JDSSV

Journal of Data Science, Statistics, and Visualisation

MMMMMMM YYYY, Volume VV, Issue II. *doi: XX.XXXXX/jdssv.v000.i00*

Compressed Sensing with a Jackknife, a Bootstrap, and Visualization

Aaron Defazio
Facebook Artificial Intelligence Research

Mark Tygert
Facebook AI Research

Rachel Ward
University of Texas at Austin

Jure Zbontar
Facebook AI Research

Abstract

Compressed sensing proposes to reconstruct more degrees of freedom in a signal than the number of values actually measured (based on a potentially unjustified regularizer or prior distribution). Compressed sensing therefore risks introducing errors — inserting spurious artifacts or masking the abnormalities that medical imaging seeks to discover. Estimating errors using the standard statistical tools of a jackknife and a bootstrap yields error "bars" in the form of full images that are remarkably qualitatively representative of the actual errors (at least when evaluated and validated on data sets for which the ground truth and hence the actual error is available). These images show the structure of possible errors — without recourse to measuring the entire ground truth directly — and build confidence in regions of the images where the estimated errors are small. Further visualizations and summary statistics can aid in the interpretation of such error estimates. Visualizations include suitable colorizations of the reconstruction, as well as the obvious "correction" of the reconstruction by subtracting off the error estimates. The canonical summary statistic would be the root-mean-square of the error estimates. Unfortunately, colorizations appear likely to be too distracting for actual clinical practice in medical imaging, and the root-mean-square gets swamped by background noise in the error estimates. Fortunately, straightforward displays of the error estimates and of the "corrected" reconstruction are illuminating, and the root-mean-square improves greatly after mild blurring of the error estimates; the blurring is barely perceptible to the human eye yet smooths away background noise that would otherwise overwhelm the root-mean-square.

Keywords: error estimation, visualization, medical imaging, magnetic resonance imaging, compressive sensing, compressive sampling.

1. Introduction

Compressed sensing is the concept that many interesting signals are recoverable from undersampled measurements of the representations of those signals in a special basis. A widely touted potential application is to the acceleration of magnetic resonance imaging (MRI), as by Lustig et al. (2007). In MRI, the special basis for representations of signals is the Fourier basis, and the goal of compressed sensing is to recover high-resolution images from relatively sparse measurements of the Fourier components of those images. Here, "sparse" means substantially fewer measurements of values in the Fourier domain than the numbers of pixels in the reconstructed images. Of course, recovering more degrees of freedom than the number of measured values is an ill-posed problem, yet has been rigorously proven to be solvable when the gradients of the images being recovered are known to be small except at a few pixels, for instance, when edges dominate the images (the proofs of Candes et al. (2006) are especially well-known). This recovery is still non-trivial, as the small number of pixels where the gradients are non-trivial very well may vary from image to image, while the same reconstruction procedure works irrespective of where the gradients concentrate (so long as they concentrate on sparse subsets of all pixels in the reconstructed domain). The requirement that gradients be concentrated on sparse subsets is sufficient but may not be necessary, and much recent research — including that of Hammernik et al. (2018) — aims to generalize beyond this requirement by applying machine learning to representative data sets. Indeed, the literature on compressed sensing is vast and growing rapidly; see, for example, the recent review of Tropp (2017) for explication of all this and more.

Needless to say, compressed sensing risks introducing errors into the resulting reconstructions, especially if the assumption of sparsity is unfounded for the real data at hand. The works of Malioutov et al. (2010) and Ward (2009) quantify these errors via a single scalar estimate of confidence in the reconstruction, namely, an estimate of the mean-square error. The present paper extends these methods, producing estimates of the entire image displaying the discrepancy between the reconstruction in compressed sensing and the actual ground truth. Of course, compressed sensing takes too few measurements to ascertain the actual ground truth, so only an estimate of the discrepancy — an error "bar" in the form of an image — is possible. However, the examples of the present paper show that "jackknife" and "bootstrap" estimates of the errors are reasonably representative of the reality, at least for the cases in MRI tested here, in which the ground truth is available for comparison and evaluation. Those unfamiliar with the jackknife and the bootstrap may wish to consult Efron and Tibshirani (1993); that said, the presentation below is completely self-contained, not presuming any prior knowledge of either the jackknife or the bootstrap. The jackknife and bootstrap images highlight when, where, and what errors may have arisen in each reconstruction from compressed sensing for MRI, tailored to the specific object being imaged.

The jackknife is similar to standard *a posteriori* tests for convergence of numerical methods; such tests for convergence often serve as proxies for estimates of accuracy. The bootstrap leverages more extensive computation, simulating measurements that could have been taken but were not in fact (recall that compressed sensing involves taking fewer measurements than the number of degrees of freedom being reconstructed). The bootstrap simulates plausible alternative reconstructions from hypothetical measurements that are consistent with the reconstruction from the measurements actually made. The alternative reconstructions fluctuate around the reconstruction from the measurements actually made; the fluctuation is an estimate of the error, when averaged over various sampling patterns for the measurements being considered.

The present paper also investigates user-friendly methods for generating visualizations and automatic interpretations of these error estimates, appropriate for display to medical professionals (especially radiologists). After testing several natural visual displays, we find that any nontrivial visualization is likely to be too distracting for physicians, as some have expressed reservations about having to look at any errors at all — they would be much happier having a machine look at the estimates and flag potentially serious errors for special consideration. We might conclude that colorization is too distracting, that the best visualizations are simple displays of the error estimates, possibly supplemented with the error estimates subtracted from the reconstructions (thus showing how the error estimates can "correct" the reconstructions). Most of the results of the present paper about visualization could be regarded as negative, however natural and straightforward the colorizations may be.

For circumstances in which visualizing errors is overkill (or unnecessarily bothersome), we find that an almost simplistic automated interpretation of the plots of errors — reporting just the root-mean-square of the denoised error estimates — works remarkably well. While background noise dominates the root-mean-square of the initial, noisy error estimates, even denoising that is almost imperceptible can remove the obfuscatory background noise; the root-mean-square can then focus on the remaining errors, which are often relatively sparsely distributed. When the root-mean-square of the denoised error estimates is large enough, a clinician could look at the visualizations mentioned above to fully understand the implications of the error estimates (or rescan the patient using a less error-prone sampling pattern).

By reducing the time required to scan patients, compressed sensing promises to lower costs, improve patients' experience, reduce artifacts due to motion, and extend imaging to patients who have trouble staying still for long periods, *inter alia*.

The structure of the remainder of the present paper is as follows: First, Section 2 introduces the jackknife and the bootstrap for compressed sensing, together with methods for their visualization and automated summarization in scalar statistics. Then, Section 3 illustrates the performance of the methods on data sets from MRI, with copious additional examples provided in the appendices. Finally, Section 4 discusses the results introduced in Section 3 and draws some conclusions.

The Python package **fbooja** reproduces all our results from Section 3 and the appendices and is available at `https://github.com/facebookresearch/fbooja` (that GitHub repository also includes a re-distribution of the MRI scans of a patient's head from the data of Loizou et al. (2013a), Loizou et al. (2011), Loizou et al. (2013b), and Loizou et al. (2015)). The software is available with a permissive (MIT) open-source license, though the package is meant solely for prototyping and demonstrating a proof of principle — the package may not be industrial-strength or suited for deployment directly to clinical practice; the software makes no attempt to implement fail-safes or be fool-proof, but rather is most appropriate for use in research or as a template for the development of production-ready implementations.

2. Methods

2.1. A jackknife and a bootstrap

We denote by X a data set $(x_i)_{i \in I}$, where each x_i is a scalar or a vector and I is a set of indices. We consider a vector-valued (or image-valued) function $f = f(X, S)$ of both X and a subset S of the index set I such that the value of f depends only on $(x_i)_{i \in S}$. Compressed sensing approximates the full $f(X, I)$ with $f(X, S)$, where S is a subset of I collecting together independent uniformly random draws from I, perhaps plus some fixed subset T of I. (Obviously, this construction makes T a subset of S. However, T need not be disjoint from the set of independent uniformly random draws.)

In compressed sensing for MRI, measured observations in the Fourier domain of the object being imaged are $(x_i)_{i \in I}$, and $f(X, I)$ uses those measurements to reconstruct the object in the original domain (hence involving an inverse Fourier transform to map from X to $f(X, I)$). The reconstruction $f(X, S)$ from a subset S of I commonly involves minimizing a total-variation objective function or deep learning of some sort, as discussed by Tao and Yang (2009), Yang and Zhang (2011), Hammernik et al. (2018), and their references.

With such undersampled measurements, the reconstruction is oblivious to much of the Fourier domain, sampling fewer measurements than at the usual Nyquist rate. We will tacitly be assuming that the procedure for reconstruction works not only for the set S specifying the measurements actually used, but also for other sets of random observations, that is, for other random realizations of S. For machine-learned reconstructions, the model for reconstruction must train on measurements taken from many different possible samplings, not just one, as otherwise the model will be blind to parts of the Fourier domain. If we can simulate on a computer what could have happened with measurements that we do not take in reality, then we can construct error "bars" highlighting when, where, and what might have gone wrong in a reconstruction from actual measurements taken with only one realized sampling set S. The computational simulation allows us to gauge what could have happened with unseen measurements. While seeing the unseen (at least in part) may seem counterintuitive, in fact the field of statistics is all about what might have occurred given observations of what actually did happen. The bootstrap defined below follows this prescription literally. The jackknife is a somewhat simpler formulation.

The goal of both the jackknife and the bootstrap is to provide an estimated bound on $f(X, S) - f(X, I)$, without having access to the full reconstruction $f(X, I)$. (The full reconstruction depends on all measurements $(x_i)_{i \in I}$ — the whole I — so is unavailable when performing compressed sensing.)

First, we define the jackknife error "bar" for f on S to be

$$d = 2 \sum_{i \in S \backslash T} \Big(f(X, S \backslash \{i\}) - f(X, S) \Big), \tag{1}$$

where the sum ranges over every index $i \in S$ such that $i \notin T$, and $S \backslash \{i\}$ is just S after removing i. The jackknife d defined in (1) characterizes what would happen to the output of f if the input S were slightly smaller; if $f(X, S)$ is close to converging on

$f(X, I)$, then $f(X, S \backslash \{i\})$ in (1) should also be close to $f(X, I)$, so $f(X, S \backslash \{i\})$ should be close to $f(X, S)$, aside from errors. We refer to $f(X, S \backslash \{i\}) - f(X, S)$ in the right-hand side of (1) as a "leave-one-out" difference, as in "leave-one-out" cross-validation. We could empirically (or semi-empirically) determine a calibration constant c such that cd becomes of the same size as the actual discrepancy $f(X, S) - f(X, I)$ on average for a training set of exemplars (the training set could consist of many different X together with the corresponding $f(X, S)$ and $f(X, I)$). We found that $c = 1$ works well in the experiments reported below.

Next, we define the bootstrap, assuming that S is the union of the set T and a set of ℓ independent uniformly random samples from I (where ℓ is a parameter, and the number of distinct members of the latter set may be less than ℓ due to repetition in the ℓ samples): First, having already computed $f(X, S)$, we solve for $\tilde{X} = (\tilde{x}_i)_{i \in I}$ such that

$$f(\tilde{X}, I) = f(X, S). \tag{2}$$

Then, we form the set R of ℓ independent uniformly random draws from I (not all ℓ of which need be distinct), plus the fixed subset T of I (in so-called parallel MRI, as described by Brown et al. (2014), T would naturally contain all the lowest frequencies). We select a positive integer k and repeat this resampling independently k times, thus obtaining sets R_1, R_2, ..., R_k. We define the bootstrap error "bar" to be

$$e = \frac{3}{k} \sum_{j=1}^{k} \Big(f(\tilde{X}, R_j) - f(\tilde{X}, I) \Big). \tag{3}$$

We could say that $f(\tilde{X}, R_j)$ arises from $f(\tilde{X}, I)$ in the same way as $f(X, S)$ arises from $f(X, I)$, having constructed $\tilde{X}$ assuming that $f(X, S)$ is "correct" in the sense of (2); so the summand in (3) is a proxy for the actual error $f(X, S) - f(X, I)$ (and the averaging over independent realizations reduces noise). We have no reason to believe that the scaling 3 in (3) is the ideal factor, but 3 seems to work well in the experiments reported below.

Remark 1. Both $\Big(f(X, S \backslash \{i\}) - f(X, S) \Big)_{i \in S \backslash T}$ from (1) and $\Big(f(\tilde{X}, R_j) - f(\tilde{X}, I) \Big)_{j=1}^{k}$ from (3) span whole spaces of errors that potentially could have happened given the actually observed measurements. (Here, "potentially" refers to being consistent with what the jackknife or bootstrap can generate.) While the sum and average in (1) and (3), respectively, of these sets of differences characterize leading modes of these spaces, principal component analysis can characterize all modes. However, looking at even just the leading modes seems somewhat overwhelming already; having to investigate more modes could really try the patience of a physician interpreting MRI scans, for instance. The present paper focuses on the leading modes observed in our experiments (namely, the sum or average).

2.2. Visualization in grayscale and in color

We include four kinds of plots displaying the full reconstructions and errors:

1. "Original" is the original grayscale image.

2. “Reconstruction” is the reconstruction via compressed sensing.

3. “Error of Reconstruction” displays the difference between the original and reconstructed images, with black (or white) corresponding to extreme errors, and middling grays corresponding to the absence of errors.

4. “Bootstrap” displays the errors estimated via the bootstrap, with black (or white) corresponding to extreme errors, and middling grays corresponding to the absence of errors.

We visualize the errors in reconstruction and the bootstrap estimates using grayscale so that the phases of oscillatory artifacts are less apparent; colorized errors look very different for damped sine versus cosine waves, whereas the medical meaning of such waves is often similar.

We consider four methods for visualizing the effects of errors (estimated via the bootstrap) simultaneously with displaying the reconstruction, via manipulation of the hue-saturation-value color space described, for example, by van der Walt et al. (2014):

1. “Reconstruction – Bootstrap” is literally the bootstrap error estimate subtracted from the reconstruction, in some sense “correcting” or “enhancing” the reconstruction.

2. “Errors Over a Threshold Overlaid” identifies the pixels in the bootstrap error estimate whose absolute values are in the upper percentiles (the upper two percentiles for horizontally retained sampling, the upper one for radially retained sampling), then replaces those pixels (retaining all other pixels unchanged) in the reconstruction with colors corresponding to the values of the pixels in the bootstrap. Specifically, the colors plotted are at the highest value possible and fully saturated, with a hue ranging from cyan to magenta, with blue in the middle (however, as we include only the upper percentiles, only hues very close to cyan or to magenta actually get plotted). This effectively marks the pixels corresponding to the largest estimated errors with eye-popping colors, leaving the other pixels at their gray values in the reconstruction.

3. “Bootstrap-Saturated Reconstruction” sets the saturation of a pixel in the reconstruction to the corresponding absolute value of the pixel in the bootstrap error estimate (normalized by the greatest absolute value of any pixel in the bootstrap), with a hue set to red or green depending on the sign of the pixel in the bootstrap. The value of the pixel in the reconstruction stays the same. Thus, a pixel gets colored more intensely red or more intensely green when the absolute value of the pixel in the bootstrap is large, but always with the value in hue-saturation-value remaining the same as in the original reconstruction; a pixel whose corresponding absolute value in the bootstrap is relatively negligible stays unsaturated gray at the value in the reconstruction.

4. “Bootstrap-Interpolated Reconstruction” leaves the value of each pixel at its value in the reconstruction, and linearly interpolates in the hue-saturation plane between green and magenta based on the corresponding value of the pixel in the

bootstrap error estimate (normalized by the greatest absolute value of any pixel in the bootstrap). Pure gray is in the middle of the line between green and magenta, so that any pixel whose corresponding error estimate is zero will appear unchanged, exactly as it was in the original reconstruction; pixels whose corresponding error estimates are the largest have the same value as in the reconstruction but get colored magenta, while those whose corresponding error estimates are the most negative have the same value as in the reconstruction but get colored green.

2.3. Summarization in a scalar

The square root of the sum of the squares of slightly denoised error estimations summarizes in a single scalar the overall size of errors. Even inconspicuous denoising can greatly improve the root-mean-square: While the effect of blurring the bootstrap error estimates with a normalized Gaussian convolutional kernel of standard deviation one pixel is almost imperceptible to the human eye (or at least preserves the semantically meaningful structures in the images), the blur helps remove the background of noise that can otherwise dominate the root-mean-square of the error estimates. The blur largely preserves significant edges and textured areas, yet can eliminate much of the perceptually immaterial zero-mean background noise. Whereas background noise can overwhelm the root-mean-square of the initial, noisy bootstrap, the root-mean-square of the slightly blurred bootstrap captures the magnitude of the more important features in the error estimates.

3. Results

The examples of this section illustrate the most commonly discussed compressed sensing for MRI, in which we reconstruct an image from measured observations of some of its values in the Fourier domain — "some" meaning significantly less than usually required by the Nyquist-Shannon-Whittaker sampling theory. To reconstruct an image from measurements taken in the Fourier domain (with independent and identically distributed centered complex Gaussian noise of standard deviation $0.02\sqrt{2}$ added to mimic machine imprecision), we minimize the sum of deviations from the measurements plus a total-variation regularizer via Algorithm 1 at the end of Section 2.2 of Tao and Yang (2009) (which is based on the work of Yang and Zhang (2011)), with 100 iterations, using the typical parameter settings $\mu = 10^{12}$ and $\beta = 10$ (μ governs the fidelity to the measurements taken in the Fourier domain, and β is the strength of the coupling in the operator splitting for the alternating-direction method of multipliers). As discussed by Tropp (2017), this is the canonical setting for compressed sensing. All computations take place in IEEE standard double-precision arithmetic. We use $k =$ 1,000 resamplings for the bootstrap in (3). The Python package **fbooja** reproduces our results and is available at https://github.com/facebookresearch/fbooja . . . **fbooja** builds upon PyTorch of Paszke et al. (2019), which is available along with instructions for installation of dependencies at https://pytorch.org

We consider two kinds of sampling patterns, radially retained and horizontally retained. All sampling takes place on an $m \times n$ Cartesian grid, allowing direct use of the fast

Fourier transform for acceleration of the reconstruction (as described by Tao and Yang (2009)). Future implementations could consider sampling off the grid, too.

With radially retained sampling, each x_i in our data set $X = (x_i)_{i \in I}$ consists of all pixels on an $m \times n$ Cartesian grid in the Fourier domain that intersect a ray emanating from the origin (each angle corresponds to x_i for a different index i). Figure 1 displays four examples of uniformly random subsets of X, sampling the angles of the rays uniformly at random. For radially retained sampling, we refrain from supplementing the subsampled set S with any fixed subset; that is, the set T is empty. To construct S, we generate $\frac{m+n}{5}$ angles uniformly at random (rounding $\frac{m+n}{5}$ to the nearest integer), which makes the errors easy to see in the coming figures, yet not too extreme.

With horizontally retained sampling, each x_i in our data set $X = (x_i)_{i \in I}$ consists of a horizontal line n pixels wide on an $m \times n$ Cartesian grid in the Fourier domain, with I consisting of the m integers from $-\frac{m}{2}$ to $\frac{m-2}{2}$. The subsampled set S always includes all horizontal lines ranging from the $-\sqrt{2m}$th lowest frequency to the $\sqrt{2m}$th lowest frequency (rounding $\sqrt{2m}$ to the nearest integer); that is, the set T consists of these low-frequency indices. To construct the remainder of S, we generate $\frac{m}{4}$ integers from $-\frac{m}{2}$ to $\frac{m-2}{2}$ uniformly at random (rounding $\frac{m}{4}$ to the nearest integer), which makes the errors easy to see in the coming figures, yet not too extreme. Recall that S is a set: each member i of S occurs only once irrespective of how many times the sampling procedure just described chooses to include the index i.

Figures 2–5 display results for retaining radial and horizontal lines, using the same original images (the third and tenth — "lower" and "upper" — slices out of twenty). Further examples are available in Appendix A. The figures whose captions specify "2×" use $\frac{2(m+n)}{5}$ random angles for radially retained sampling and $\frac{m}{2}$ random integers for horizontally retained sampling instead of the $\frac{m+n}{5}$ random angles and $\frac{m}{4}$ random integers used in all other figures. All figures concern MRI scans of a patient's head from the data of Loizou et al. (2013a), Loizou et al. (2011), Loizou et al. (2013b), and Loizou et al. (2015).[1] The resolutions in pixels of the original images range from 376 × 286 to 456 × 371. The acquisitions used turbo spin-echo pulse sequences for T2 weighting on a 1.5 tesla scanner; the repetition time was 4408 ms, the echo time was 100 ms, and the echo spacing was 10.8 ms. The thickness of a cross-sectional slice was 5 mm. The resolution was 2.226 pixels per mm. Discrete Fourier transforms of the downloaded full reconstructions yielded proxies for the measurements in the spatial Fourier domain ("k-space"). The scans did not accelerate the acquisitions via any form of parallel imaging. Ongoing work of others generalizes our approach to raw multi-coil data.

In the figures, "Original" displays the original image, "Reconstruction" displays the reconstruction $f(X, S)$, "Error of Reconstruction" displays the difference between the original image and the reconstruction, "Jackknife" displays the jackknife d from (1), and "Bootstrap" displays the bootstrap e from (3). The values of the original pixels are normalized to range from 0 to 1 (that is, the pixels in the original data for each cross-sectional slice get normalized to range from 0 to 1; the images corresponding to

[1] The Python software package **fbooja** includes a re-distribution of all data processed for the present paper, along with modular codes for automatically reproducing all our results, and is available at `https://github.com/facebookresearch/fbooja` … an extended data set with MRI scans from multiple patients (one of whom is the patient considered in the results reported here) is available at `http://www.medinfo.cs.ucy.ac.cy/old/doc/Publications/Datasets/MRIFreeDataset.zip`

the same cross-sectional slice do not get further normalized individually). In the images "Original" and "Reconstruction," pure black corresponds to 0 while pure white corresponds to 1. In the images "Error of Reconstruction," "Jackknife," and "Bootstrap," pure white and pure black correspond to the extreme values ±1, whereas 50% gray (halfway to black or to white) corresponds to 0. Thus, in the images displaying errors and potential errors, middling halftone grays correspond to little or no error, while extreme pure white and pure black correspond to more substantial errors.

For the visualizations from Subsection 2.2, we focus on two cross-sectional slices: the lower slice is the third of twenty in Appendix A, while the upper slice is the tenth. Figures 6–13 display the visualizations from Subsection 2.2. Figures 14 and 15 depict the effects of the blur from Subsection 2.3, implemented with `skimage.filters.gaussian` (a function for convolution with a Gaussian) from **scikit-image** of van der Walt et al. (2014). Table 1 reports how drastically such a nearly imperceptible blur changes the square roots of the sums of the squares of the error estimates. Background noise clearly overwhelms the root-mean-square without any denoising of the error estimates — the root-mean-square decreases dramatically even with just the mild denoising of blurring with a normalized Gaussian convolutional kernel whose standard deviation is one pixel, as in Table 1 and Figures 14 and 15. Tables 2 and 3 report how blurring with wider Gaussians affects the root-mean-square; of course, wider Gaussian blurs are much more conspicuous and risk washing out important coherent features of the error estimates, while the last column of Table 3 shows that denoising with wider Gaussian blurs brings diminishing returns. The width used in Table 1 and Figures 14 and 15 — only one pixel — may be safest. Appendix B displays the grayscale reconstructions overlaid with the blurred bootstraps (blurring with a Gaussian whose standard deviation is one pixel), thresholded and colorized as in Subsection 2.2.

