Dr. Yang Xia for their patience, guidance, and insights provided during the development of this work. Thanks also to the NIH for providing funding to complete this work.

XII. References

1. Pauly, J.M. (2013). Gridding & the NUFFT for non-Cartesian image reconstruction.

2. O'Sullivan, J. (1985). A fast sinc function gridding algorithm for Fourier inversion in computer tomography. *IEEE Transactions on Medical Imaging, 4*(4), 200-207.

3. Fessler, J. & Sutton, B. (2003). Nonuniform fast Fourier transforms using min-max interpolation. *IEEE Transactions on Signal Processing 51*(2), 560-574.

4. The MathWorks, Inc. (n.d.). *griddata.* https://www.mathworks.com/help/matlab/ref/griddata.html.

5. Amidror, I. (2002). Scattered data interpolation methods for electronic imaging systems: a survey. *Journal of Electronic Imaging 11*(2), 157-176.

6. Otazo, R. Reconstruction of non-Cartesian MRI data. https://www.cai2r.net/sites/default/files/documents/BIGP_Docs/Courses_Lectures/Practical%20MRI_2/lecture3.pdf.

7. K. Derpanis. *Fourier transform of the Gaussian.* http://www.cse.yorku.ca/~kosta/CompVis_Notes/fourier_transform_Gaussian.pdf.

