

RECONSTRUCTION FROM NONUNIFORM SAMPLES OF K-SPACE

Many thanks to Kenneth Johnson and Dr. James Pipe for the images appearing in these notes.

1. INTRODUCTION

In medical imaging, as well as in some other applications such as radio astronomy, it is not unusual to collect data which are measurements of the (continuous) Fourier transform (CFT) of the objective function, but not on a rectilinear grid. For example, the measurements of the Radon transform in CT are easily converted, via a one-dimensional fast Fourier transform (FFT) to measurements of the CFT on a uniform polar grid, i.e., along radial slices of frequency space. The same sort of data can be collected using projection imaging in MRI. In fact, since our choice of gradients determines the location in frequency space in MRI, nearly any sampling pattern is possible, and one area of active research is a search for the fastest, most efficient scans. One of the most popular scans is a spiral scan out from the origin, which has the advantage of allowing the gradients to vary continuously, rather than being turned on and off as in typical rectilinear imaging. This allows much faster scans than traditional methods, and makes 3D imaging a much more realistic possibility. Another scheme that involves nonuniform sampling is PROPELLER imaging, in which we sample various ‘blades’ of frequency space, all including a region about the origin, in order to account for patient motion during the scan. Figure 1 shows three of these trajectories, the traditional Cartesian trajectory and the alternatives of spiral and PROPELLER trajectories.

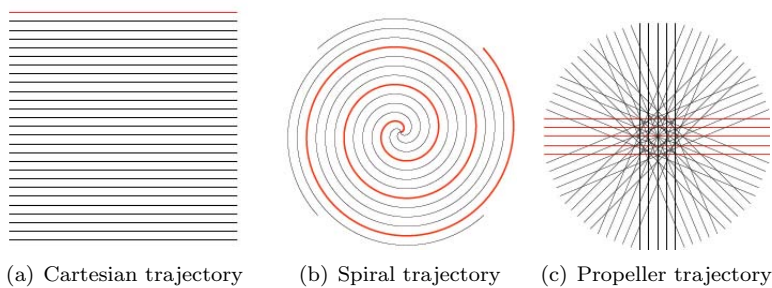


FIGURE 1

Unfortunately, nonuniform sampling makes the use of practical algorithms for reconstruction much more difficult. The Poisson summation formula,

$$\sum_{n \in \mathbb{Z}} f(x + 2nL) = \frac{1}{2L} \sum_{n \in \mathbb{Z}} \hat{f}\left(\frac{n}{2L}\right) e^{\pi i n x / L}, \quad (1)$$

⁰Written by Steven Troxler

gives a clear descriptions of the errors encountered through the use of rectilinear sampling and discrete approximations to the CFT, and the FFT gives a fast way to compute the approximate inverse. When we attempt to reconstruct from nonuniform samples, we have neither advantage: the aliasing patterns are largely unknown, depending in a subtle way on the sampling pattern and how we weight each sampling point in the reconstruction algorithm, and the FFT is of no help unless we somehow interpolate to a grid.

For the specific case of radial or polar samples, as in Figure 2 there is an alternative to Fourier reconstruction. In two-dimensional imaging the filtered backprojection algorithm, which we will discuss in more detail when we cover CT, gives highly accurate reconstructions from Radon transform data, which can be computed rapidly from radial samples of the continuous Fourier transform (CFT) via a 1D FFT, in $O(N^3)$ operations. This is not as good as the $O(N^2 \log N)$ computations needed for Fourier reconstruction from uniform samples, but far better than the $O(N^4)$ that would be necessary if we tried to use a Riemann sum to directly invert the Fourier data from a nonuniform sample. In addition, the filtered backprojection algorithm permits a great deal of parallelization, which allows its computation in considerably less time than the number of operations would suggest. For this reason, it is the most commonly used algorithm on commercial CT machines, although the algorithm we describe in this section has now become a viable alternative. MRI machines, unlike CT machines, are not typically designed to make use of this parallelization, so having a more efficient algorithm is more critical for the use of projection imaging in MRI.

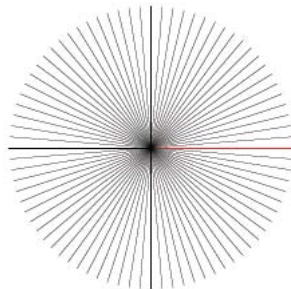


FIGURE 2. A polar scan, on which we could use filtered backprojection methods.