4. Discussion and Conclusion

The jackknife images are generally noisier than the bootstrap images. The bootstrap directly explores parts of the Fourier domain outside the observed measurements, whereas the jackknife is more like a convergence test or a differential approximation to the bootstrap — see, for example, the review of Efron and Tibshirani (1993). Both the jackknife and the bootstrap occasionally display artifacts where in fact the reconstruction was accurate. Moreover, they miss some anomalies; if the reconstruction completely washes out a feature of the original image, then neither the jackknife nor the bootstrap can detect the washed-out feature (consider, for instance, the stark long straight line in the upper-right quadrants of the errors in reconstruction from Figures 2 and 3 — the jackknife and bootstrap fail to detect that line). Neither the jackknife nor the bootstrap can hallucinate or otherwise introduce a feature in a reconstruction that the reconstruction procedure is not designed to handle; in particular, any machine-learned model is practically guaranteed to miss any feature not represented in the data set used for training the model. That said, in most cases they show the actual errors nicely. The estimates bear an uncanny qualitative resemblance to the actual errors. Using both the jackknife and the bootstrap may be somewhat conservative, but if the jackknife misses an error, then the bootstrap usually catches it, and vice versa.

Similarly, most clinical practice involves comparison of MRI scans of the same patient across proton density, T1, and T2 weighting, and sometimes also with other weightings (FLAIR or fat-suppression, for example) and other physical quantities such as susceptibility, diffusion, or densities of radiological contrast agents. Comparing across these different sequence types (and possibly different imaging modalities) affords an additional opportunity to check for inconsistencies that may indicate errors in the acquisition or reconstruction, including errors arising from compressed sensing. An anonymous reviewer also suggested that careful study of the errors introduced by particular schemes for reconstruction (schemes such as the minimization of total variation tested in Section 3) could inform Bayesian priors useful for interpreting individual reconstructions in the clinical setting. The reviewer noted that the most prominent anomalies in the examples of Section 3 appear to focus on the boundary interfaces of different tissues, on artifacts due to motion of the cerebro-spinal fluid, and on idiosyncrasies in the geometry of the sampling scheme (radial versus horizontal). Clinicians will likely need to consider the reviewer's observations and perhaps develop suitable Bayesian priors.

Broadly speaking, the bootstrap-saturated reconstructions and bootstrap-interpolated reconstructions look similar, even though the details of their constructions differ. Both the bootstrap-saturated reconstruction and the bootstrap-interpolated reconstruction highlight errors more starkly on pixels for which the reconstruction is bright; dark green, dark red, and dark magenta (that is, with a relatively low value in hue-saturation-value) simply do not jump out visually, even if the green, red, or magenta are fully saturated. That said, retaining the value of the pixel in the reconstruction makes the colorization of the bootstrap-saturated reconstruction and the bootstrap-interpolated reconstruction far less distracting than in errors over a threshold overlaid, with much higher fidelity to the form of the grayscale reconstruction in the colored regions. Of course, the errors over a threshold overlaid do not alter the grayscale reconstruction at all when the errors are within the threshold, so the fidelity to the grayscale reconstruction is perfect in those areas of the images with overlaid errors where the error estimates do not go beyond the threshold.

Thus, none of the colorizations is uniformly superior to the others, and all may be too distracting for actual clinical practice. Alternatives include direct display of the bootstrap error estimates, possibly complemented by the bootstrap subtracted from the reconstruction (to illustrate the effects of "correcting" the reconstruction with the error estimates), which are readily interpretable and minimally distracting.

The bootstrap subtracted from the reconstruction tends to sharpen the reconstruction and to add back some features such as lines or textures that the reconstruction obscured. However, this reconstruction that is "corrected" with the bootstrap estimations may contain artifacts not present in the original image — the error estimates tend to be conservative, possibly suspecting errors in some regions where in fact the reconstruction is accurate. The "corrected" reconstruction (that is, the bootstrap subtracted from the reconstruction) can be illuminating, but only as a complement to plotting the bootstrap error estimates on their own, too.

A sensible protocol could be to check if the root-mean-square of the blurred bootstrap is large enough to merit further investigation, investigating further by looking at the full bootstrap image together with the reconstruction "corrected" by subtracting off the bootstrap error estimates (or colorizations). Consulting the jackknife is another (albeit noisier) possibility.

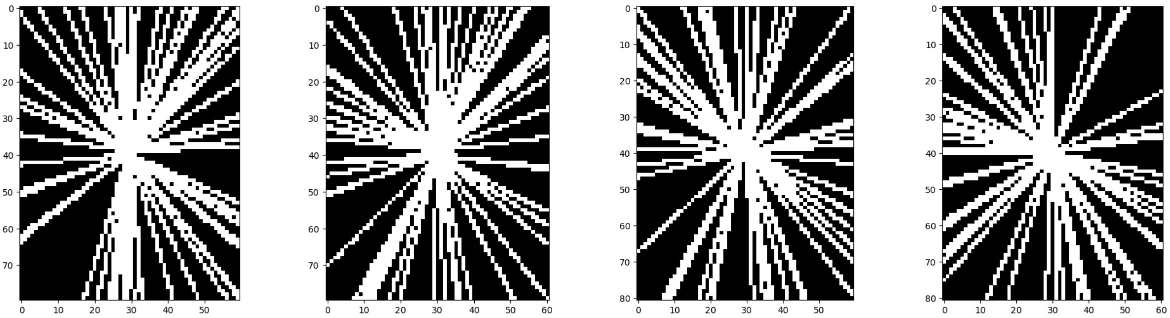

Figure 1: Radially retained sampling — sampling on a Cartesian grid along rays emanating from the origin

Table 1: Square roots of the sums of the squares of the error estimates

Sampling	Slice	Bootstrap	Blurred Bootstrap
horizontally	lower	12.9	6.25
horizontally	upper	13.8	7.34
radially	lower	17.5	10.5
radially	upper	18.0	11.6

For obvious reasons, the clinical setting needs to be rather conservative in adopting novel techniques such as those proposed in the present paper. Medical research, however, might be able to more unreservedly leverage the proposed methods to screen for problems or otherwise quantify uncertainty in advanced procedures for image acquisition, reconstruction, and analysis. The jackknife and the bootstrap can estimate errors in any scheme for reconstruction from undersampled measurements, without requiring any modification of the software for reconstruction — a convenience that could favor accelerated adoption as a handy tool in research, whether as inputs to a pipeline of data analysis or merely for flagging egregious errors.

Computational details

The results in this paper were obtained using the Python package **fbooja** available at `https://github.com/facebookresearch/fbooja` ... **fbooja** builds upon PyTorch of Paszke et al. (2019), which is available at `https://pytorch.org` along with instructions for installation of the dependencies **Matplotlib** of Hunter (2007), **NumPy** of Harris et al. (2020), **Pillow** of Clark (2021), **Scikit-image** of van der Walt et al. (2014), and **SciPy** of Virtanen et al. (2020).

Acknowledgements

We would like to thank Florian Knoll, Jerry Ma, Jitendra Malik, Matt Muckley, Mike Rabbat, Dan Sodickson, Larry Zitnick, and the anonymous referees and editor.

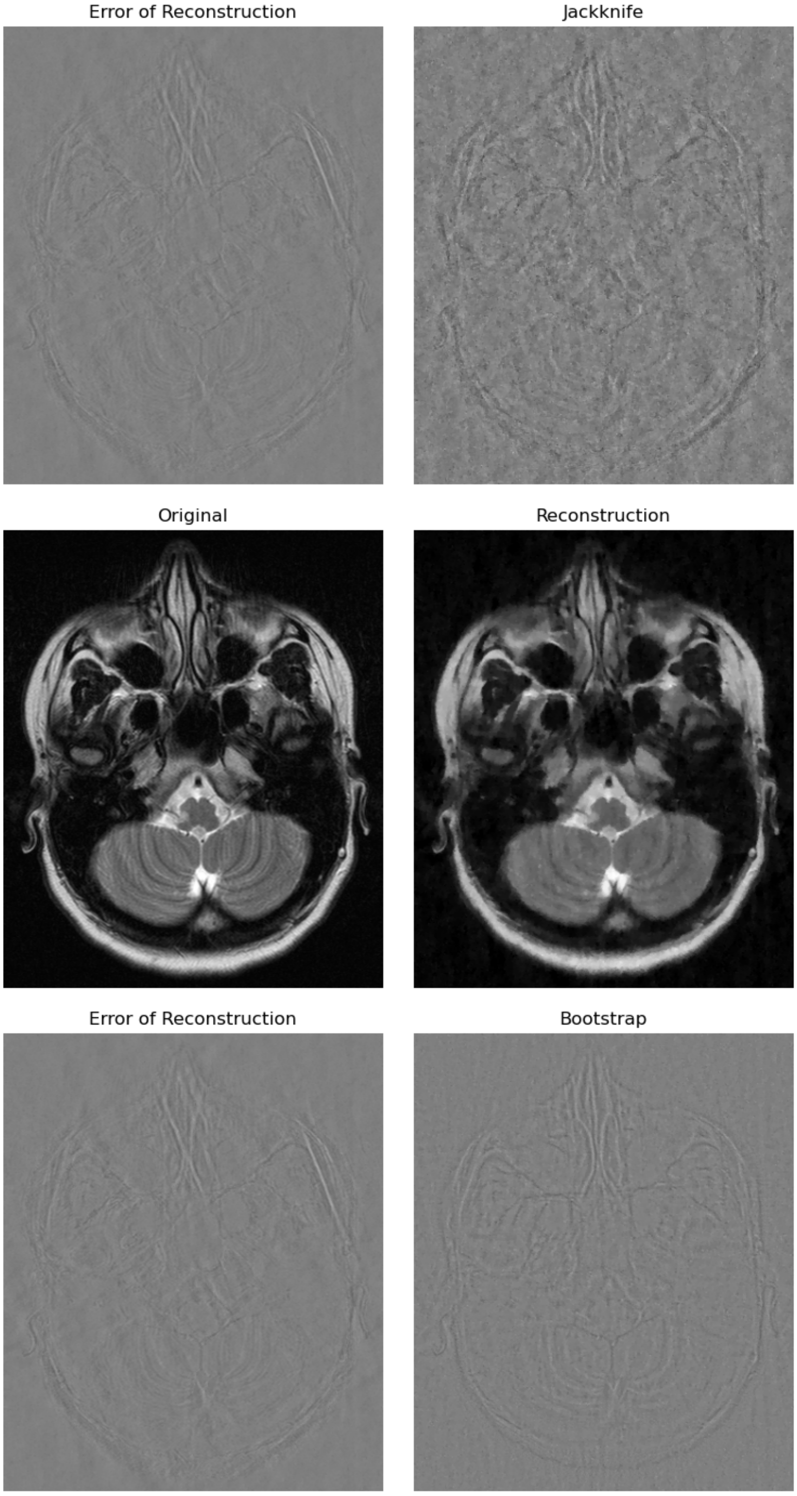

Figure 2: Radially retained sampling — lower slice

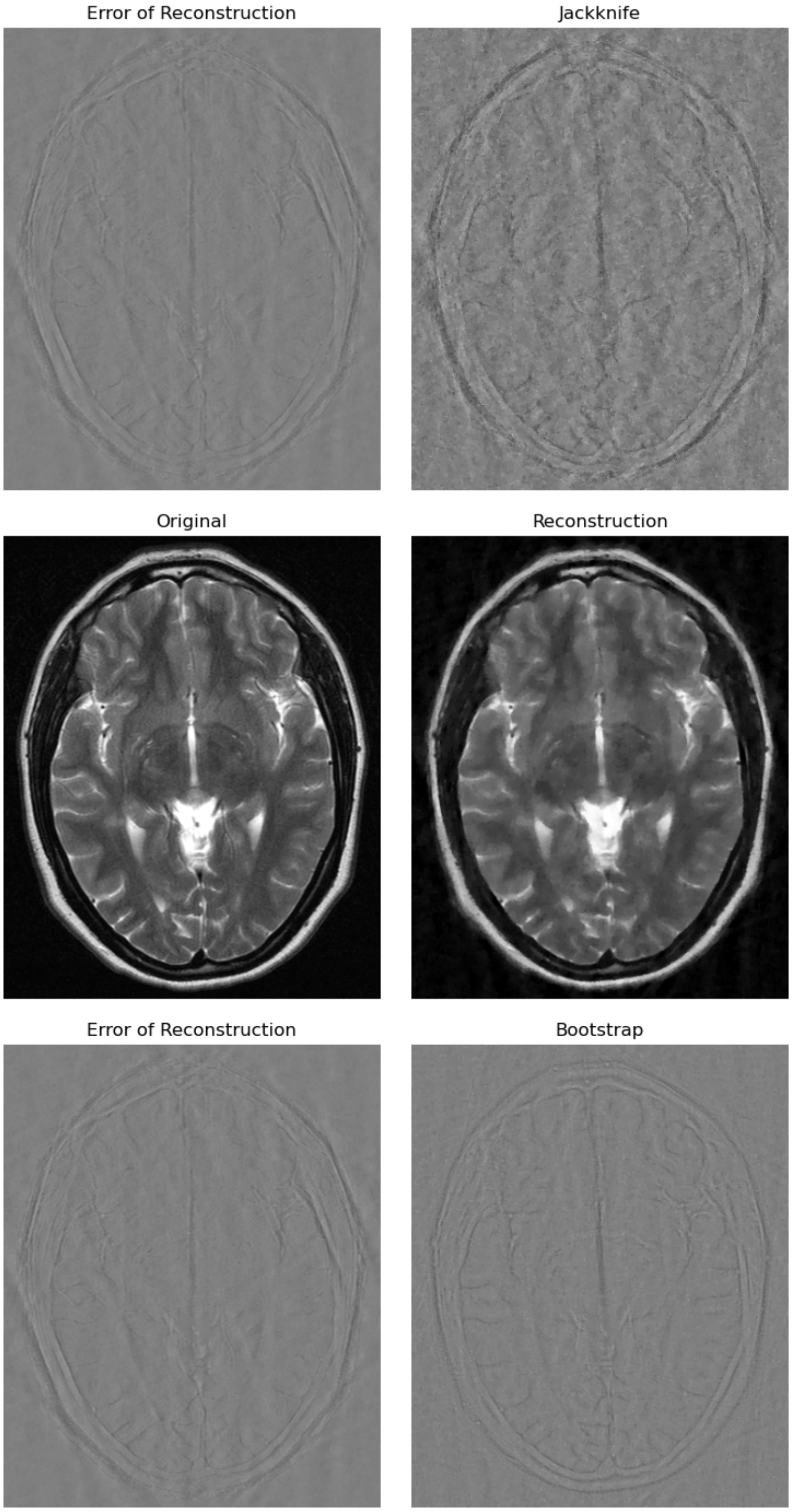

Figure 3: Radially retained sampling — upper slice

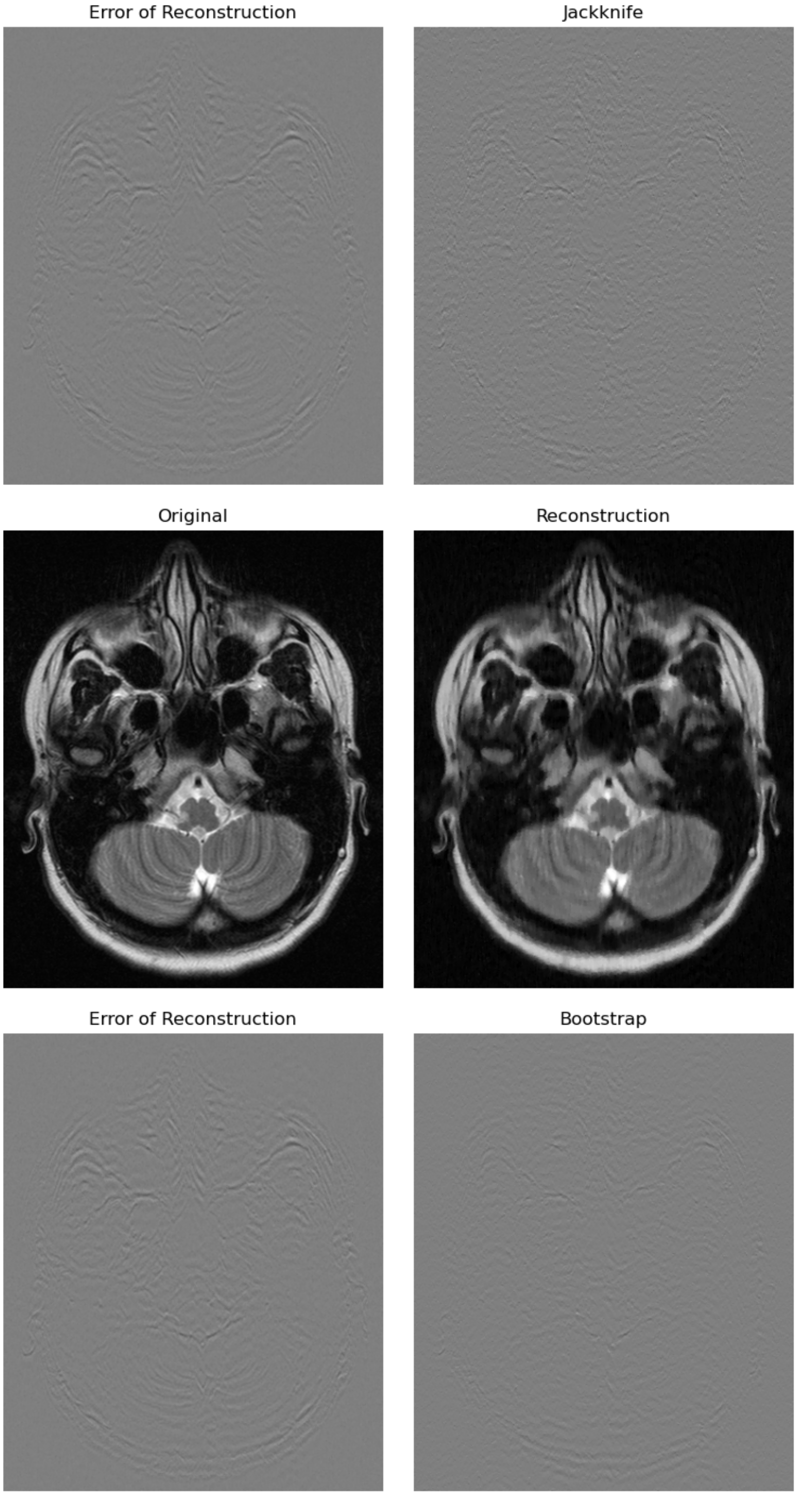

Figure 4: Horizontally retained sampling — lower slice

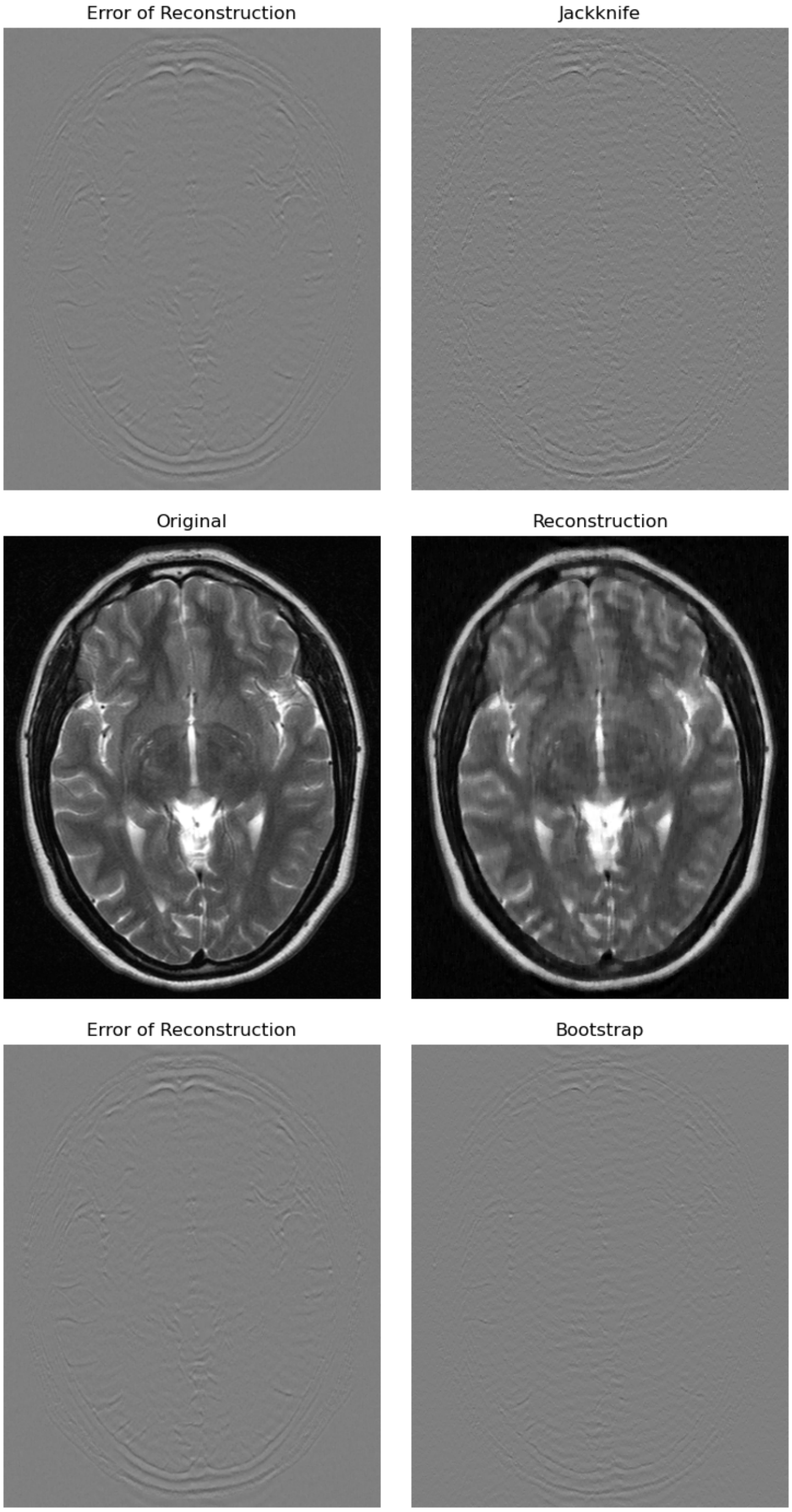

Figure 5: Horizontally retained sampling — upper slice

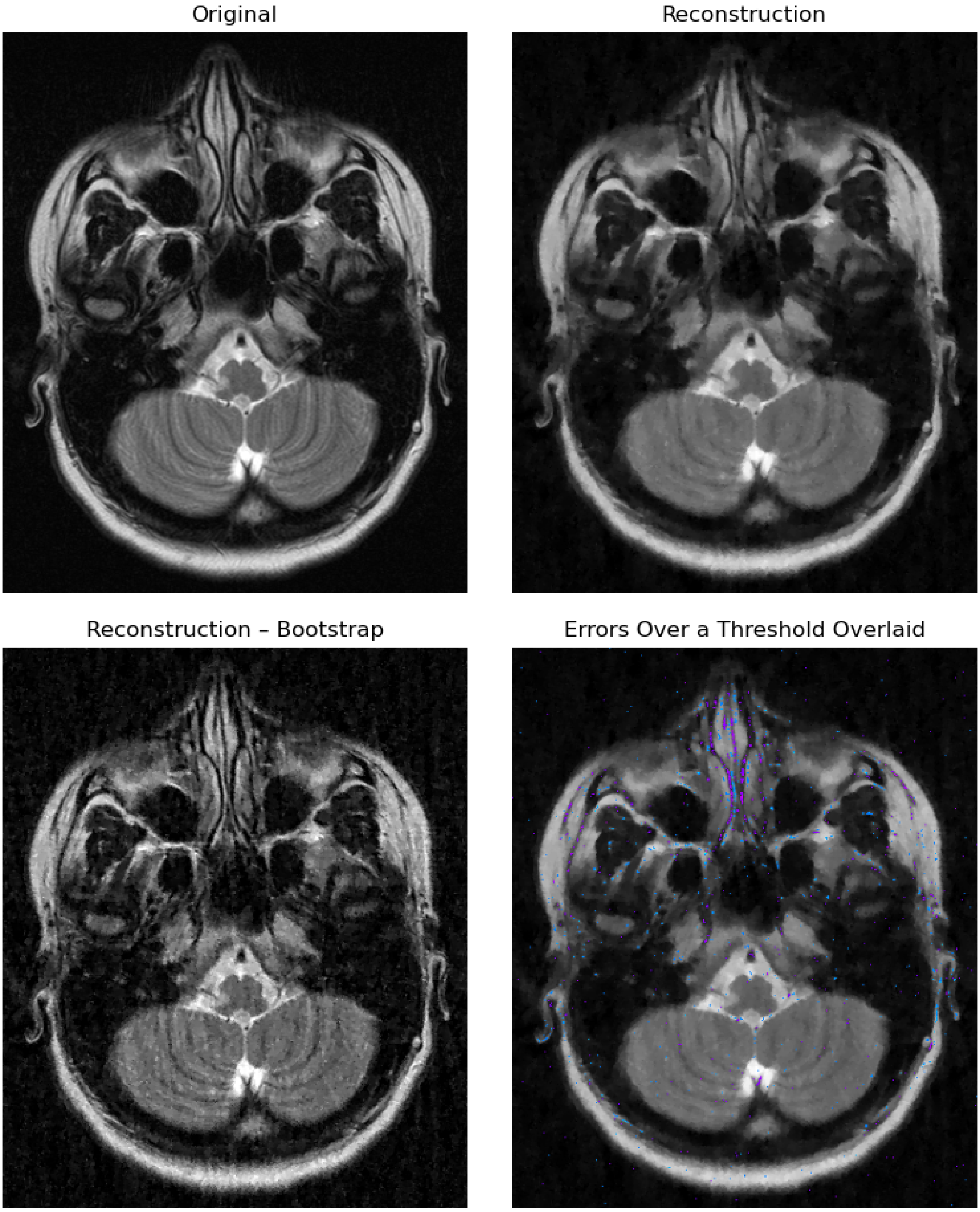

Figure 6: Radially retained sampling — lower slice (a)