8. W. Akram, S. Gee, C. Gamiz, C. Pan, and J. Romberg. Image processing using SPECT analysis.

9. The MathWorks, Inc. (n.d.). *iradon.* https://www.mathworks.com/help/matlab/ref/iradon.html.

XIII. Appendix

Bruker AV II 256×256 Horizontal Intensity Profiles

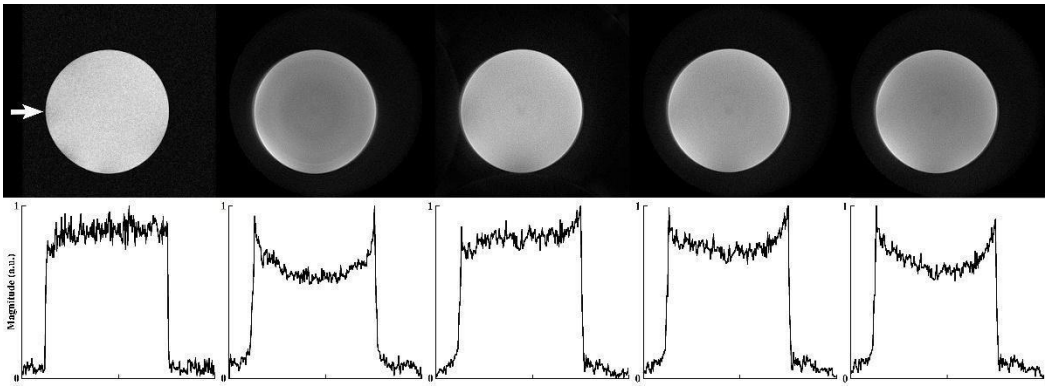

Figure 10

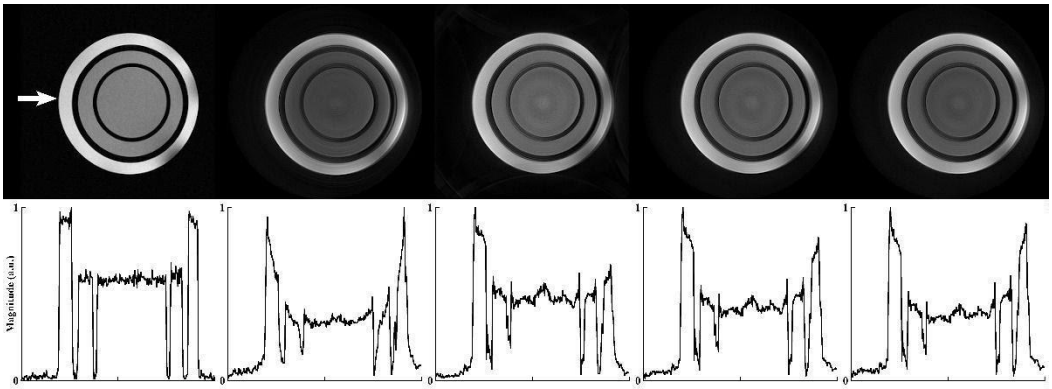

Figure 11

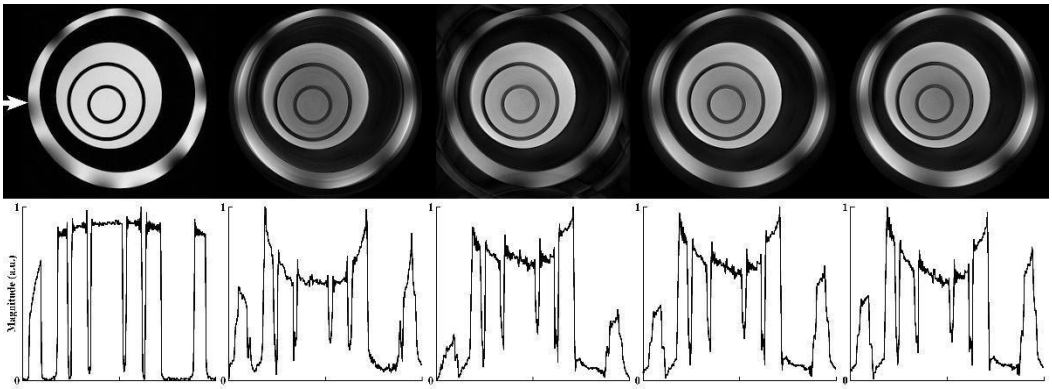

Figure 12

Bruker AV III HD 256 × 256 Horizontal Intensity Profiles

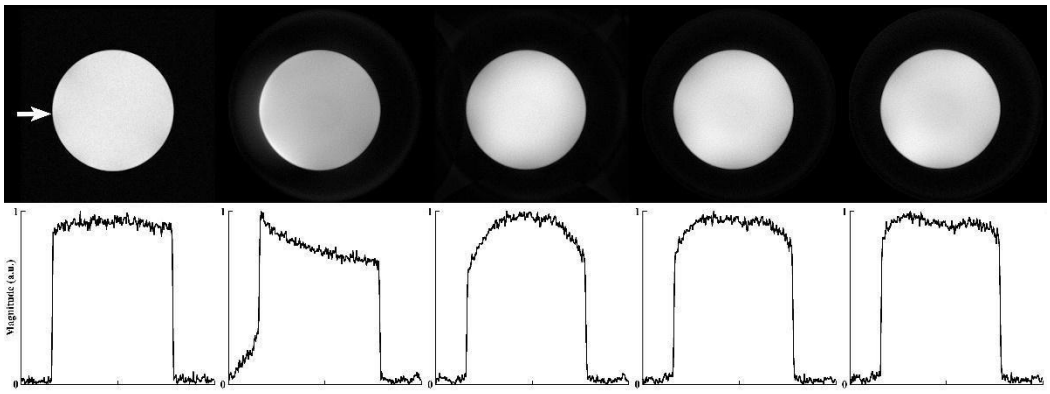

Figure 13

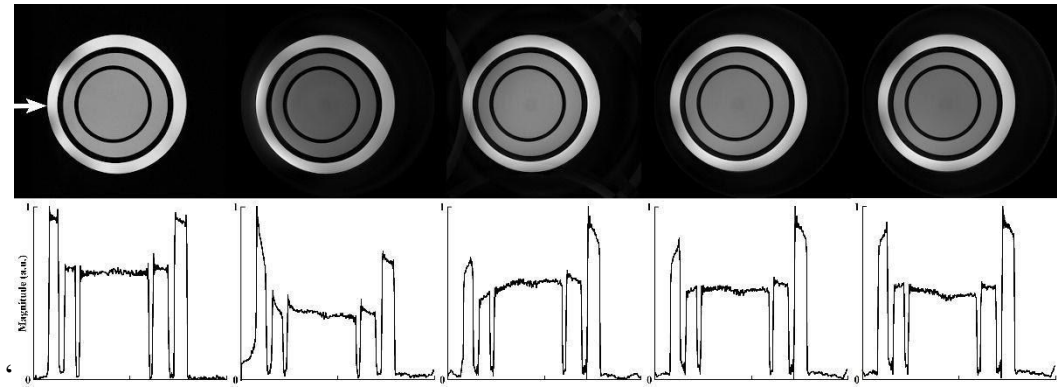

Figure 14

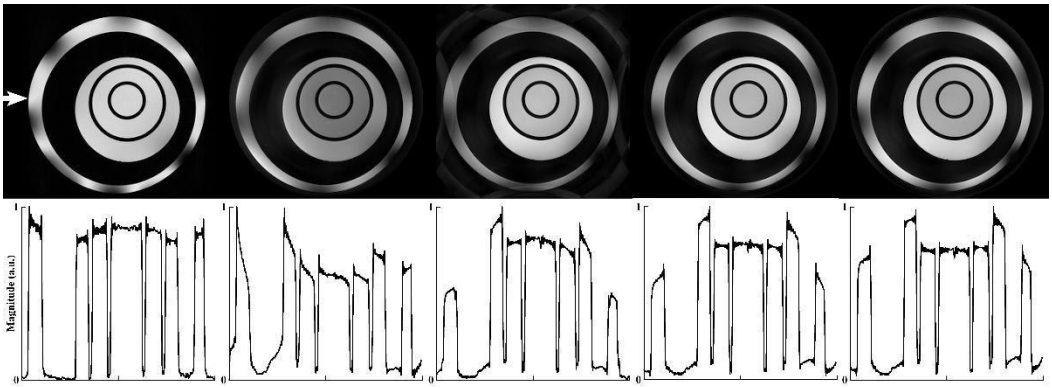

Figure 15

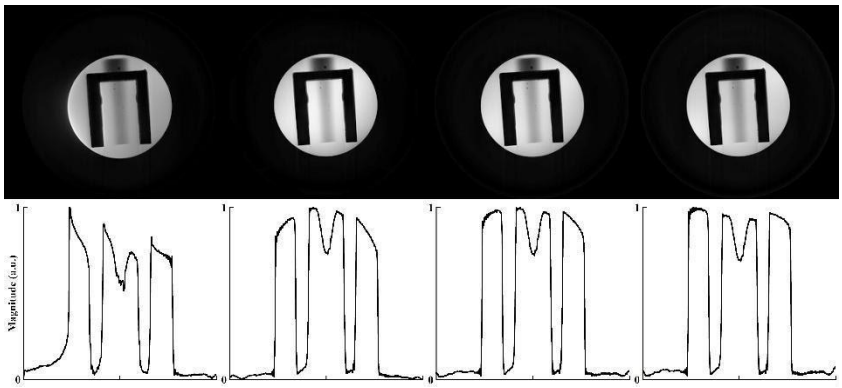

Figure 16: No Cartesian reconstruction available.