Perhaps the most obvious solution, both for projection imaging in MRI and for more general nonuniform samples, is to simply use linear interpolation around each uniform grid point, fitting some kind of linear least-squares approximation using the nearest sample points. Since the approximation near each uniform grid point would involve only a few of the nearest sample points, the linear interpolation should be fast, and afterward we could use the FFT. Particularly in light of the fact that our objective function has compact support, implying that its Fourier transform is C^∞ , we might expect this to produce good results, at least when the sampling pattern is dense enough. In fact, empirical studies indicate that this simple method does work if the nonuniform samples have spacing roughly twice the Nyquist rate and if we interpolate to a grid with half the Nyquist spacing. This, however, requires that

we collect 2^n times the data that would be necessary using uniform spacing, where $n = 2$ or 3 is the number of dimensions in which we are imaging. Since the whole goal of nonuniform sampling trajectories is to collect data faster, particularly in three dimensions, where sampling at twice the Nyquist rate would require 8 times the data, this solution is less than satisfactory.

Recall that there are two major issues in nonuniform sampling: (1) there is no fast algorithm for approximating an inverse Fourier transform using nonuniform samples, and (2) the errors from nonuniform sampling do not follow the predictable patterns indicated by the Poisson summation formula. The most common method used rapidly estimates a Riemann sum approximation to the inverse Fourier transform by using a convolution to quickly interpolate to a uniform grid and then using the FFT on that grid. Computing the Riemann sum directly is called the Direct Fourier transform (DFT), and is quite time consuming, while the fast interpolation method is known as *gridding*. We will look at the gridding algorithm in more detail later, but first note that it solves only the first problem; it gives a rapid method of estimating an approximate inverse transform. It does not, however, solve the second problem of making sure that the errors due to the approximate inverse are not large. The accuracy of the approximate inverse depends on how well we choose the weights on each point used in the Riemann sum. The system used to choose these weights is known as the *sampling density correction*.

In the first section below, we review some key facts about sampling theory. The theory of gridding is easiest to present using distribution theory, since distributions provide a link between discrete algorithms (which are essentially algorithms involving Dirac delta impulses) and continuous functions, so we will restate a number of familiar results, such as the Poisson summation formula, in terms of distributions. The next section discusses the problem of choosing an appropriate sampling density correction. Once we have done this, we can get an accurate approximate reconstruction using the direct Fourier transform. The final section discusses how to quickly estimate the direct Fourier transform using the gridding algorithm.

1.1. Review of Key Facts. Below, we list some facts about Fourier theory, sampling theory, and distribution theory which play major roles in the theory of sampling density correction and gridding:

- The convolution theorem states that $\widehat{f \cdot g} = \hat{f} * \hat{g}$ and $\widehat{\hat{f} \hat{g}} = f * g$, where ‘ \cdot ’ indicates pointwise multiplication and ‘ $*$ ’ indicates convolution. This formula holds true even if f is a tempered distribution and g is a function.
- The Fourier transform of the Dirac delta impulse δ is 1. In general, the Fourier transform of a linear combination of translates of deltas will be a nonharmonic Fourier series, that is, a sum of the form $\sum_j a_j e^{c_j x}$ where the c_j ’s are not proportional via rational constants. Although such a sum will not be periodic, we can see that it *will* be bounded and will *not* decay at infinity.
- Let the Shah or comb function III_d indicate an infinite sum of evenly spaced delta impulses with uniform spacing d . That is, if $d \in \mathbb{R}^n$ and we let $\cdot*$ and $\cdot/$ indicate pointwise multiplication and division of vectors in \mathbb{R}^n , then we have

$$\text{III}_d = \sum_{j \in \mathbb{R}^n} \delta_{j \cdot d}.$$

The Poisson summation formula is equivalent (in tandem with the convolution theorem) to the following:

$$\widehat{\text{III}}_d = \text{III}_{1/d}, \quad \text{or equivalently,} \quad (\text{III}_d \cdot \hat{f})^\vee = f * \text{III}_{1/d}, \quad (2)$$

where \vee indicates an inverse Fourier transform. This is the distributional formulation of the Poisson summation formula. This is a direct consequence of the Poisson summation formula (1), coupled with the fact that the Fourier transform of a distribution like III is determined its action on the Fourier transform of a function. See the notes on distributions for more detail.

The practical importance of this is that, assuming that \hat{f} has compact support, the Fourier transform $\widehat{\text{III}_d \cdot \hat{f}}$ can be computed via an FFT, since the delta impulses in III mean that the Fourier transform can be computed as a discrete sum.

From geometric considerations, given that the support of the convolution of two functions is the sum of their supports, we see that $\widehat{\text{III}_d \cdot \hat{f}} = f$ on the support $[-L, L]$ of f if and only if $c \leq \frac{1}{2L}$, so that $\frac{1}{c} \geq 2L$. If this is not clear, the reader is invited to draw a diagram and consider the effect of convolving with $\text{III}_{1/c}$ for a nonzero c ; the result is a replication of f every $1/c$ units, and these replications will be nonoverlapping if and only if the radius of the support of f is half of $1/c$. This result is called the Nyquist theorem, and the sampling rate it calls for is called the Nyquist rate.