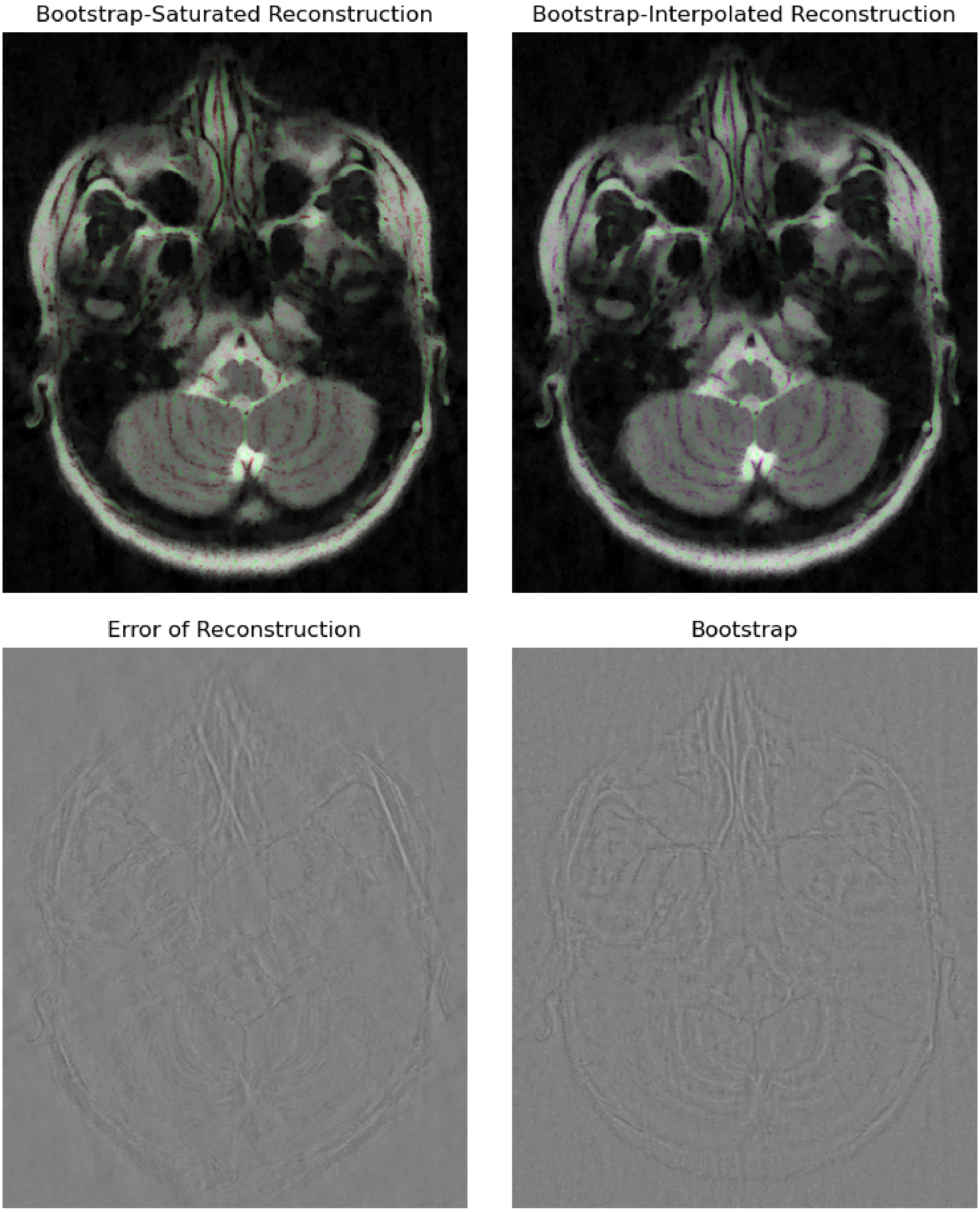
Bootstrap-Saturated Reconstruction
Bootstrap-Interpolated Reconstruction
Error of Reconstruction
Bootstrap

Figure 7: Radially retained sampling — lower slice (b)

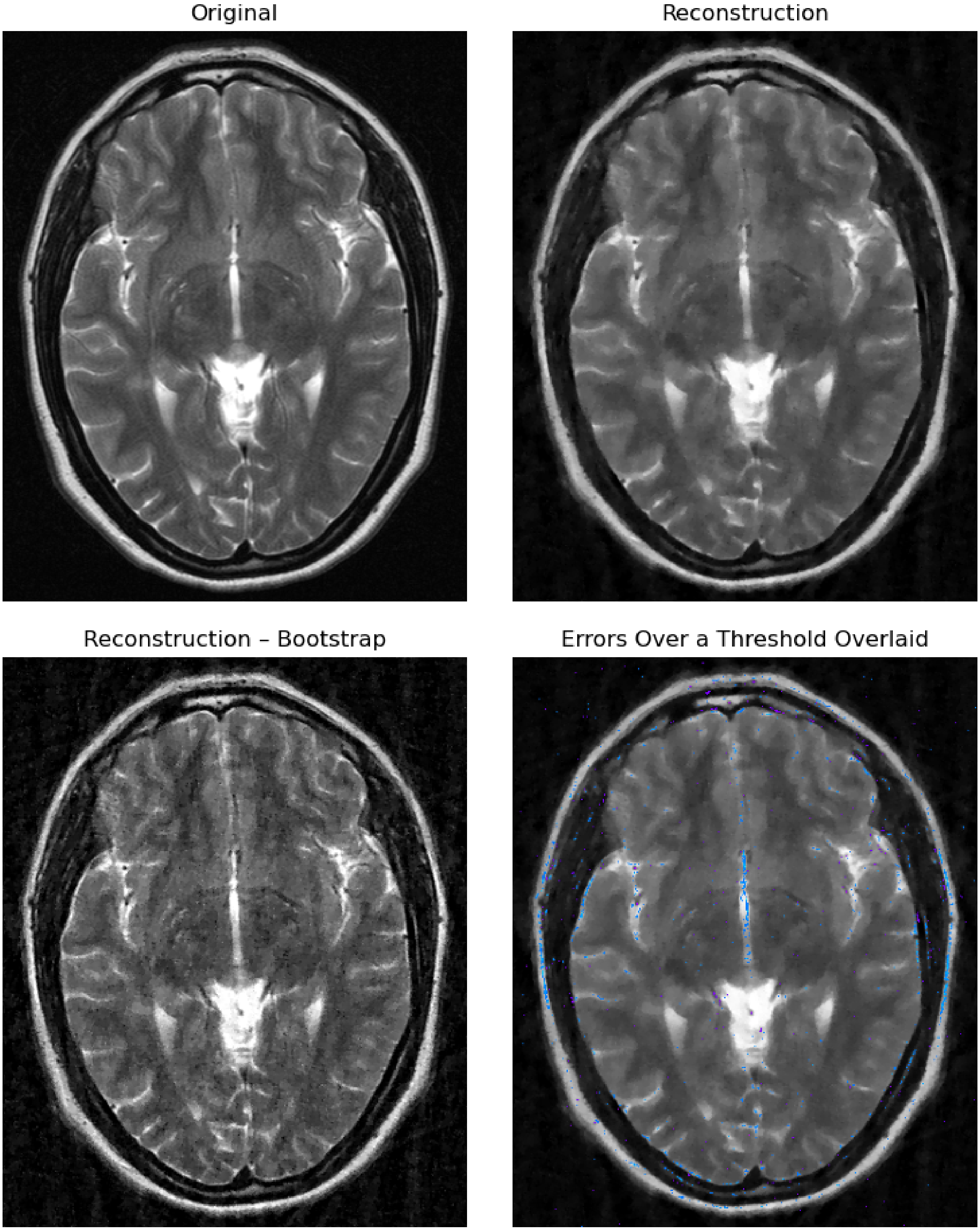

Figure 8: Radially retained sampling — upper slice (a)

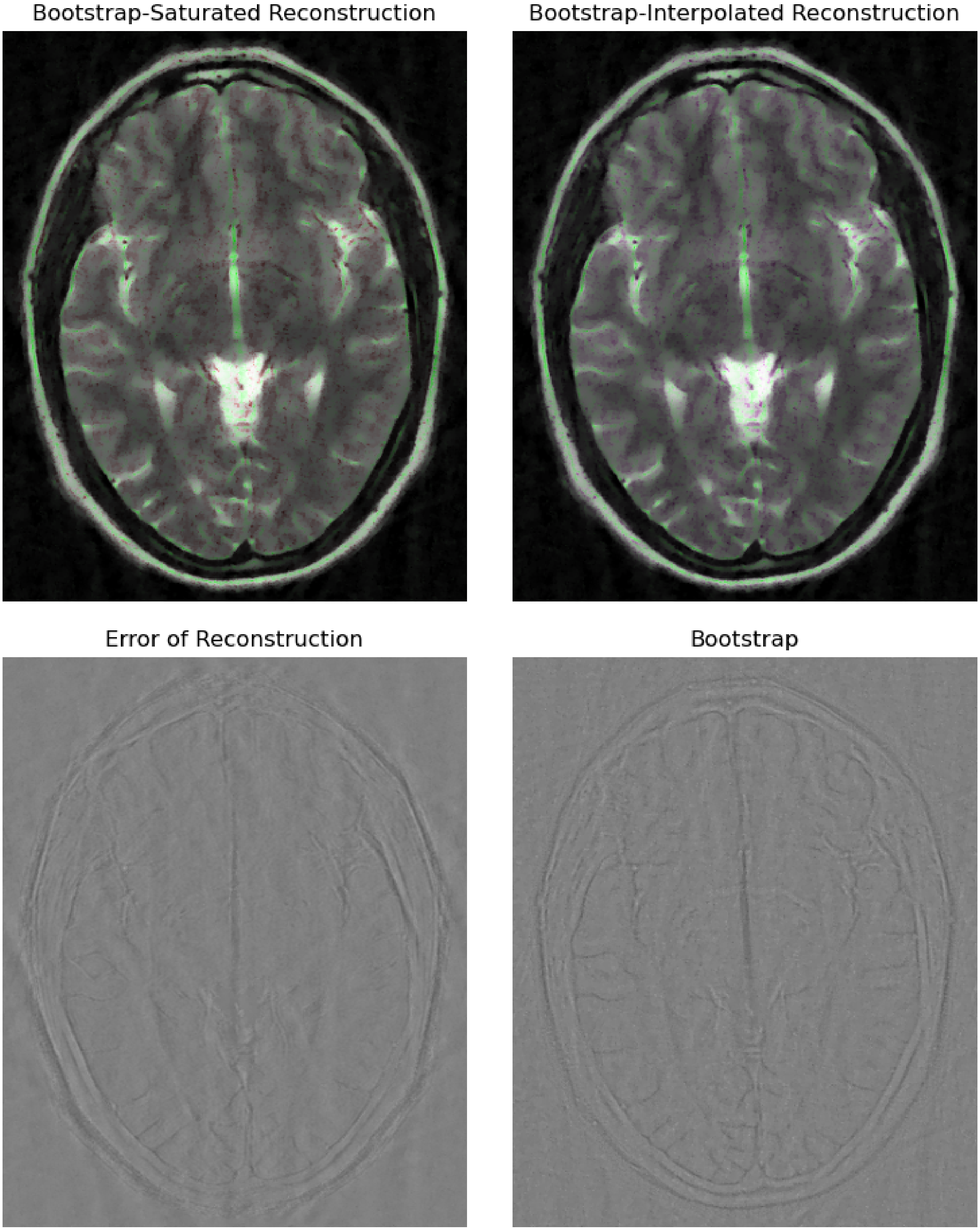
Bootstrap-Saturated Reconstruction
Bootstrap-Interpolated Reconstruction
Error of Reconstruction
Bootstrap

Figure 9: Radially retained sampling — upper slice (b)

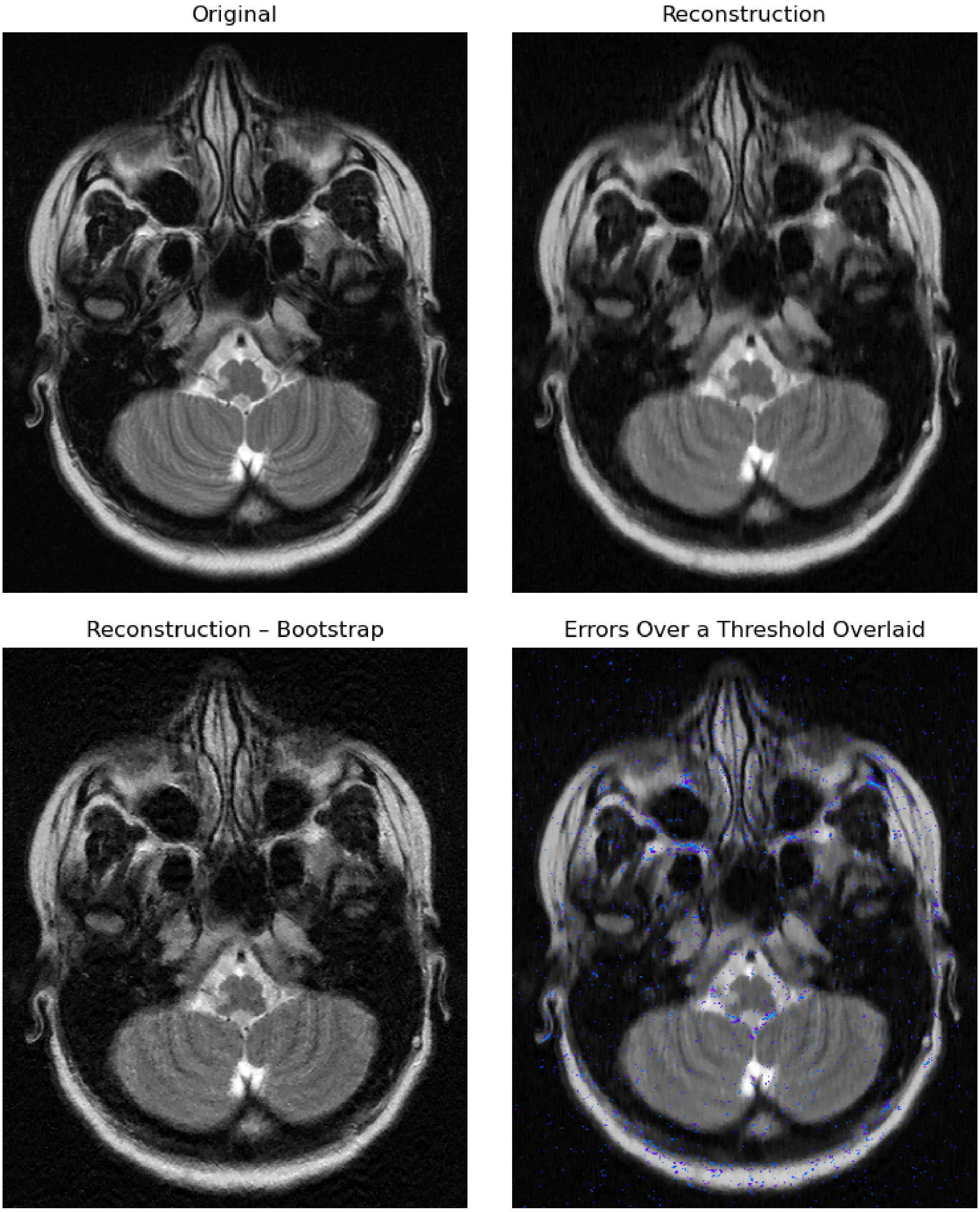

Figure 10: Horizontally retained sampling — lower slice (a)

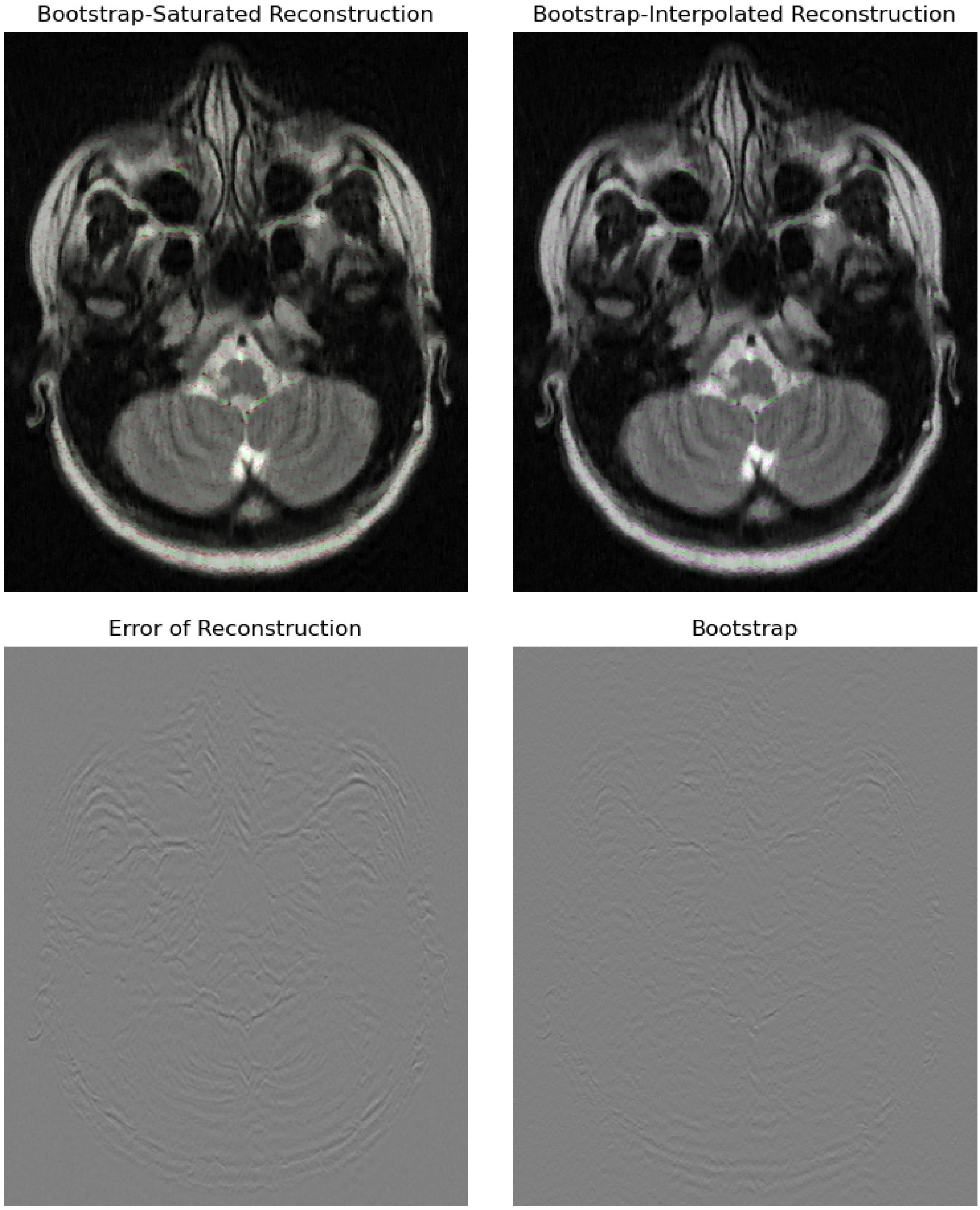
Bootstrap-Saturated Reconstruction
Bootstrap-Interpolated Reconstruction
Error of Reconstruction
Bootstrap

Figure 11: Horizontally retained sampling — lower slice (b)