If f is not supported in such a region $[-L, L]$ but is small and rapidly decaying outside such a region—we often say f is effectively supported in $[-L, L]$ —then $f * \text{III}_{1/c}$ will not be equal to f in the support of f , but will be very close. It is also worth pointing out that errors are unavoidable, since, even if f has compact support, \hat{f} will not, since the Fourier transform of a function with compact support never has compact support. The assumption, then, is that both f and \hat{f} have small effective support, whether or not either one is technically has compact support.

2. SAMPLING DENSITY CORRECTION

The development in this section is adapted from [4], [3], and [1].

Throughout this section and the next, capital letters will typically be used for functions in the frequency domain and the corresponding lowercase letters for their representations in the image domain, so that we will, for instance, write F for \hat{f} , where f is the objective function we wish to reconstruct from nonuniform samples of its Fourier transform F .

For readability, we will generally use notation as though the functions were one dimensional, $\mathbb{R} \rightarrow \mathbb{C}$, primarily because this allows us to use intervals for symmetric regions about zero, such as the field of view we wish to reconstruct. The statements we will make are valid for rectangles or, as is most commonly the case in practice, circles and spheres in \mathbb{R}^2 and \mathbb{R}^3 ; they are just not as easy to notate in these cases. Thus, when we write $[-a, a]$, it should be understood that this could be replaced by $[-a, a]^n$ for $n = 2$ or 3 , or by the circle or sphere of radius a .

2.1. The Direct Fourier Transform. The direct Fourier transform (DFT) reconstructs using the formula

$$f_W(x) = \sum_j W_j F(k_j) e^{2\pi i k_j \cdot x}, \quad (3)$$

here the x 's lie on a uniform grid and the k_j 's are the sample points, which will not generally lie on a uniform grid. If they do lie on a grid, then the W_j 's can be taken as constant, and the DFT is just a discrete Fourier transform, which can be rapidly computed via and FFT, and results in known, predictable errors that can be controlled by making sure that the sample spacing is small enough for the field of view and the amount of frequency space covered is sufficient to resolve the level of detail desired. In general, when the samples are nonuniform, a DFT approximates the inverse FT using a Riemann sum, and the W_j 's are the weights assigned to each sample point in that reconstruction.

The accuracy of the reconstruction, then, plays a critical role in the performance of the DFT, and the goal of sampling density correction is to find appropriate weights W_j . The subscript W explicitly indicates the dependence of the reconstruction on the weights W_j . Computing this sum using a DFT is a $O(N^4)$ process for an N by N image, since the number of samples k_j needs to be of the same order of magnitude as the number of grid points at which we reconstruct. But, if the direct Fourier transform results in a satisfactory image, we can use the gridding method described in the next section to estimate the direct transform in $O(N^2 \log N)$ operations, so finding the appropriate weights is a first step to a practical reconstruction method.

Intuitively, we need the correction because the sample points need to be weighted heavily wherever they are spread out and lightly wherever they are densely packed in order for the Riemann sum in the direct Fourier transform to be a good approximation to the continuous integral. This was not an issue in Cartesian sampling because the sampling points were evenly spread. To demonstrate the importance of the density correction, consider the following images:

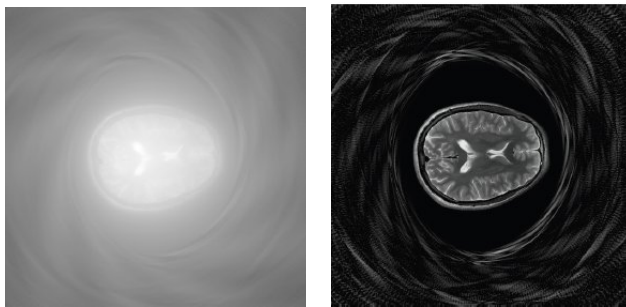


FIGURE 3. If we make no correction for sampling density in a typical spiral scan, the results look like (a). An accurate density correction gives much better results (b).

The image at left is what we get if we use a DFT without any density correction, while the image at right is using a DFT with an accurate density correction. Without any correction, the image is blurred, as though it was convolved with a

very poor approximation to the delta impulse. This, in fact, is exactly what has happened, as we will see in the next section.

Minimizing the error can be done in a number of ways. Various algebraic methods have been proposed, although many of them perform poorly in practice. Analytic methods have also been proposed. One method simply breaks up k-space into polyhedrons and takes the weight of each point to be the volume or area of that polyhedron [7]. This method is very flexible and can be used with any trajectory, but it generally leads to large errors. Another method is to use the Jacobian of a change of variables to estimate the density [2]. This leads to excellent results with certain trajectories, such as spirals, but does not work well on others like radial scans (in which the change of variables has a singularity), and does not work at all for many scans, such as PROPELLER scans, which cannot be parameterized smoothly.

The method we consider here, which was developed in [3], [6], and [5], minimizes error by viewing the errors resulting from the approximation in equation (3) in the image domain as a convolution using distribution theory.

2.2. Point Spread Function. Let us consider the situation more carefully. We have an objective function f , assumed to be compactly supported in a field-of-view (FOV) $[-L, L]$, which we wish to recover from a nonuniform sample of its Fourier transform F at the collection of points $\{k_j\}$ in frequency space. Again, we want an approximate reconstruction of the form

$$f_W(x) = \sum_j W_j F(k_j) e^{2\pi i k_j \cdot x} \quad (4)$$

on a uniform grid.

In order to choose appropriate weights for the sample, we need to understand the basic properties of f_W . To see these, we rewrite f_W as an inverse CFT using distributional notation. Specifically, let

$$W = \sum_j W_j \delta_{k_j},$$

where δ_{k_j} indicates a translation of the Dirac delta impulse to k_j . Then f_W can be written as

$$f_W = \mathcal{F}^{-1}(W \cdot F) = w * f, \quad (5)$$

where \cdot indicates pointwise multiplication, \mathcal{F}^{-1} indicates the inverse CFT (in a distributional setting), $*$ indicates convolution, w is the (distributional) Fourier transform of W , and the second equality is a consequence of the convolution theorem. According to equation (5), then, the accuracy of our approximation f_W depends on w .

This function w is called the *point spread function* (PSF). The name is because, if $f = \delta$ were a point mass function, then f_W would be spread out to $w * \delta = w$. One can see from (5) that w determines the success of the DFT; if w behaves like a δ within a suitable region, as we will see below, then the reconstruction will be perfect. This is not possible in practice, but the closer we can get, the better the reconstruction.

Since $W = \sum_j W_j \delta_{k_j}$ is a linear combination of translates of delta impulses and $\hat{\delta} \equiv 1$, the basic linearity and translation properties of the Fourier transform imply

that w takes the form

$$w = \sum_j W_j e^{-2\pi i x \cdot k_j},$$

which is called a *nonharmonic Fourier series* because it is a sum of complex exponentials, but in general the periods of those exponentials will not have rational ratios with one another so that the sum will not be periodic. Although w need not be periodic, neither will it decay at infinity. This means that, even if f has compact support (as is usually the case in medical imaging), not only will f_w fail to have compact support, f_W will not in general even decay at infinity. This observation will be very important in the next section on gridding, since any attempt to use the FFT to obtain f_W risks aliasing these large sidelobes of f_W into the field of view, which would lead to a completely unusable image.

The discussion in the preceding paragraph shows that we cannot hope to have $f \approx f_w$ everywhere, but, in order to get an accurate approximation to the desired image using the DFT, we only need f_W to approximate f within the field of view. The errors elsewhere are irrelevant (later, when we introduce the gridding algorithm, we will see that we need $f_w \approx f$ on a slightly larger region, but the methods of achieving this are the same. Our goal, then, is to choose w to minimize error in the field of view. This means choosing w to approximate δ , at least within some region.

2.3. Iterative Solution Using a Convolution. Suppose, for a moment, that we could have $w = \delta$ on the region $[-2L, 2L]$. Then $f_W = f$ would hold exactly on $[-L, L]$, and the DFT would give a perfect reconstruction. In fact, the only way to guarantee this is to have $w = \delta$ on $[-2L, 2L]$, since the value of w at every point in this region appears in the convolution defining f_W somewhere in the field of view. One way to approach the problem of choosing a sampling density, then, is to try to minimize the error in this point spread function

$$e = w - \delta$$

over $[-2L, 2L]$, which is the same as minimizing the error

$$e = p \cdot w - \delta \tag{6}$$

everywhere, where $p = \chi_{[-2L, 2L]}$ is the *error weighting profile*, which determines how important we consider errors in the PSF w . If we use a characteristic function for p , we are weighting all errors within the region of optimization equally, but ignoring errors elsewhere.

A more general approach would minimize (6) for some p which is positive on $[-2L, 2L]$ and zero everywhere else, so that errors within $[-2L, 2L]$ are penalized and errors outside that region are not. The Fourier domain representation of the error (6) is

$$E = P * W - 1, \tag{7}$$

so we may equivalently attempt to minimize the error in (7). This cannot be accomplished everywhere (since W has compact support), but should intuitively occur at least at the sampling points where the delta impulses in W are located. One method of accomplishing this is to take

$$W = \frac{S}{S * P}, \tag{8}$$

where

$$S = \sum_j \delta_{k_j}$$

is the uncorrected sampling function [3]. Intuitively, the denominator will be large where $P * S$ is too large and small where $P * S$ is too small, so that $W * P$ will be closer to 1 at the sample points. An iterative solution improves on this by taking the weight obtained in (8) to be W_1 and then iteratively defining

$$W_i = \frac{W_{i-1}}{P * W_{i-1}}. \quad (9)$$

The process after an ‘optimal’ number of iterations, best determined experimentally [6], since there is some question of whether the algorithm converges in a technical sense. An important consideration is that P needs to have small effective support, because otherwise the convolution will have a large number of nonzero terms and will take too long to compute. The in Figure 4 graphically demonstrate the use of this method.