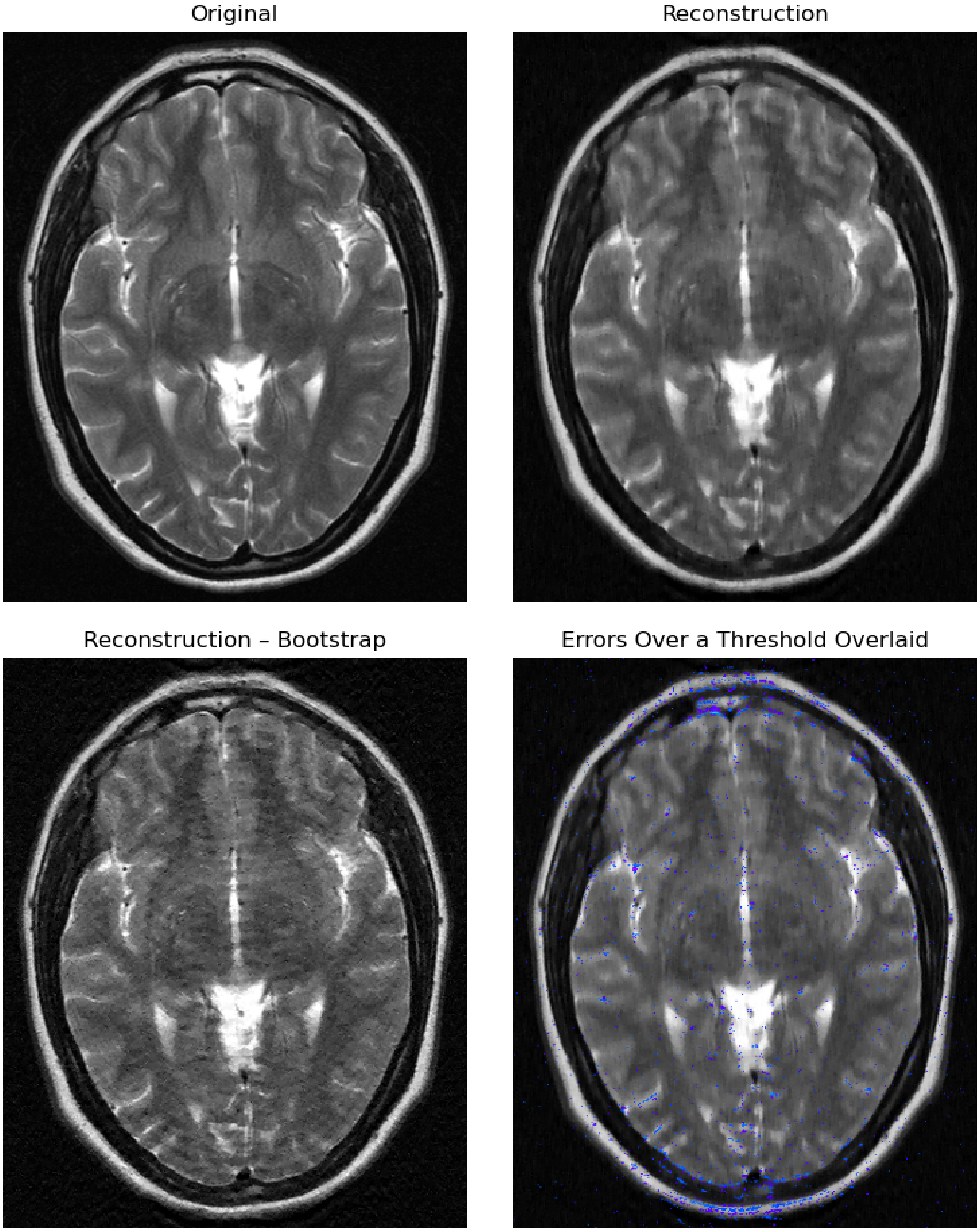

Figure 12: Horizontally retained sampling — upper slice (a)

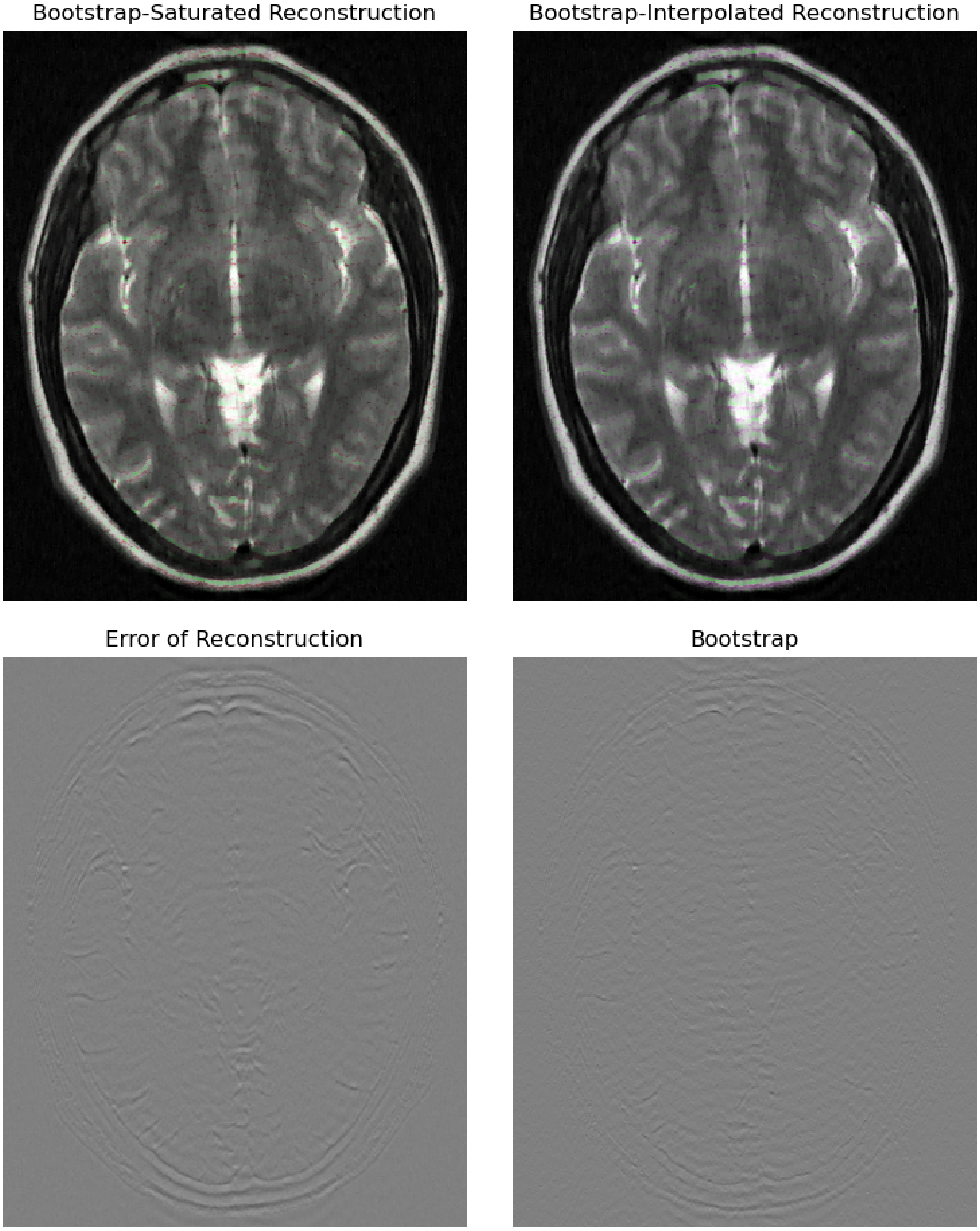
Bootstrap-Saturated Reconstruction
Bootstrap-Interpolated Reconstruction
Error of Reconstruction
Bootstrap

Figure 13: Horizontally retained sampling — upper slice (b)

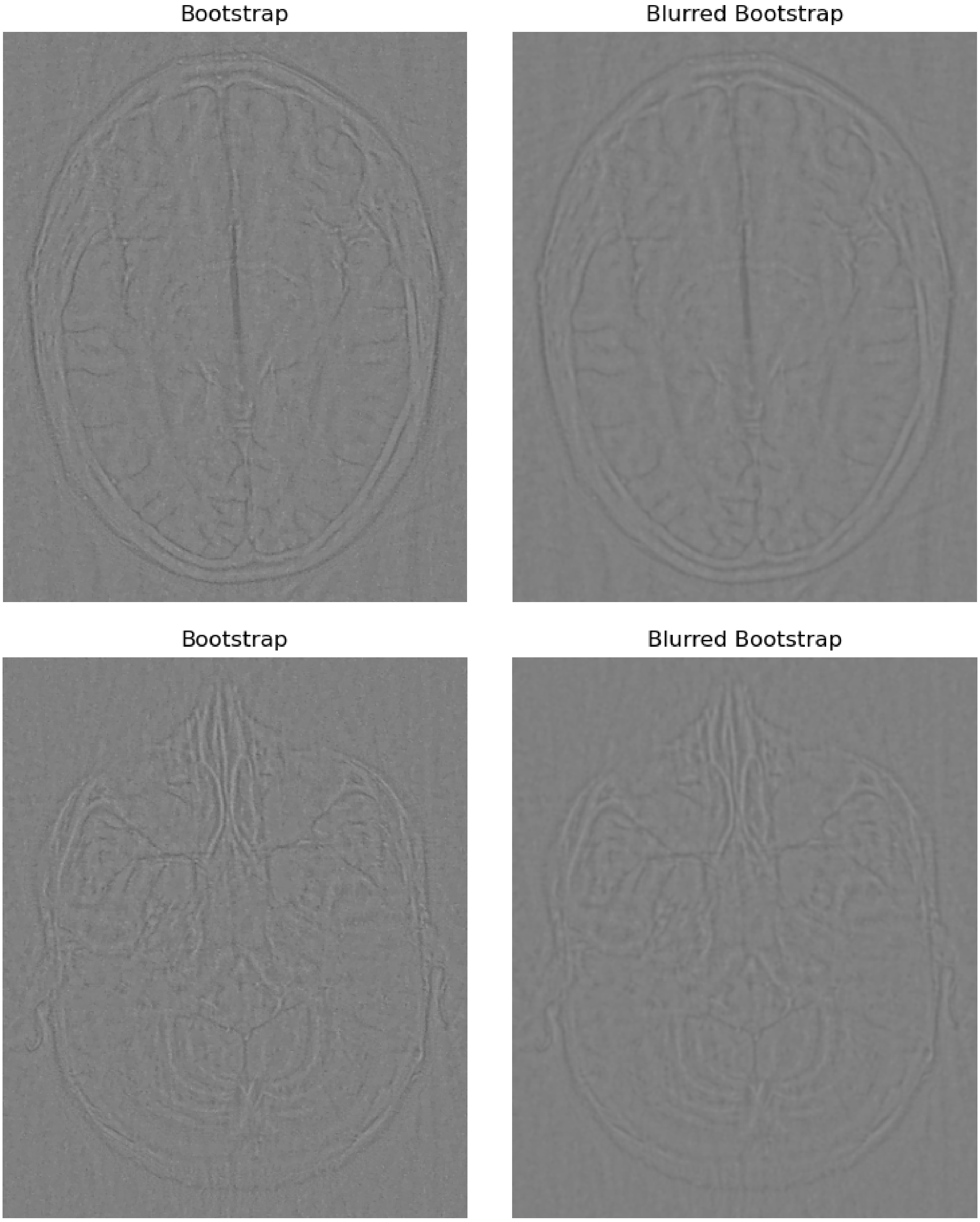

Figure 14: Radially retained sampling — upper plots display the upper slice; lower plots display the lower

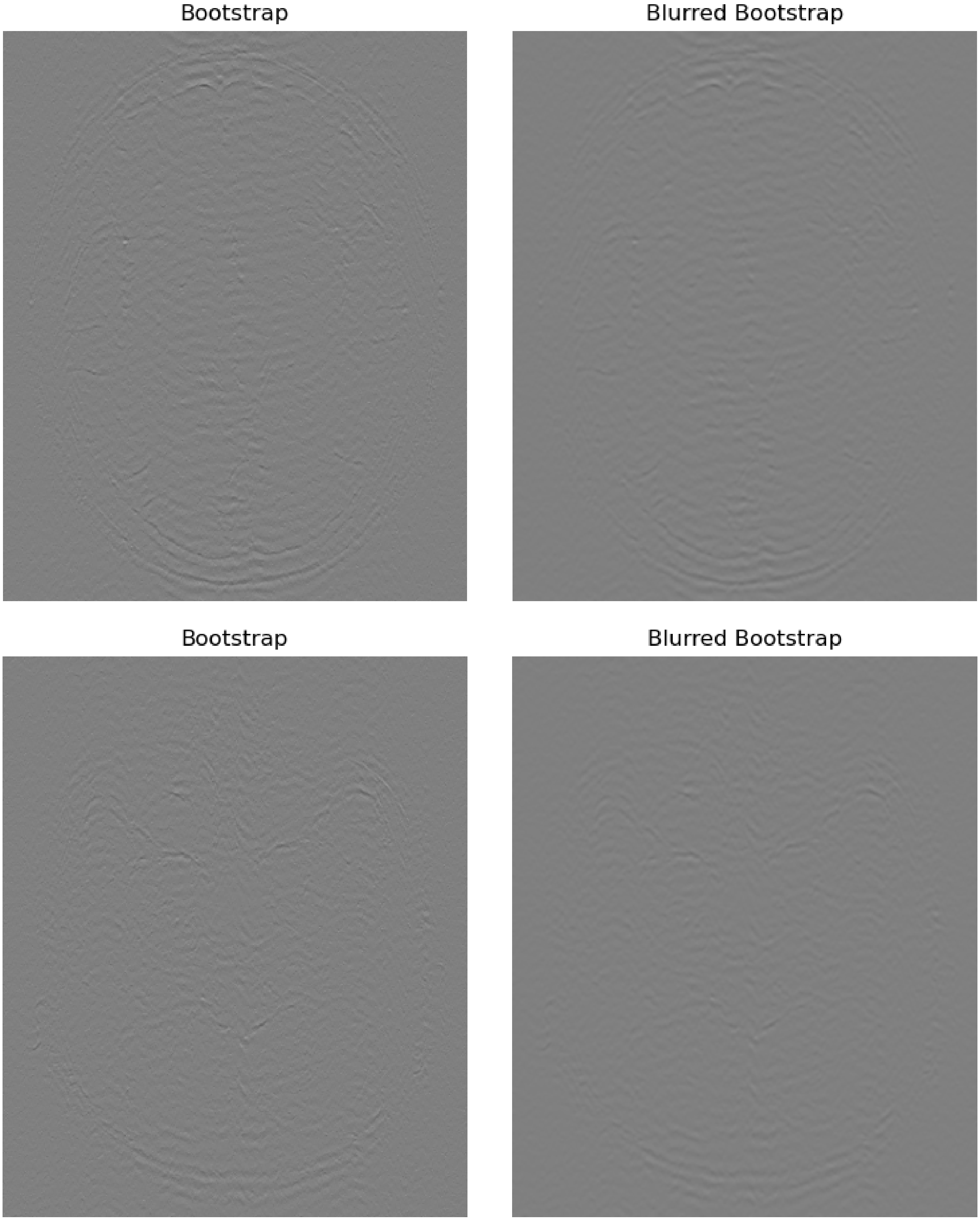

Figure 15: Horizontally retained sampling — upper plots display the upper slice; lower plots display the lower

Table 2: Square roots of the sums of the squares of the error estimates for the lower slice blurred against a Gaussian convolutional kernel of the specified standard deviation (the standard deviation is in pixels), for sampling retained horizontally or radially

Std. Dev.	Horizontally	Radially
0.0	12.9	17.5
0.5	9.94	14.6
1.0	6.25	10.5
1.5	4.38	8.06
2.0	3.03	6.34
2.5	2.04	5.06
3.0	1.33	4.09
3.5	.847	3.34
4.0	.535	2.75

Table 3: Square roots of the sums of the squares of the error estimates for the upper slice blurred against a Gaussian convolutional kernel of the specified standard deviation (the standard deviation is in pixels), for sampling retained horizontally or radially

Std. Dev.	Horizontally	Radially
0.0	13.8	18.0
0.5	10.9	15.3
1.0	7.34	11.6
1.5	5.35	9.50
2.0	3.82	7.97
2.5	2.63	6.79
3.0	1.75	5.87
3.5	1.14	5.13
4.0	.745	4.54

References

Brown, R. W., Cheng, Y.-C. N., Haacke, E. M., Thompson, M. R., and Venkatesan, R. (2014). *Magnetic Resonance Imaging: Physical Principles and Sequence Design*. Wiley-Blackwell, 2nd edition, DOI: 10.1002/9781118633953.

Candes, E. J., Romberg, J., and Tao, T. (2006). Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, DOI: 10.1109/TIT.2005.862083.

Clark, A. (2021). **Pillow** (**PIL** fork) documentation. https://readthedocs.org/projects/pillow/downloads/pdf/latest.

Efron, B. and Tibshirani, R. J. (1993). *An Introduction to the Bootstrap*. CRC Monographs on Statistics & Applied Probability. Chapman & Hall, New York, NY, DOI: 10.1201/9780429246593.

Hammernik, K., Klatzer, T., Kobler, E., Recht, M. P., Sodickson, D. K., and Knoll, F. (2018). Learning a variational network for reconstruction of accelerated MRI data. *Magnetic Resonance in Medicine*, 79(6):3055–3071, DOI: 10.1002/mrm.26977.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del R'ıo, J. F., Wiebe, M., Peterson, P., G'erard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. (2020). Array programming with **NumPy**. *Nature*, 585(7825):357–362, DOI: 10.1038/s41586-020-2649-2.

Hunter, J. D. (2007). **Matplotlib**: a 2D graphics environment. *Computing in Science & Engineering*, 9(3):90–95, DOI: 10.1109/MCSE.2007.55.

Loizou, C. P., Kyriacou, E. C., Seimenis, I., Pantziaris, M., Petroudi, S., Karaolis, M., and Pattichis, C. (2013a). Brain white matter lesion classification in multiple sclerosis subjects for the prognosis of future disability. *Intelligent Decision Technologies Journal*, 7:3–10, DOI: 10.1007/978-3-642-23960-1_47.

Loizou, C. P., Murray, V., Pattichis, M., Seimenis, I., Pantziaris, M., and Pattichis, C. (2011). Multi-scale amplitude-modulation–frequency-modulation (AM-FM) texture analysis of multiple sclerosis in brain MRI images. *IEEE Transactions on Information Technology in Biomedicine*, 15(1):119–129, DOI: 10.1109/TITB.2010.2091279.

Loizou, C. P., Pantziaris, M., Pattichis, C., and Seimenis, I. (2013b). Brain MRI image normalization in texture analysis of multiple sclerosis. *Journal of Biomedical Graphics and Computing*, 3(1):20–34, DOI: https://doi.org/10.5430/jbgc.v3n1p20.

Loizou, C. P., Petroudi, S., Seimenis, I., Pantziaris, M., and Pattichis, C. (2015). Quantitative texture analysis of brain white matter lesions derived from T2-weighted MR images in MS patients with clinically isolated syndrome. *Journal of Neuroradiology*, 42(2):99–114, DOI: 10.1016/j.neurad.2014.05.006.

Lustig, M., Donoho, D., and Pauly, J. M. (2007). Sparse MRI: the application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine*, 58(6):1182–1195, DOI: 10.1002/mrm.21391.

Malioutov, D. M., Sanghavi, S. R., and Willsky, A. S. (2010). Sequential compressed sensing. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):435–444, DOI: 10.1109/JSTSP.2009.2038211.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). PyTorch: an imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, pages 8024–8035. Curran Associates, https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

Tao, M. and Yang, J. (2009). Alternating direction algorithms for total variation deconvolution in image reconstruction. Technical Report TR0918, Department of Mathematics, Nanjing University. Available at http://www.optimization-online.org/DB_HTML/2009/11/2463.html.

Tropp, J. A. (2017). Book review: A mathematical introduction to compressive sampling by S. Foucart and H. Rauhut. *Bulletin of the American Mathematical Society*, 54(1):151–165, DOI: 10.1090/bull/1546.

van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., and Yu, T. (2014). **Scikit-image**: image processing in Python. *PeerJ*, 2(e453):1–18, DOI: 10.7717/peerj.453.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors (2020). **SciPy** 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17:261–272, DOI: 10.1038/s41592-019-0686-2.

Ward, R. (2009). Compressed sensing with cross-validation. *IEEE Transactions on Information Theory*, 55(12):5773–5782, DOI: 10.1109/TIT.2009.2032712.

Yang, J. and Zhang, Y. (2011). Alternating direction algorithms for ℓ_1-problems in compressive sensing. *SIAM Journal on Scientific Computing*, 33(1):250–278, DOI: 10.1137/090777761.

A. Supplementary jackknives and bootstraps

This appendix supplements Figures 2–5 of Section 3 with analogous figures for twenty cross-sectional slices through the head of the patient (Figures 2–5 are for slices 3 and 10).