A comparison of the original unweighted sampling function and the resulting profile are shown in Figure 5. The sample points near the center, where the sampling is dense, are weighted much less than those near the edges, where the sampling is spread out. Within the area of optimization, the point spread function of the re-weighted samples approximates a delta impulse much better. In realistic images, the improvement is more dramatic than here, where everything is scaled to be easy to see.

The justification for the algorithm in equation (9) is conceptual rather than rigorous, and experimental evidence indicates that (9) does not technically converge, either due to numerical instability or due to problems with the algorithm itself. Nonetheless, empirical studies using root mean-square error indicate that the algorithm will provide a w which does a relatively good job of minimizing the error $p \cdot w - \delta$, and, with a proper choice of p , this algorithm provides the best results of any existing method, with the advantage that it works on arbitrary trajectories.

2.4. Choosing an Optimal Error Weighting Profile. Equation (9) leads to a natural question: what is the ideal choice of p ? Since the goal is to minimize error over $[-2L, 2L]$, a natural choice would be $p = \chi_{[-2L, 2L]}$, but this leads to two problems. The first is that, given this choice of p , its Fourier transform P will not have small support, so the convolution in the denominator will take a very long time to compute. The second is that not all parts of the point spread function w are manifested with equal frequency in the field of view, and an ideal choice of p should penalize most heavily those errors which occur often. It turns out that solving the second problem also solves the the first, because the optimal choice of p is continuous and differentiable almost everywhere, so that its transform P has small sidelobes, which allows it to be truncated without large errors.

Since f has compact support and we are only interested in the convolution $w * f$ within that support, the values of w closer to the origin in the point spread function will appear as nonzero terms in convolutions defining values of f_W within the field of view much more often than those near $\pm 2L$. The diagrams in Figure 6 below illustrate this.

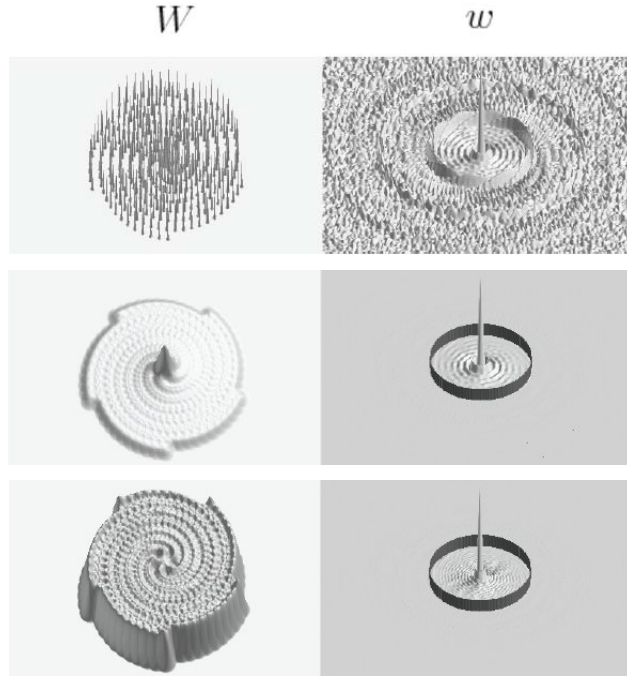


FIGURE 4. These images, which plot the Fourier domain at left and the image domain at right, display the optimization process. If we do not weight the samples, that is, take $W = S$, then the point spread function w will be a poor approximation to the delta impulse (a). In order to find more appropriate weights, we convolve W with P in the frequency domain, which is equivalent to multiplying w by the error weighting profile p in the image domain (b). Then, to find more appropriate weights, we divide S (or W_i , in the iterative algorithm) by the convolution's value at each point, which levels out the density in the Fourier domain, and in the image domain results in a point spread function which better approximates the delta impulse within the region of optimization (c).

In fact, the frequency with which errors in the PSF w appear in the final image will be proportional to the value taken at that point by $\chi_{[-L,L]} * \chi_{[-L,L]}$, the convolution of the characteristic function of the field of view with itself, as shown in Figure 7.

This observation, it turns out, solves both problems, because if we take $p = \chi_{[-L,L]} * \chi_{[-L,L]}$, then we will have $P = (\widehat{\chi_{[-L,L]}})^2$. Given that $\widehat{\chi_{[-L,L]}}$ decays like a sinc function, its square will decay rapidly enough to truncate with comparatively little error, as shown in Figure 8.

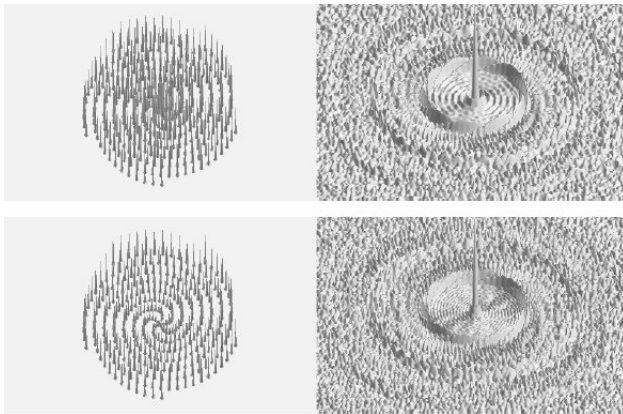


FIGURE 5. The images above show the sampling functions in Fourier space at left and the point spread functions in physical space at right with and without density correction

With this kernel, the algorithm in (9) can be computed quickly, and the resulting sampling density corrections are very accurate, as the image at the beginning of this section showed.

2.5. The Gridding Algorithm. Gridding is by far most popular way to deal with nonuniform samples in recent years because it is highly accurate and runs in the ideal $O(N^2 \log N)$ operations. Recall that the DFT computes an approximate reconstruction of the form

$$f_W(x) = \sum_j W_j F(k_j) e^{2\pi i k_j \cdot x} \quad (10)$$

The idea of gridding is to estimate this sum quickly by converting observations on the nonuniform sample points $\{k_j\}$ to samples on a uniform grid $\{\xi_j\}$ in the frequency domain via a convolution. That is, we choose a gridding kernel C (an ordinary function not a distribution) and we compute

$$C * (F \cdot W)(\xi_l) = \sum_j W_j F(k_j) C(\xi_l - k_j). \quad (11)$$

The sum in (11) is finite, so it can be computed exactly. For a general kernel C , this convolution would be time-consuming to compute, requiring about N^4 computations for an N by N image, which is no improvement on the DFT. The *raison d'être* for gridding is that, if C has small effective support, sum defining the convolution will include only a few nonzero terms, and the rest can be ignored. If the support is small enough, the time required will be roughly proportional to the number of grid points, N^2 for an $N \times N$ image.

Once we perform gridding, we will have the values of $C * (F \cdot W)$ on a uniform grid, or in distributional terms, we will have $C * (F \cdot W) \text{III}_d$ where d is the grid spacing. We then perform an FFT. When we do this, it follows from the convolution Theorem and the Poisson summation formula that our result is

$$[c \cdot f_W] * \text{III}_{1/d}, \quad (12)$$

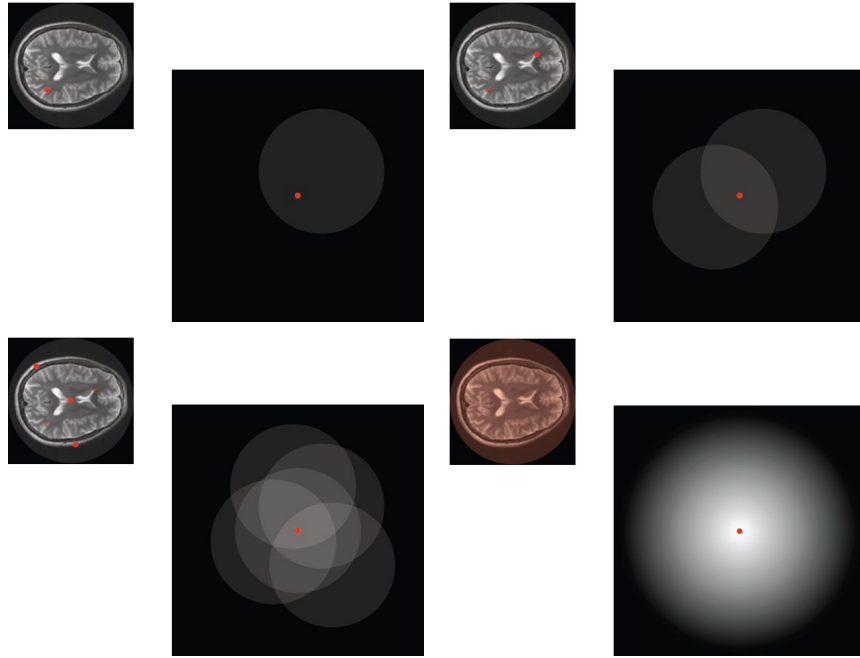


FIGURE 6. These images graphically show how we find the optimal error weighting profile. If we pick a particular point in the final image, (a) shows the region of the PSF which contributes nonzero terms to the convolution defining F_W at that point. Doing the same with another point, we get (b), and with several more, we get (c). The pattern at this point should be clear: locations near the center of the point spread function w are expressed more often in the final image than points farther away. An optimal error weighting profile should penalize errors in accordance with how often they appear, so, by looking at a continuum of points in the FOV, we find the ideal profile (d).