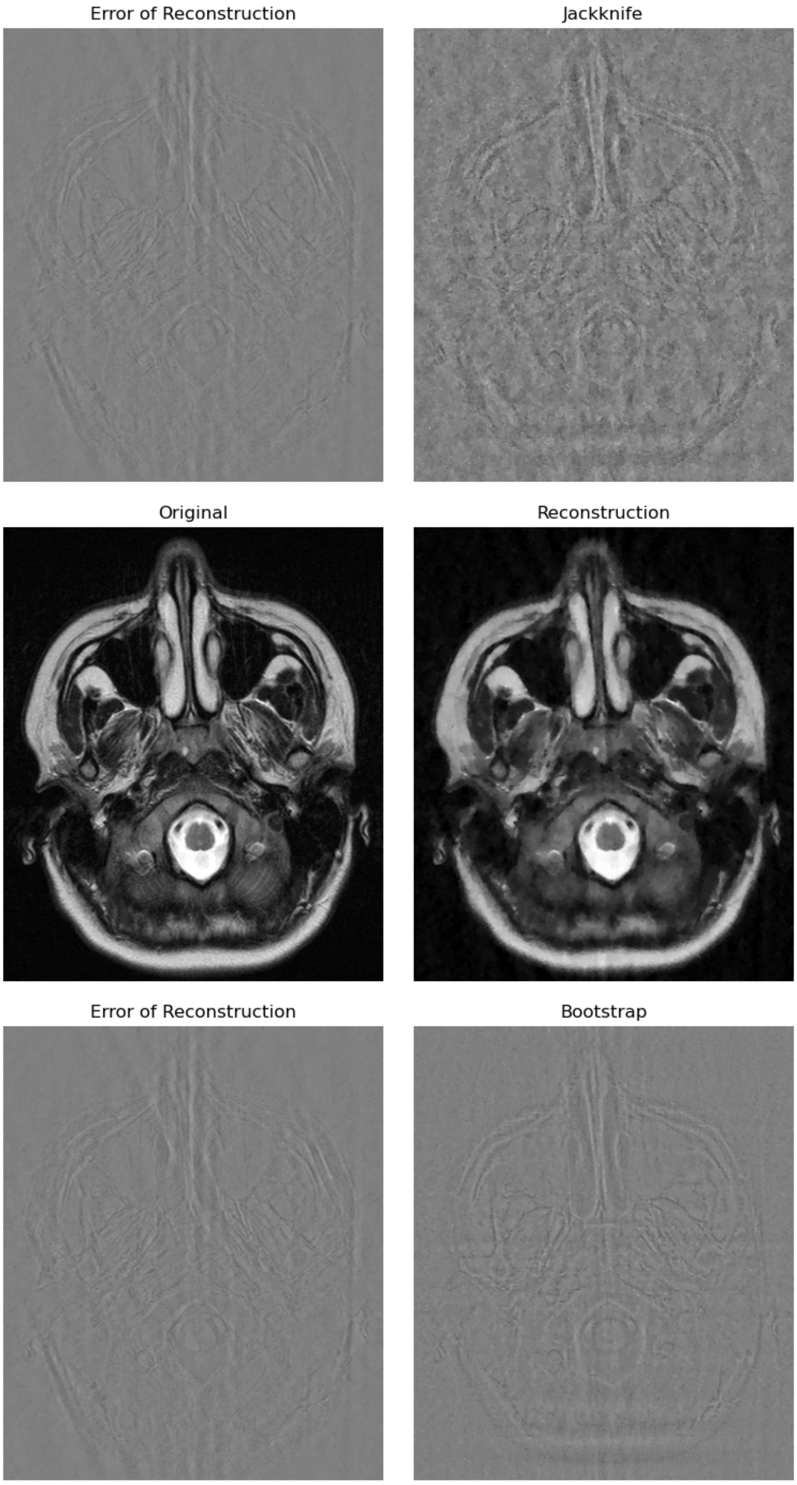

Figure 16: Radially retained sampling — slice 1

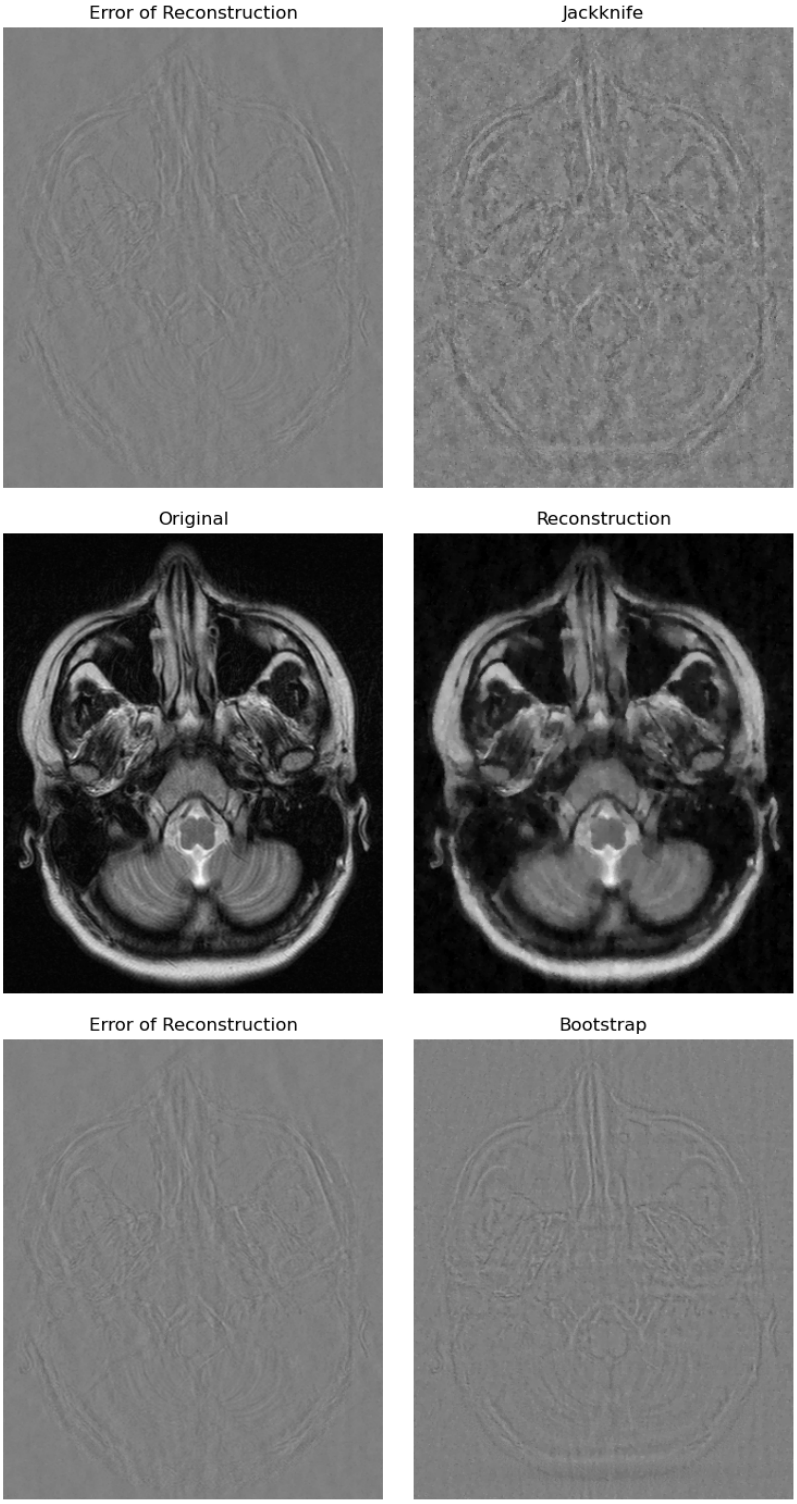

Figure 17: Radially retained sampling — slice 2

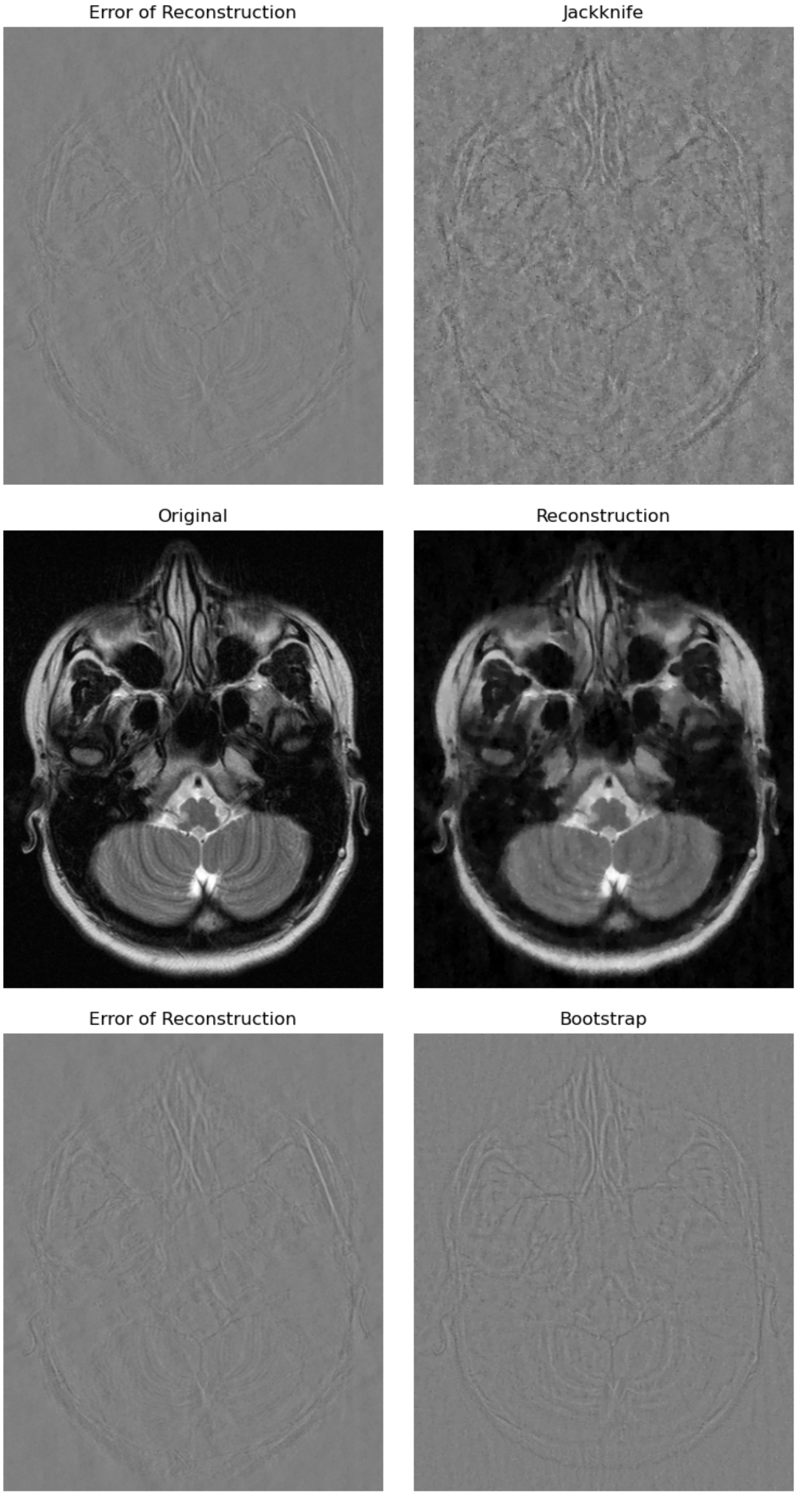

Figure 18: Radially retained sampling — slice 3

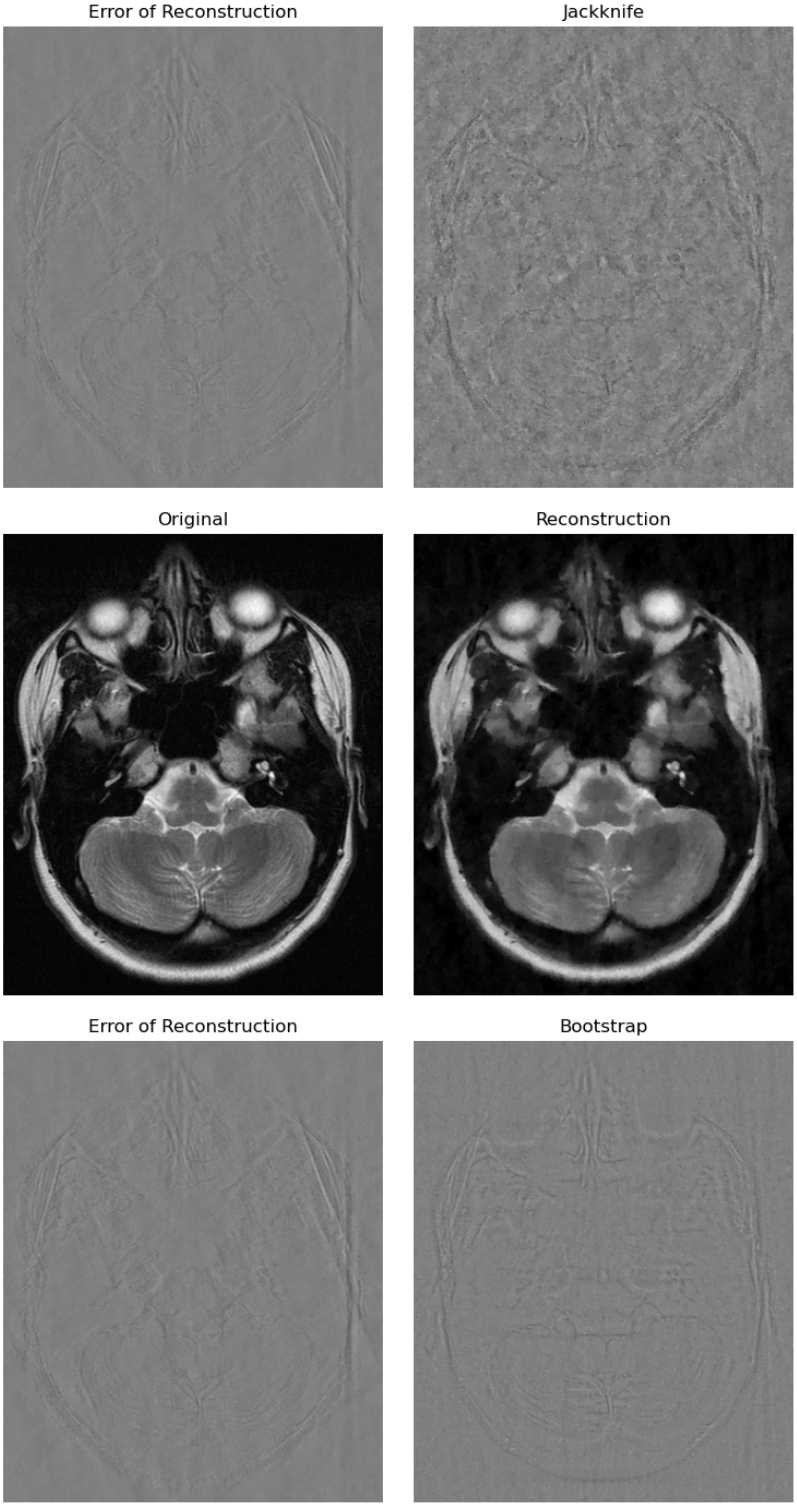

Figure 19: Radially retained sampling — slice 4

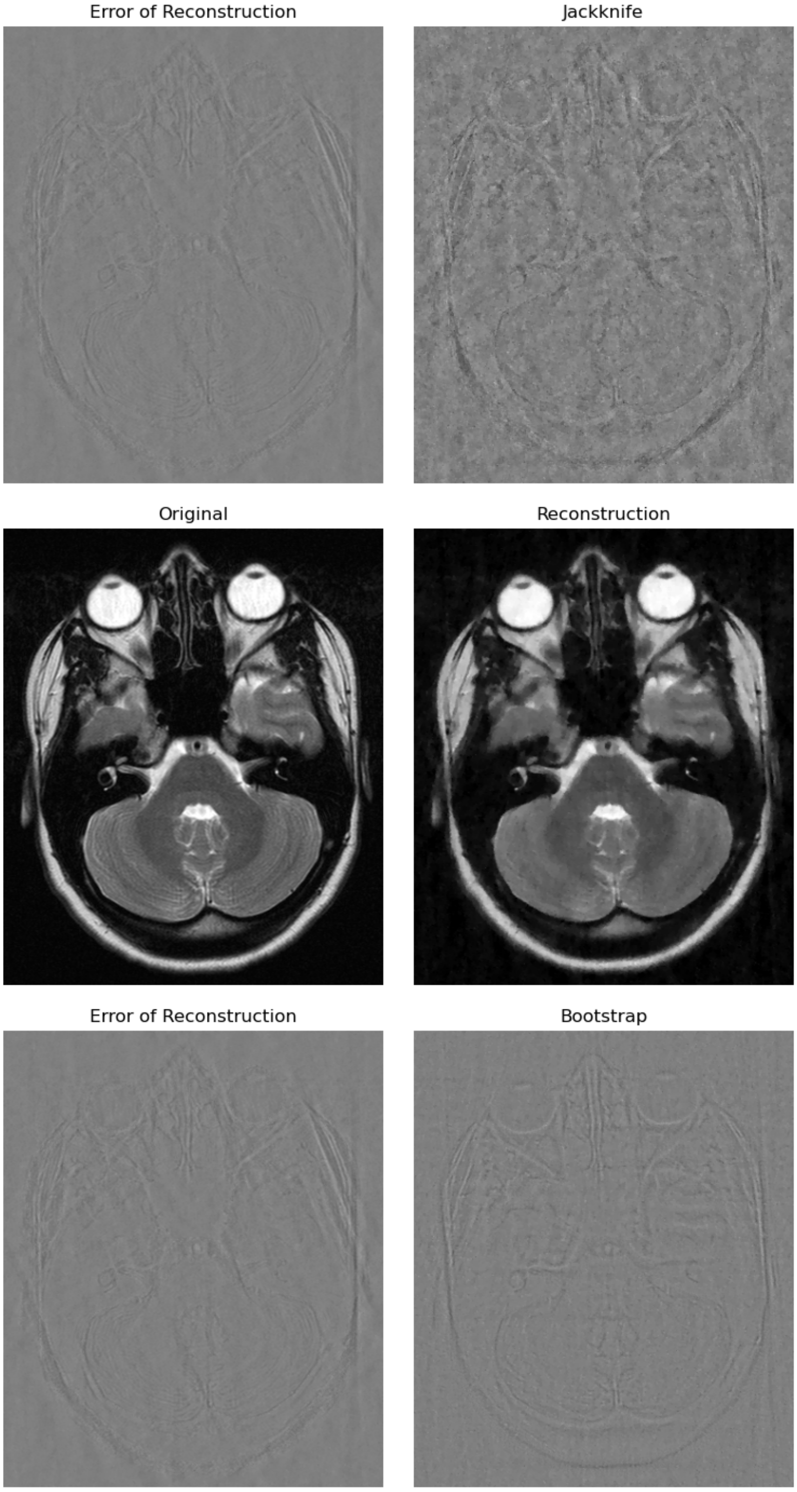

Figure 20: Radially retained sampling — slice 5

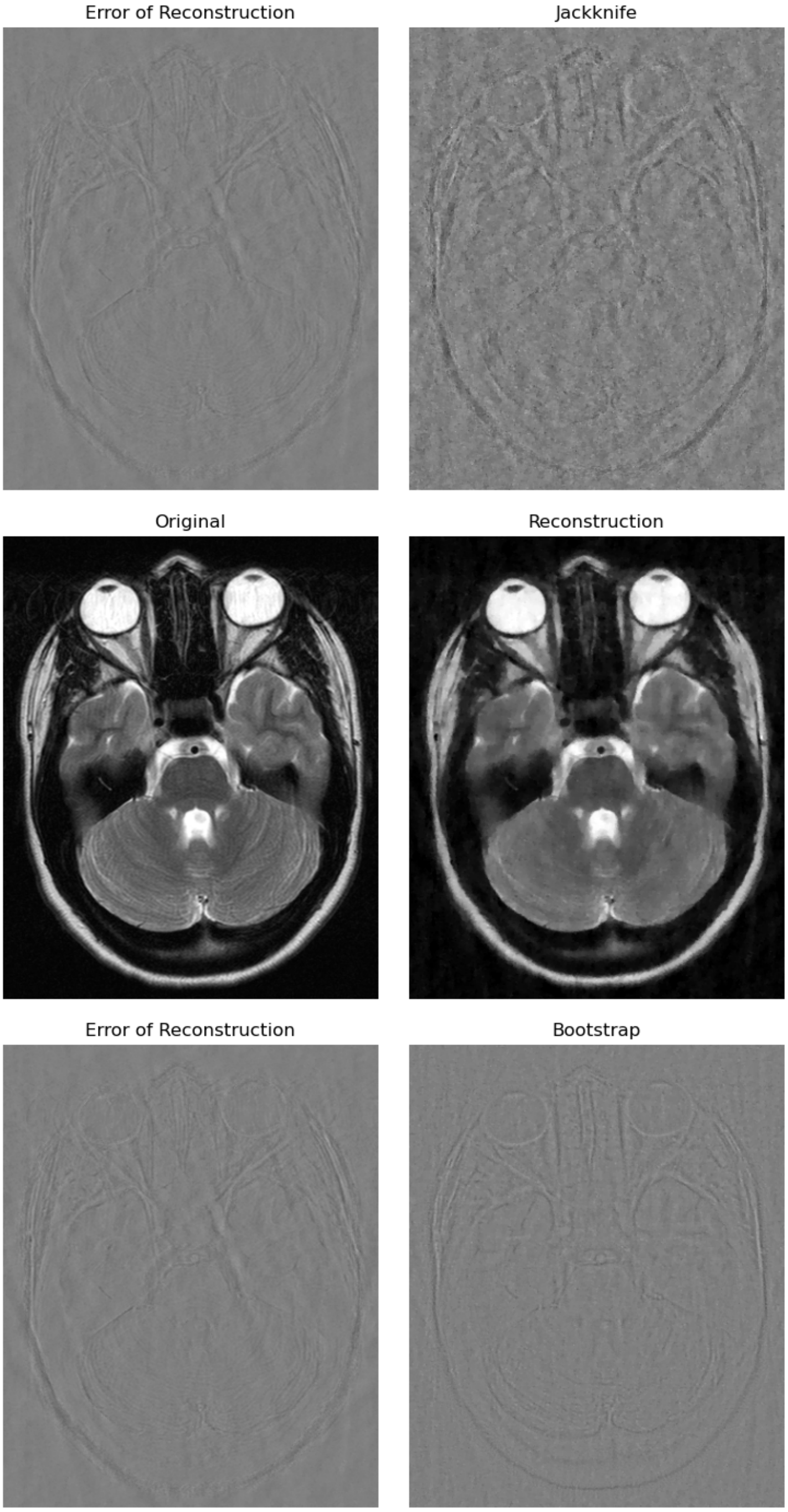

Figure 21: Radially retained sampling — slice 6

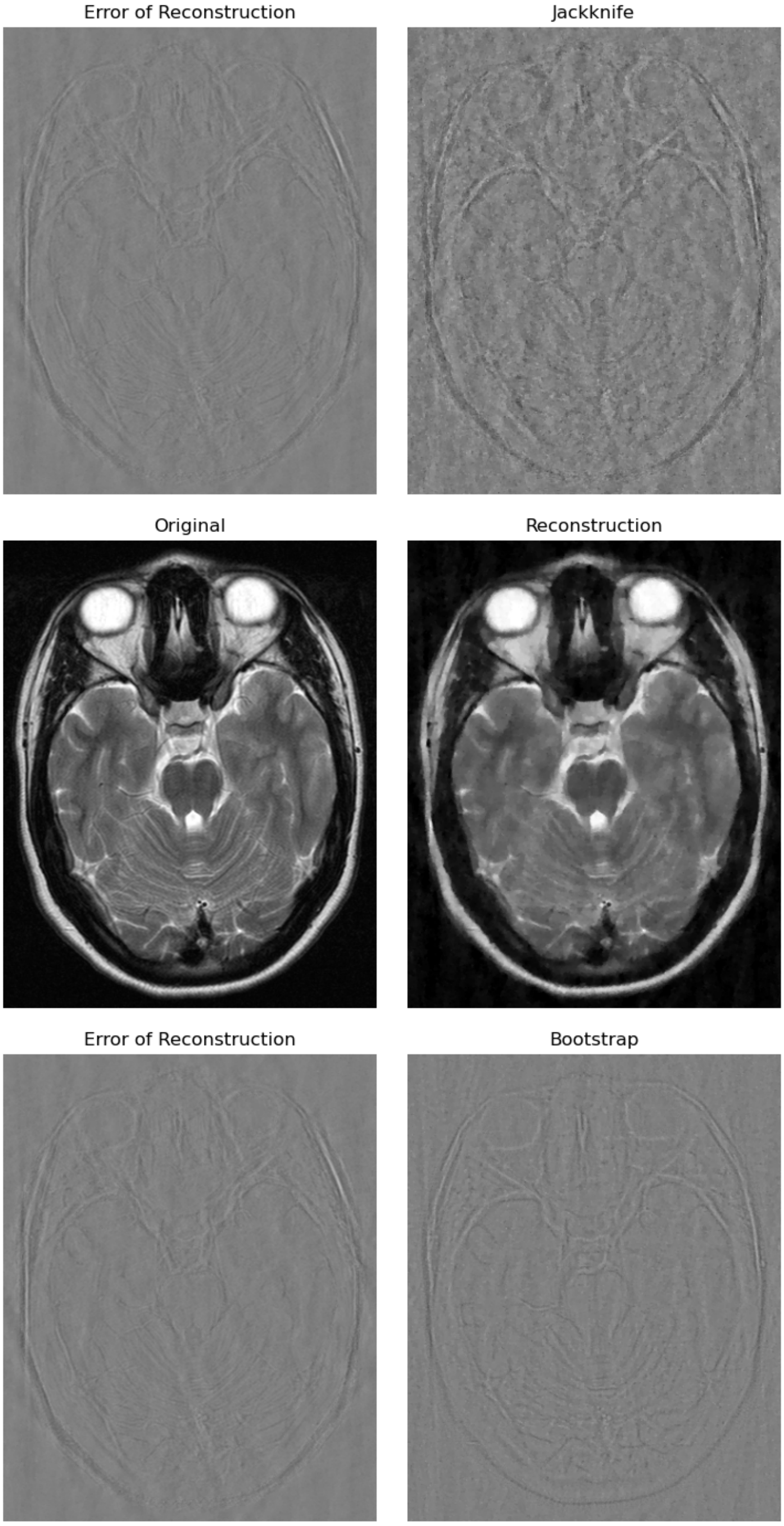