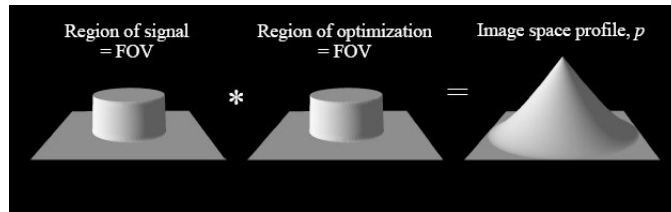


FIGURE 7. The ideal profile is the convolution of the characteristic function of the FOV with itself.

where c , often called the apodizing function or apodization, is the Fourier transform of C . We can see from (12) that there are two conditions that c needs to satisfy in order for this reconstruction to work. The first is obvious: c needs to be bounded

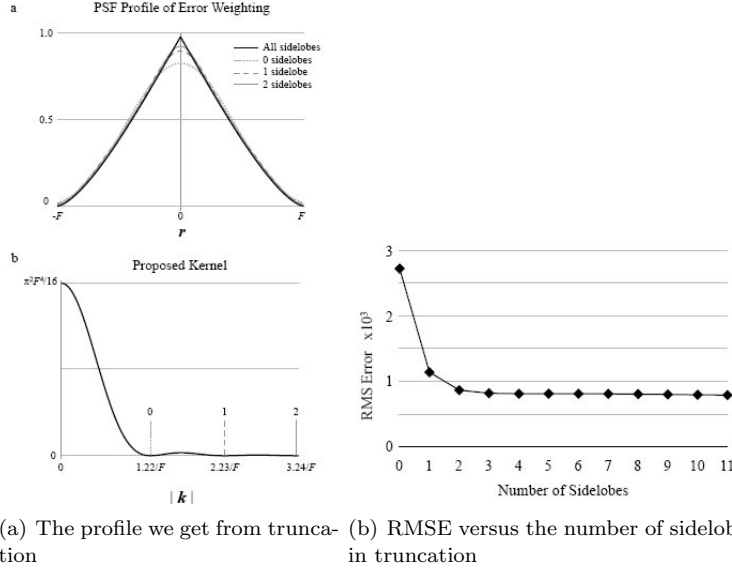


FIGURE 8. The diagrams above demonstrate that we can truncate the kernel to two sidelobes, which leads to small effective support, and still come out with very small errors.

away from zero in the field of view $[-L, L]$. So long as this is true, we can undo the convolution by dividing by c or *deapodizing* within the field of view, and we will recover f_W plus any aliasing that results from the FFT.

This aliasing leads to the second criterion on c . Recall that we have assumed that f has compact support in $[-L, L]$, and that $f_W \approx f$ in that region. From our discussion in the previous section, we know that f_W does not have compact support, and indeed does not even go to zero. We do, however, know that f_W is bounded, so that $c \cdot f_W$ will converge to zero as c does. Now, recall from our review of the Poisson Summation Formula and the related Nyquist criterion that the FFT gives (essentially) aliasing-free results whenever $d < \frac{1}{2a}$, where the function we wish to reconstruct is (essentially) supported in $[-a, a]$. Putting these facts together, we see that, if we convolve onto a grid with spacing d , then c needs to be essentially supported in $[-\frac{1}{2d}, \frac{1}{2d}]$ in order to mitigate the aliasing caused by nonuniform sampling¹.

In order to reconstruct a given n -dimensional image with a certain spatial resolution, the amount of data we need will be proportional to $(\frac{1}{d})^n$. Making d as large as possible, then, that is, making a small as possible, minimizes the amount of data we have to collect and process. If this were the only consideration, then it would follow from the fact that c needs to be bounded away from zero on $[-L, L]$ that the ideal choice is $a = L$, so that c would be a characteristic function or something

¹By the way, I got confused in class and stated that we also needed f_W to approximate f in this larger region, which means w needs to approximate δ over a bigger region. This is not true; we only need $f \approx f_W$ on the support $[-L, L]$ of f . Suppression of aliasing error is a separate matter having to do with the interaction between C and the sampling density.

of the sort. The problem with this is that if we take c to be such a function, c will have a jump discontinuity, so its Fourier transform C will behave similarly to a sinc function. It will not have small effective support, so we will not be able to truncate the sum in (11), which defeats the purpose of gridding in the first place, since our goal was to quickly approximate the sum in (3).

The solution is to instead take $a = \lambda L$ for some $\lambda > 1$, called the oversampling ratio, creating a *passband* $[-L, L]$ where c is bounded away from zero so that we can divide by c and recover f_W , and a *stopband* where $c \rightarrow 0$ in a smooth fashion, which allows us to design c so that its frequency domain representation C will have compact effective support. The traditional oversampling ratio is 2, largely because the common FFT method requires a vector of length equal to some power of 2, and the image typically has sides also equal to a power of two. Moreover, it is comparatively easy to design a kernel C with very small effective support when the oversampling ratio is this large. With such a large oversampling ratio, however, we have to process 2^n times as much data for an n dimensional image as would be required with uniform sampling, which particularly in 3D imaging can considerably slow computation and place major stress on computer memory.