Figure 22: Radially retained sampling — slice 7

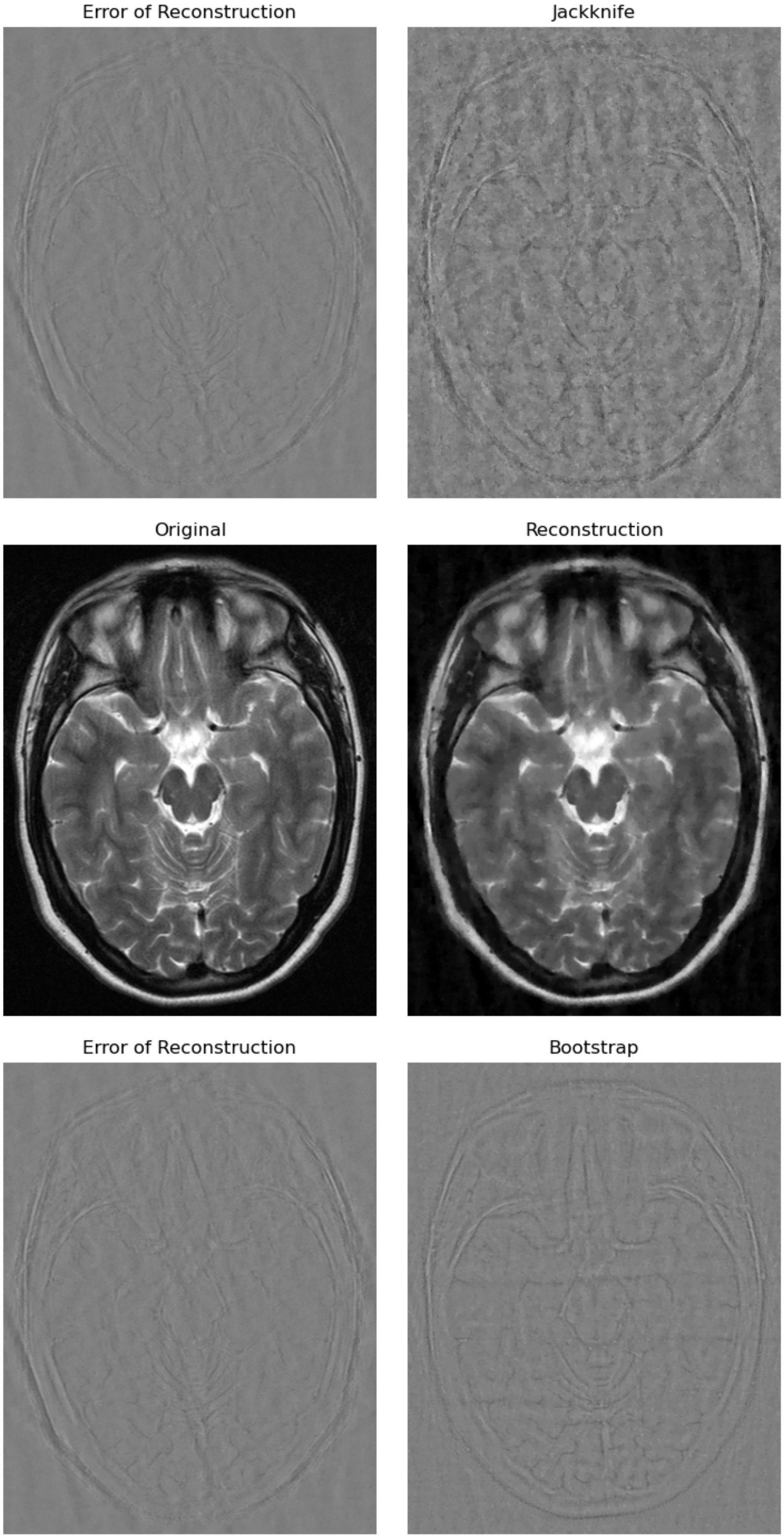

Figure 23: Radially retained sampling — slice 8

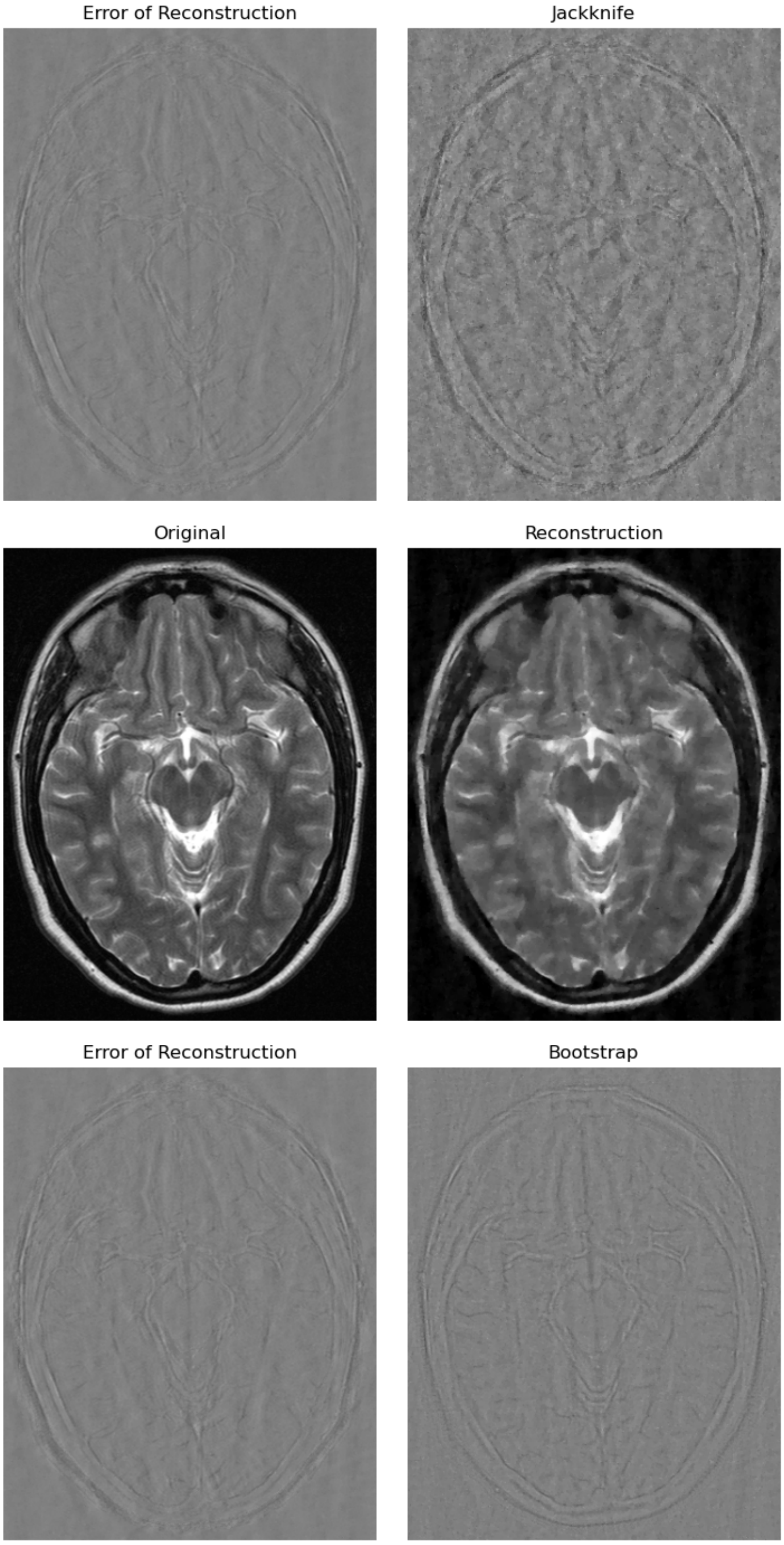

Figure 24: Radially retained sampling — slice 9

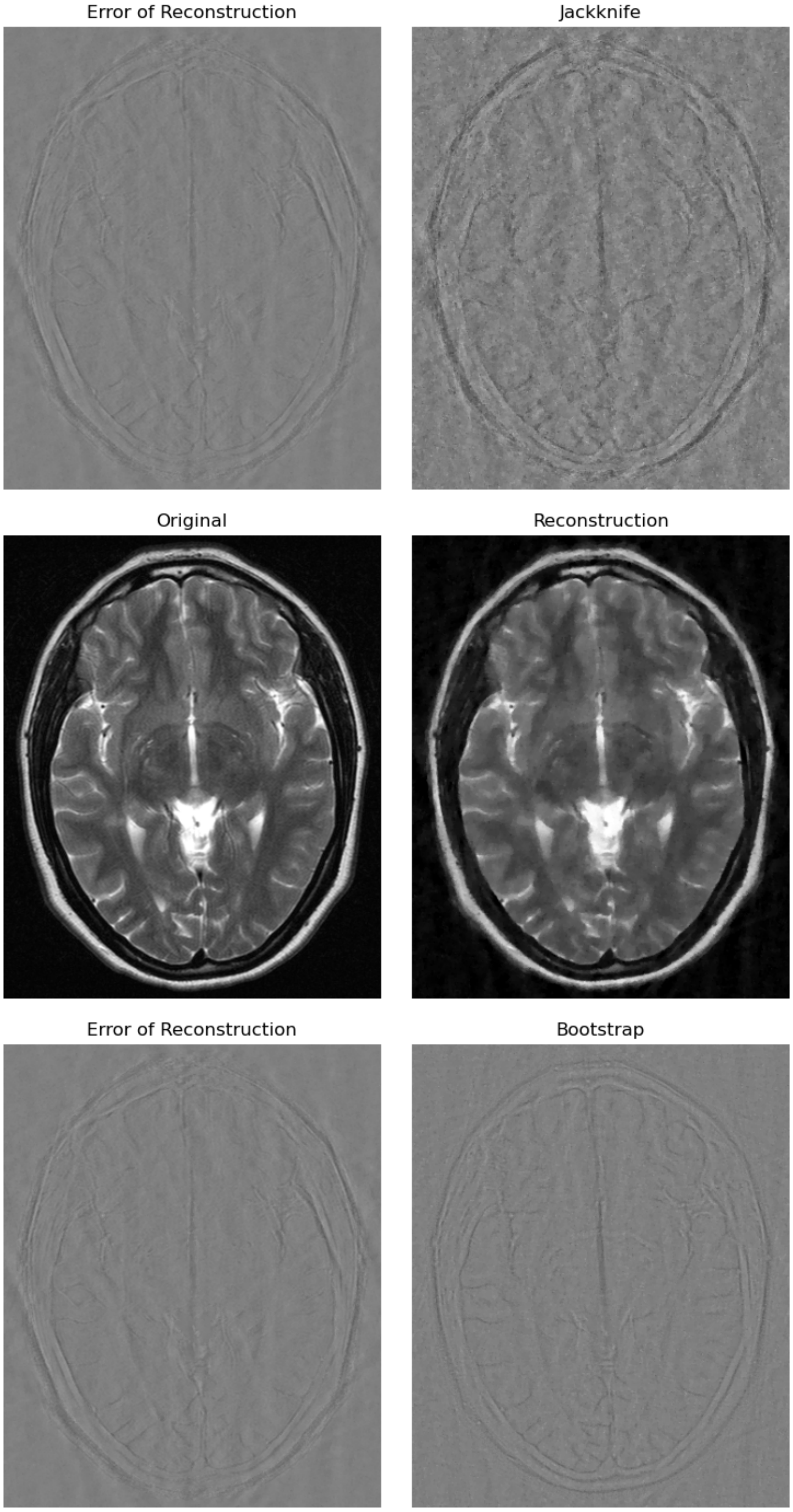

Figure 25: Radially retained sampling — slice 10

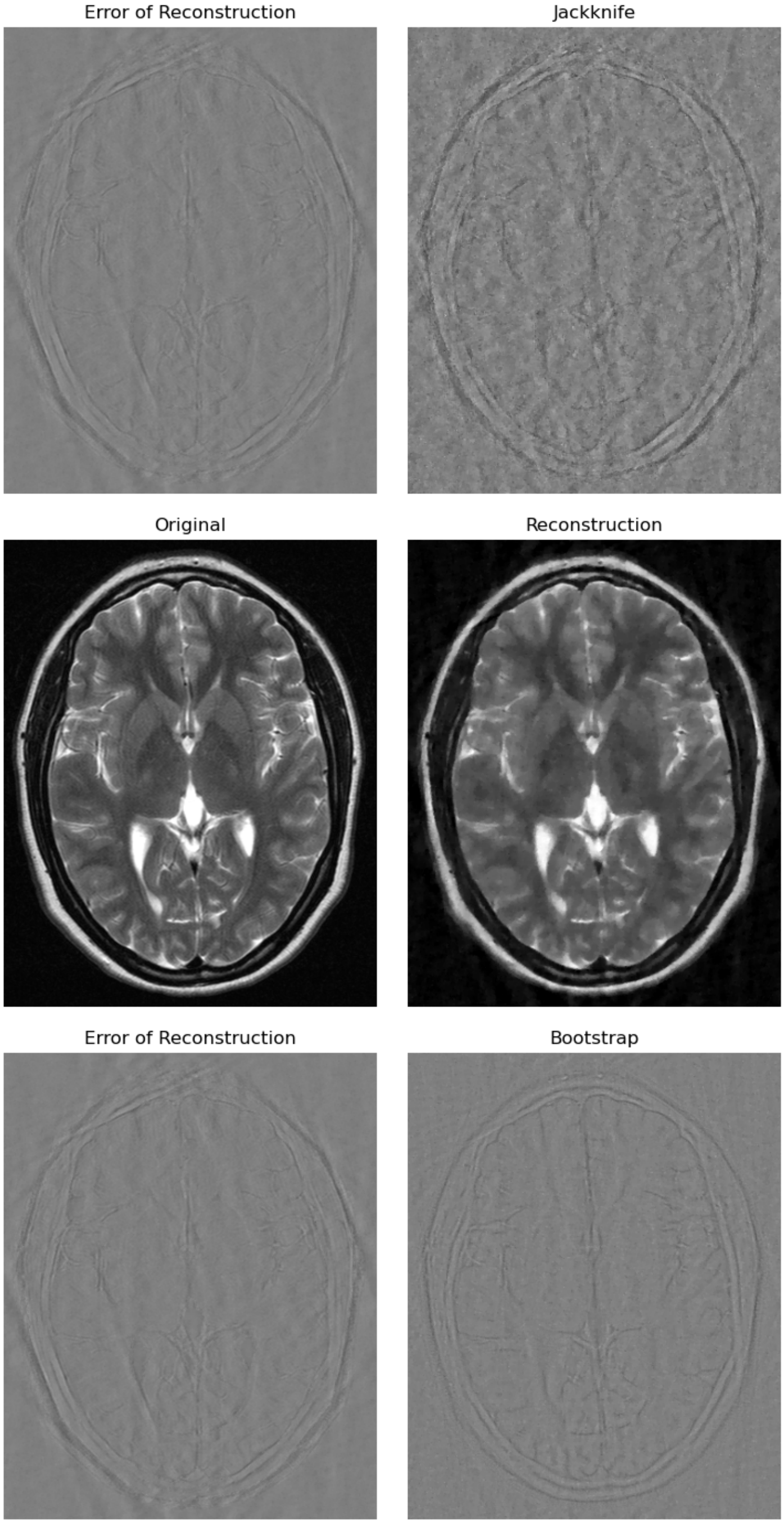

Figure 26: Radially retained sampling — slice 11

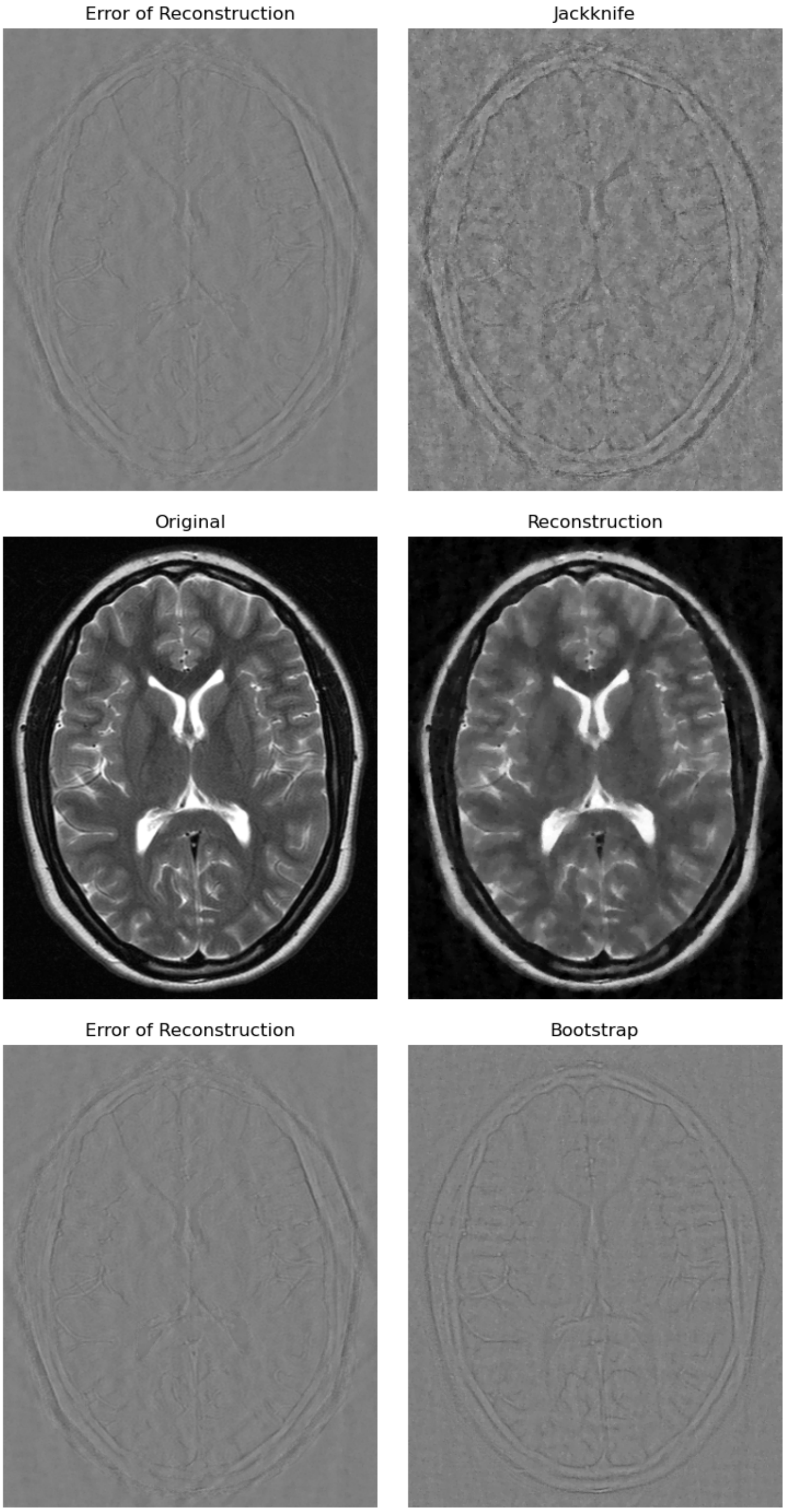

Figure 27: Radially retained sampling — slice 12

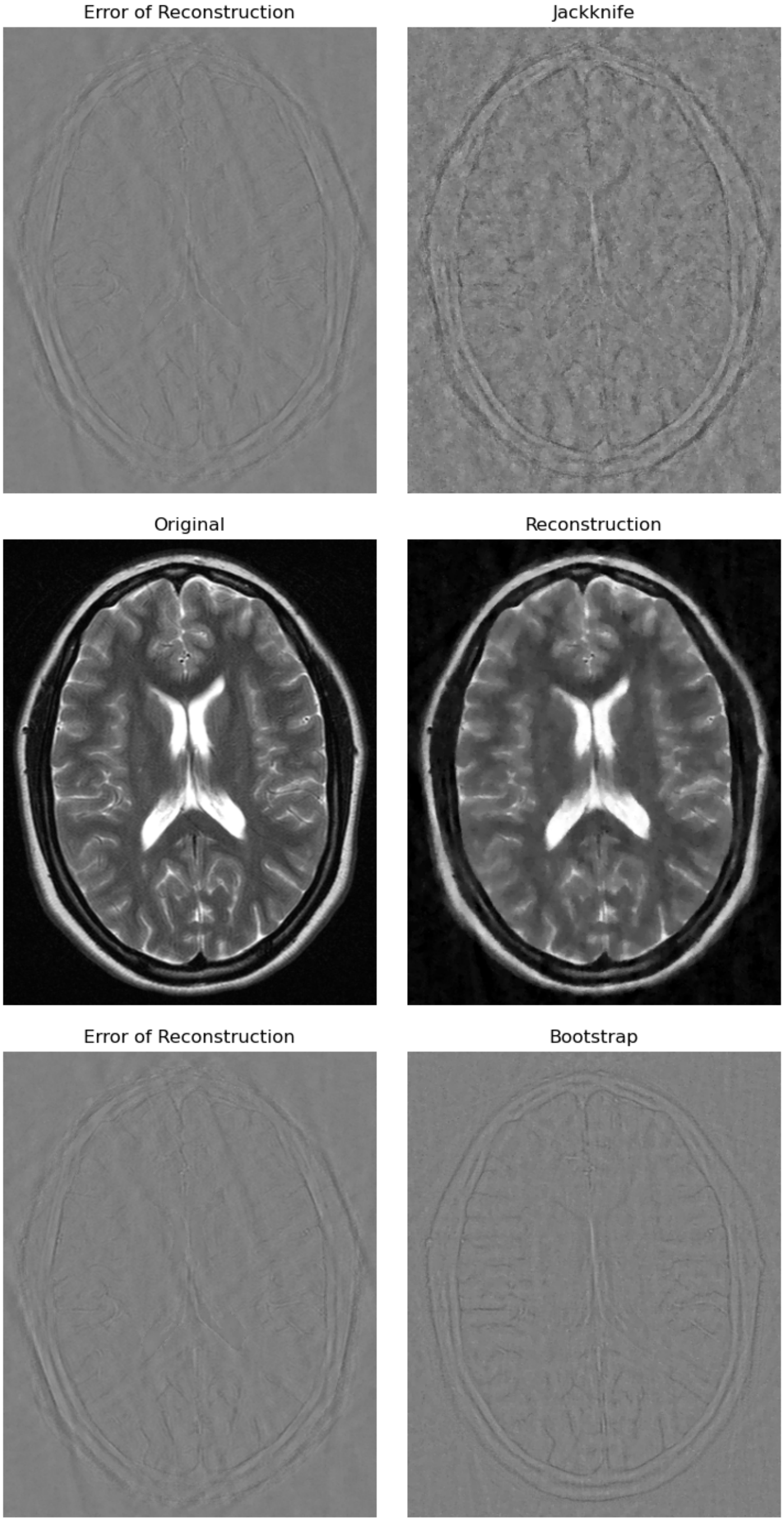

Figure 28: Radially retained sampling — slice 13

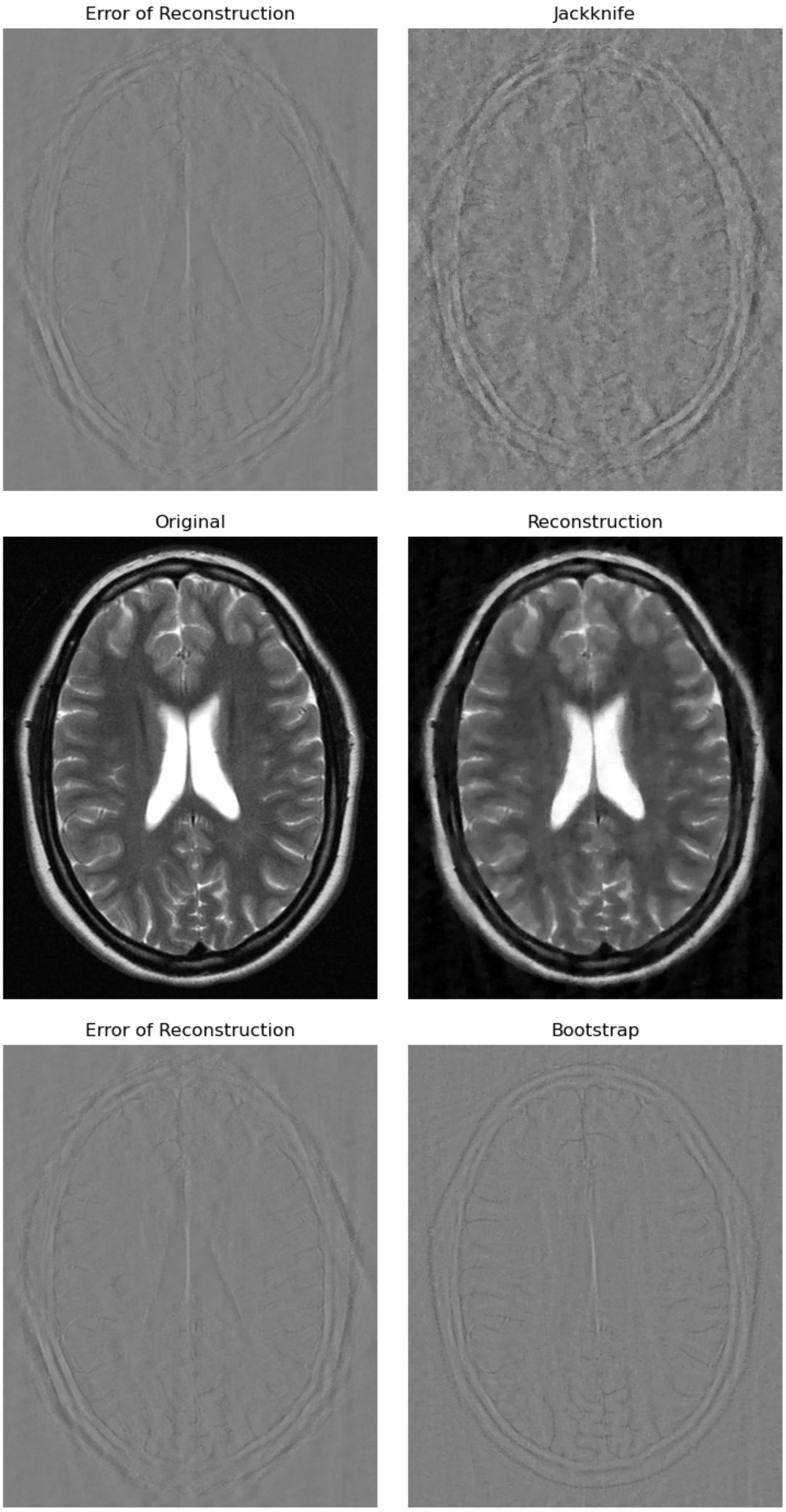

Figure 29: Radially retained sampling — slice 14

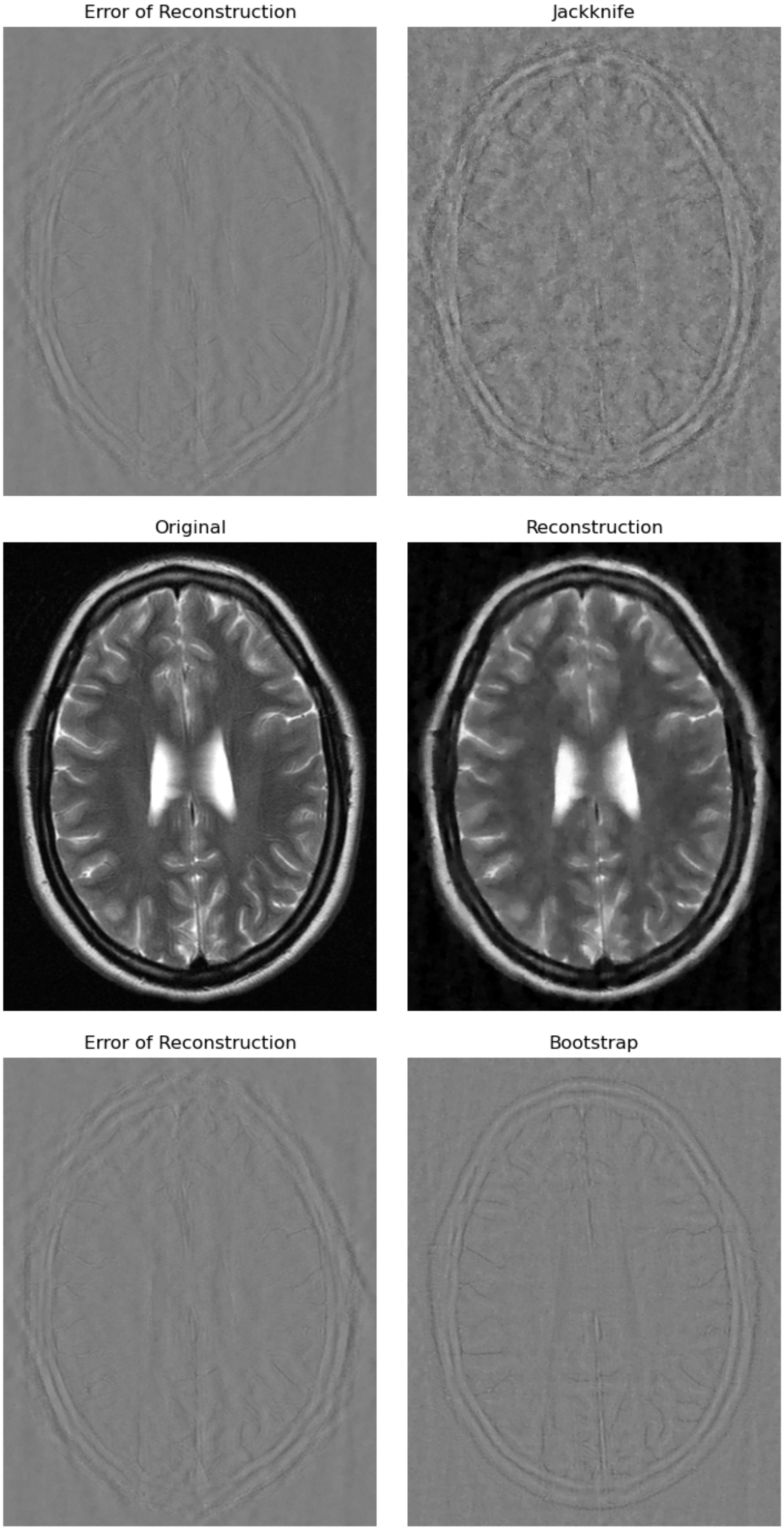

Figure 30: Radially retained sampling — slice 15

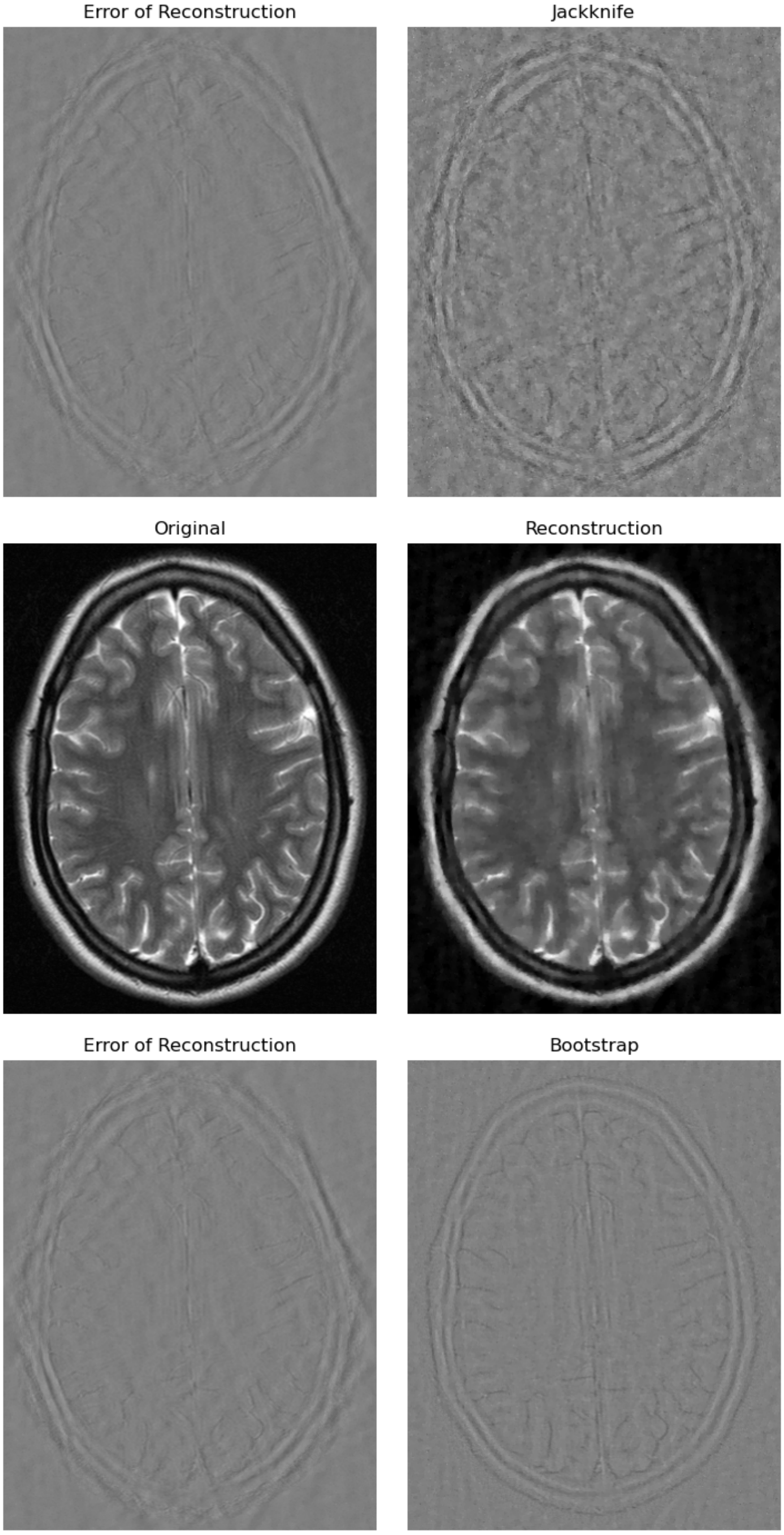

Figure 31: Radially retained sampling — slice 16

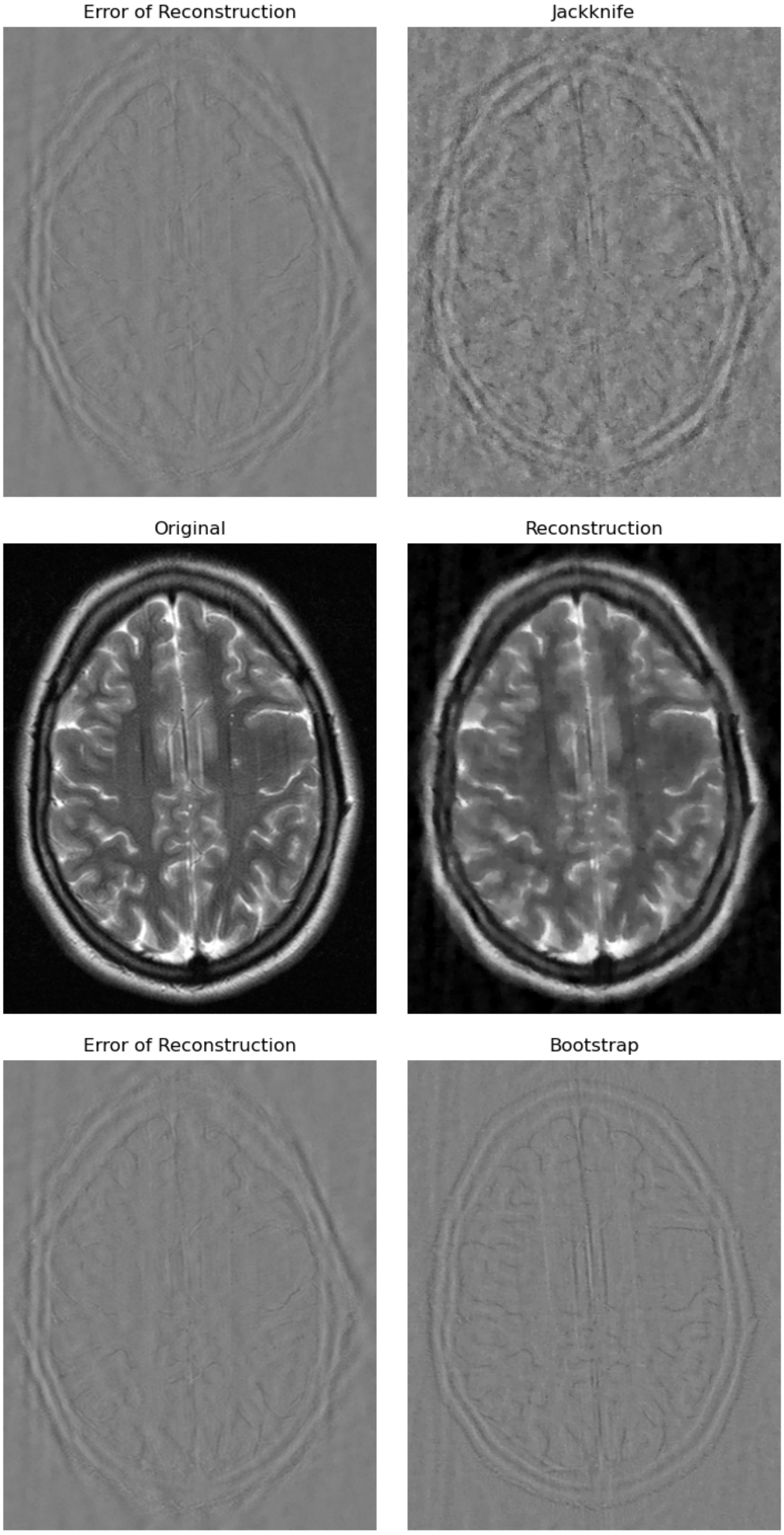

Figure 32: Radially retained sampling — slice 17

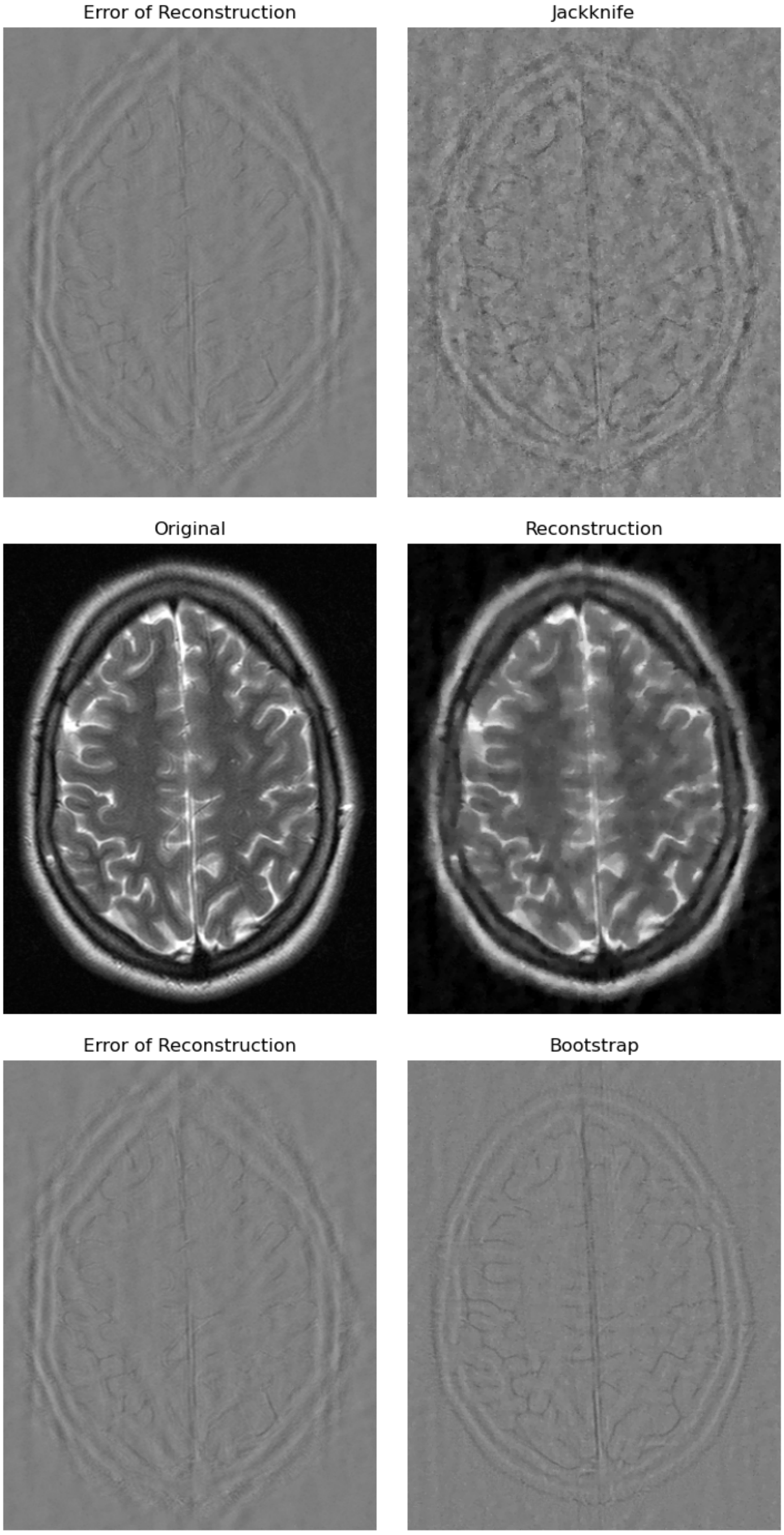

Figure 33: Radially retained sampling — slice 18

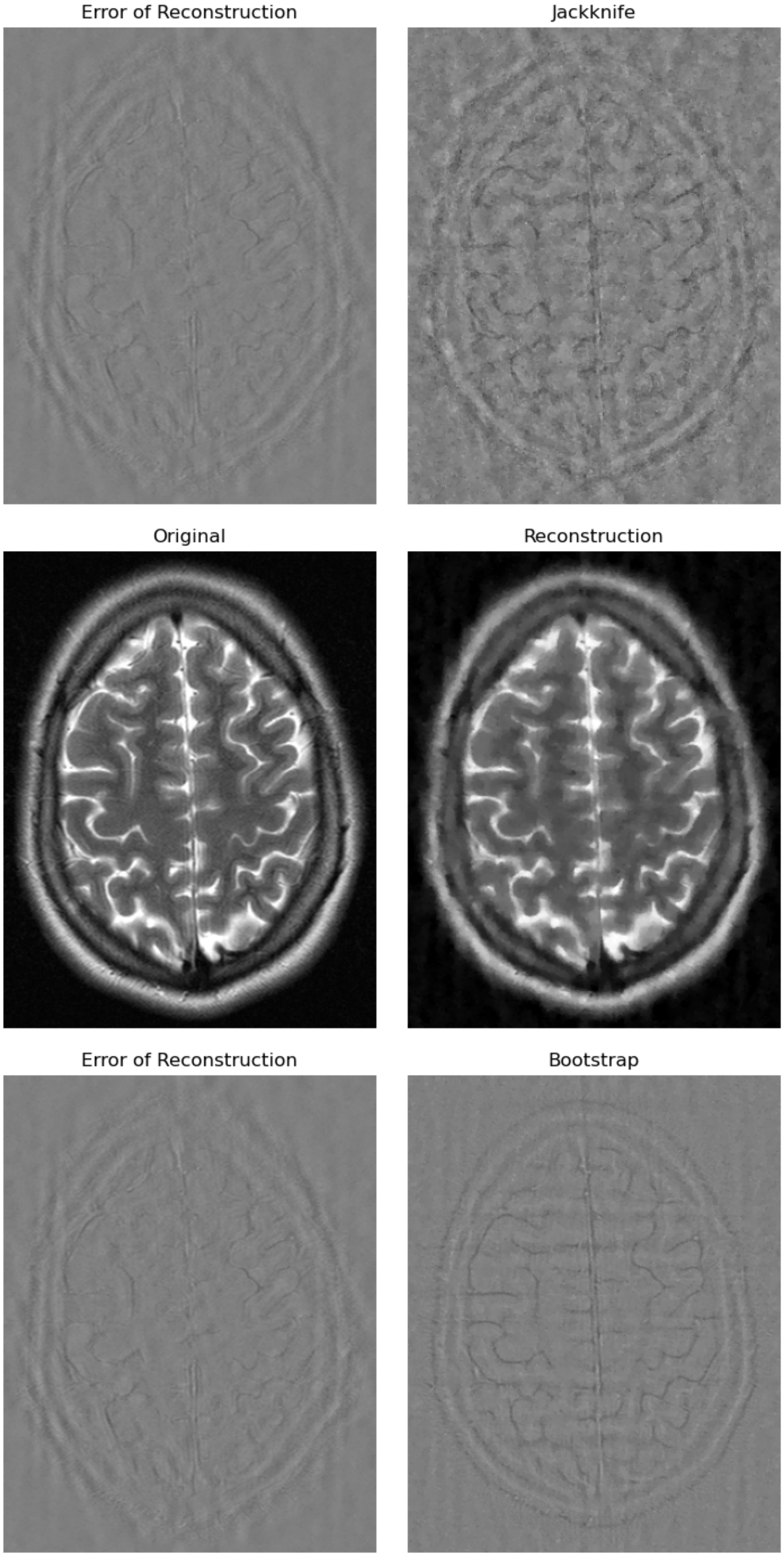

Figure 34: Radially retained sampling — slice 19