Recently new FFT algorithms have been developed that work just as quickly on vectors whose length is not 2, making lower oversampling ratios more workable [1]. In order to design a kernel so that its transform c will have a narrower stopband relative to the passband, the support of C generally has to be made slightly larger so that the gridding itself, given by equation (11), will take longer. This problem can be largely, although not completely, mitigated by pre-sampling C^2 and using linear interpolation during the calculation of (11), so that the evaluation of $C(\xi_j - k_j)$ takes very little time. By using such a pre-sampled kernel and making an optimal trade-off between trying to minimize the kernel width and minimize the oversampling ratio, we can speed the process significantly. Empirical studies indicate that oversampling ratios between 1.3 and 1.5 produce excellent results much more quickly than an oversampling ratio of 2.

Figure 8 demonstrates the cropping process. Note that the final image is basically indistinguishable from that obtained using the direct Fourier transform.

REFERENCES

- [1] Philip J. Beatty, Dwight G. Nishimura, and John M. Pauly. Rapid gridding reconstruction with a minimal oversampling ratio. *IEEE Trans. Med. Imag.*, 24(6):799–808.
- [2] R.D Hodge, R.K.S. Kwan, and G.B. Pike. Density compensation functions for spiral MRI. *Magnetic Resonance in Medicine*, 38:117–128.
- [3] J. Jackson, C. Meyer, D. Nishimura, and A. Macovski. Selection of a convolution function for fourier inversion using gridding. *IEEE Trans. Med. Imag.*, 10(3):473–478.
- [4] J. O’sullivan. A fast sinc function gridding algorithm for fourier inversion in computerized tomography. *IEEE Trans. Med. Imag.*, MI-4:200–207.
- [5] J.G. Pipe and Kenneth Johnson. Convolution kernel design and efficient algorithm for sampling density correction. *Preprint*.
- [6] J.G. Pipe and P. Menon. Sampling density compensation in MRI: Rationale and an iterative numerical solution. *Magnetic Resonance in Medicine*, 41:799–808.
- [7] V. Rasche, R. Proksa, R. Sinkus, P. Bornert, and H. Eggers. Resampling of data between arbitrary grids using convolution interpolation. *IEEE Trans. Med. Imag.*, 18:427–434.

²This would not be necessary if C were some easily calculated function, but the evaluation of the Kaiser-Bessel windows typically used for C requires several computations. When we do this thousands of times in the gridding process, it can considerably slow the overall algorithm.

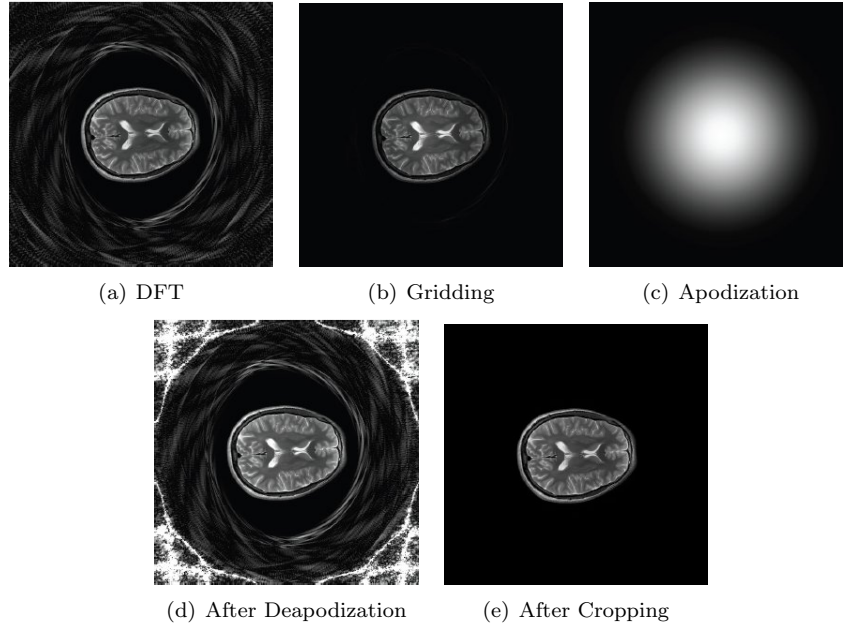


FIGURE 9. These images give a DFT reconstruction for comparison, and show the steps of the gridding algorithm: The first image, in (a), is a reconstruction using a DFT. The image in (b) is a reconstruction using gridding with a Kaiser-Bessel function without deapodizing. Note that the apodization (c) leads to darker edges of the image. When we deapodize, getting the image in (d), the aliasing outside the FOV is greatly magnified, but this is irrelevant when we crop the image to the FOV (e).