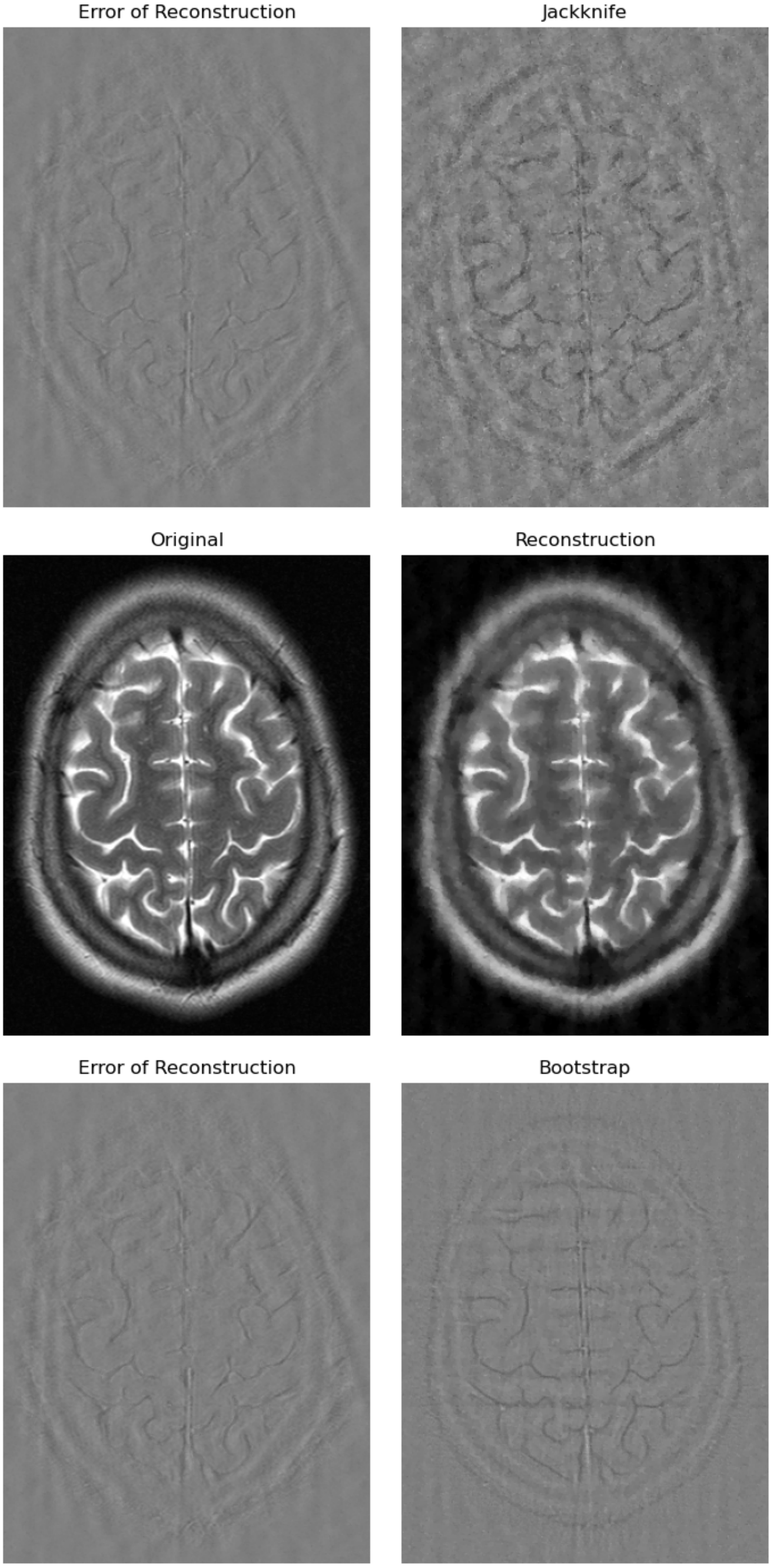

Figure 35: Radially retained sampling — slice 20

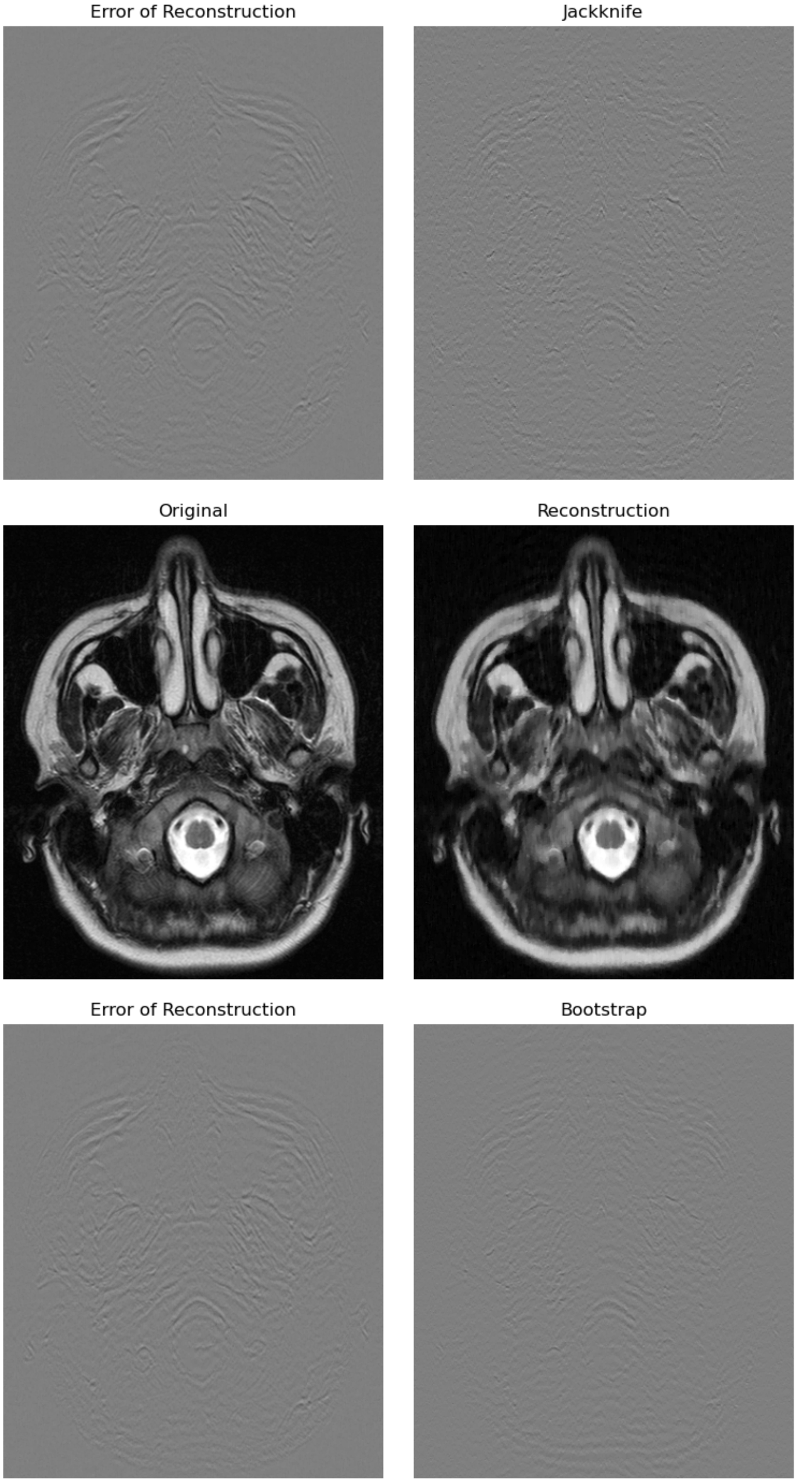

Figure 36: Horizontally retained sampling — slice 1

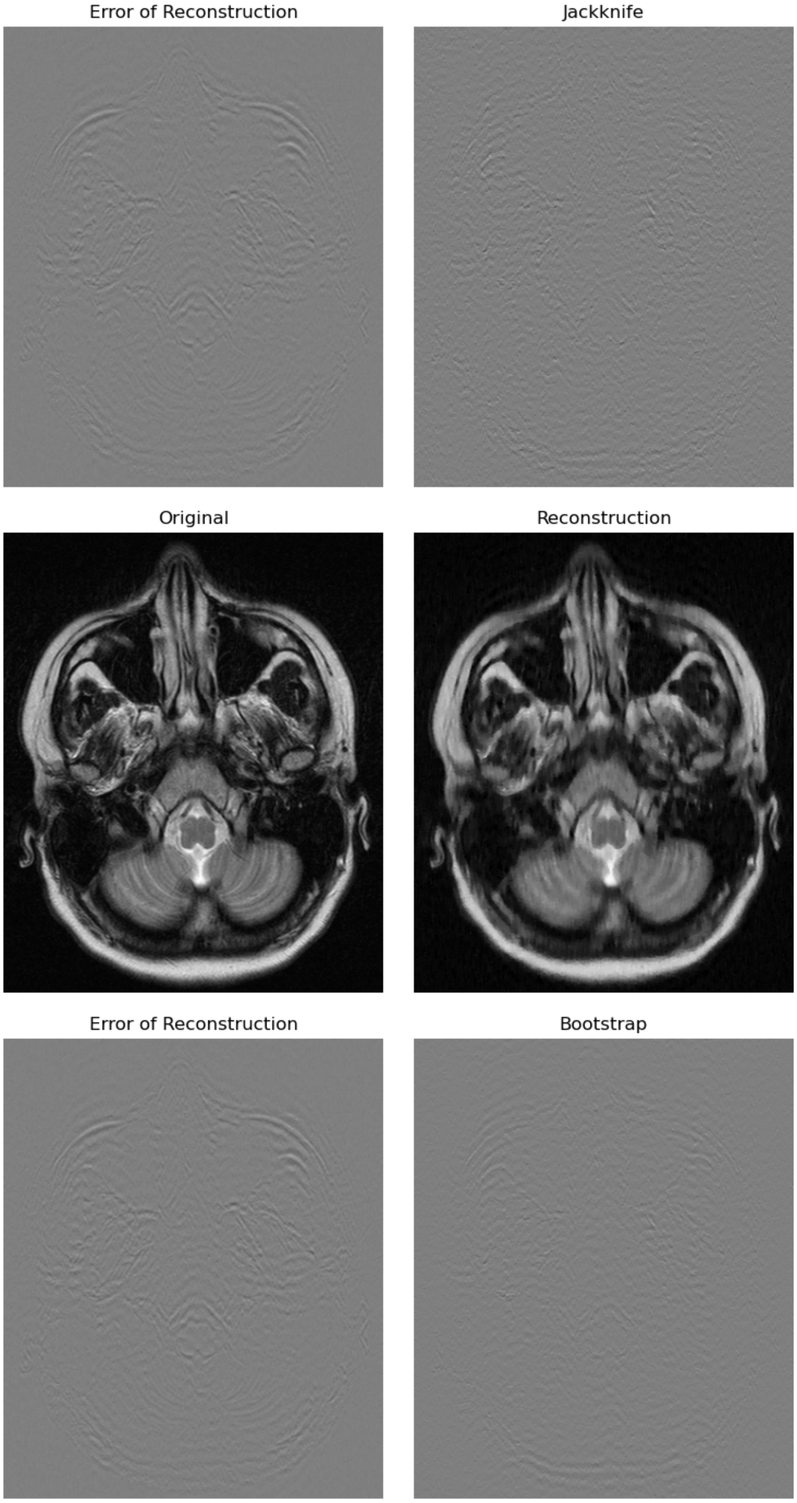

Figure 37: Horizontally retained sampling — slice 2

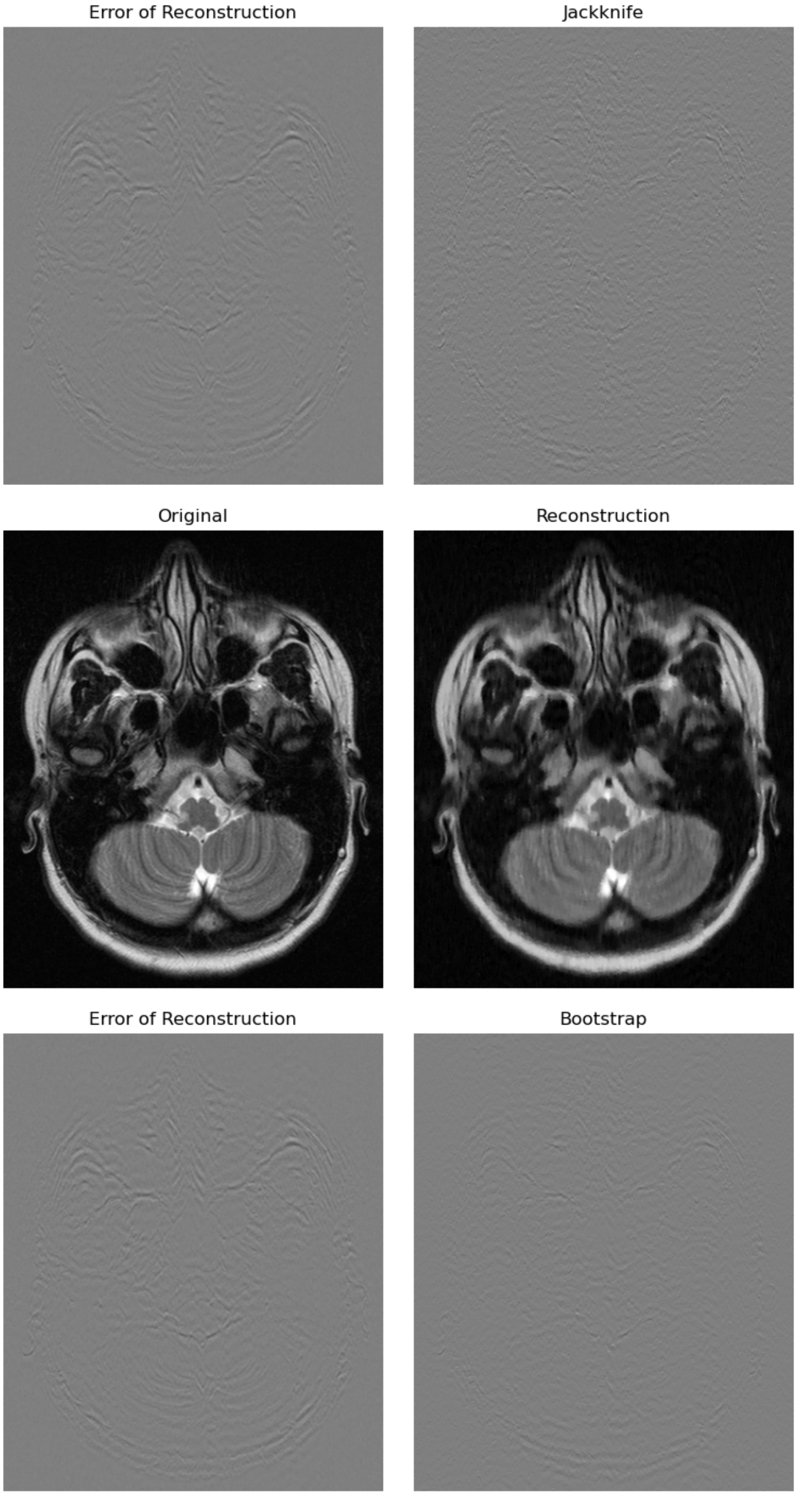

Figure 38: Horizontally retained sampling — slice 3

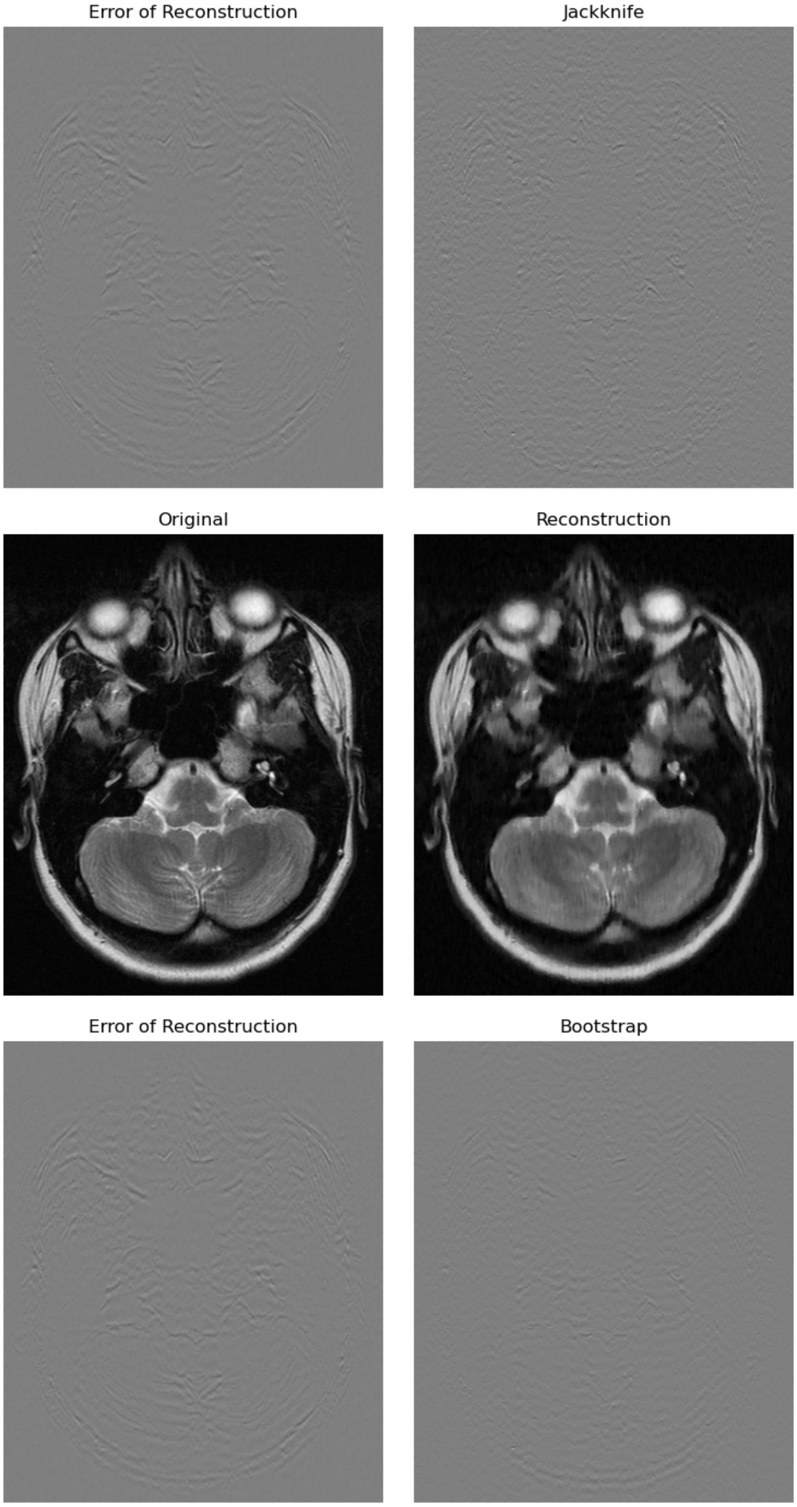

Figure 39: Horizontally retained sampling — slice 4

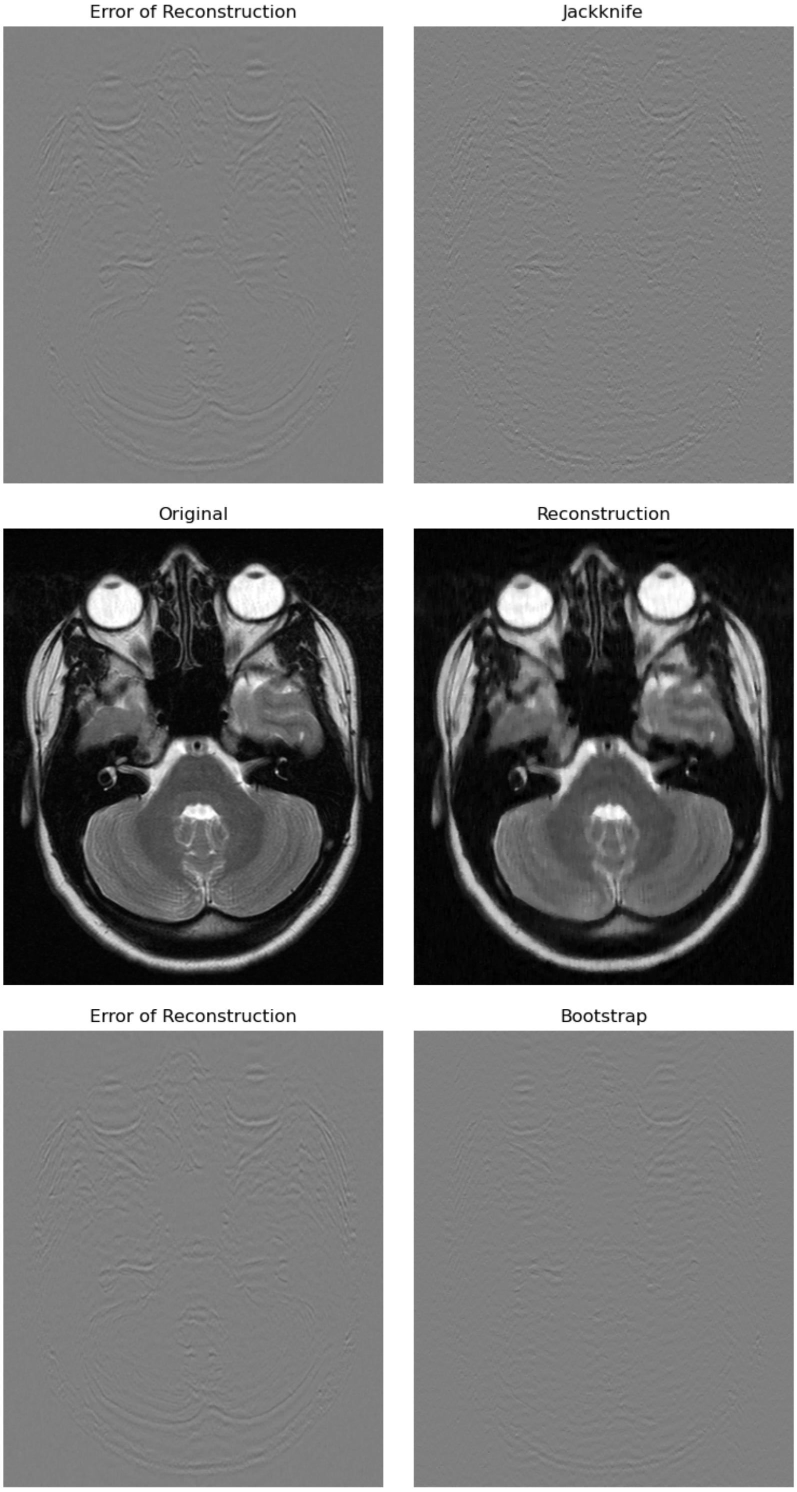

Figure 40: Horizontally retained sampling — slice 5

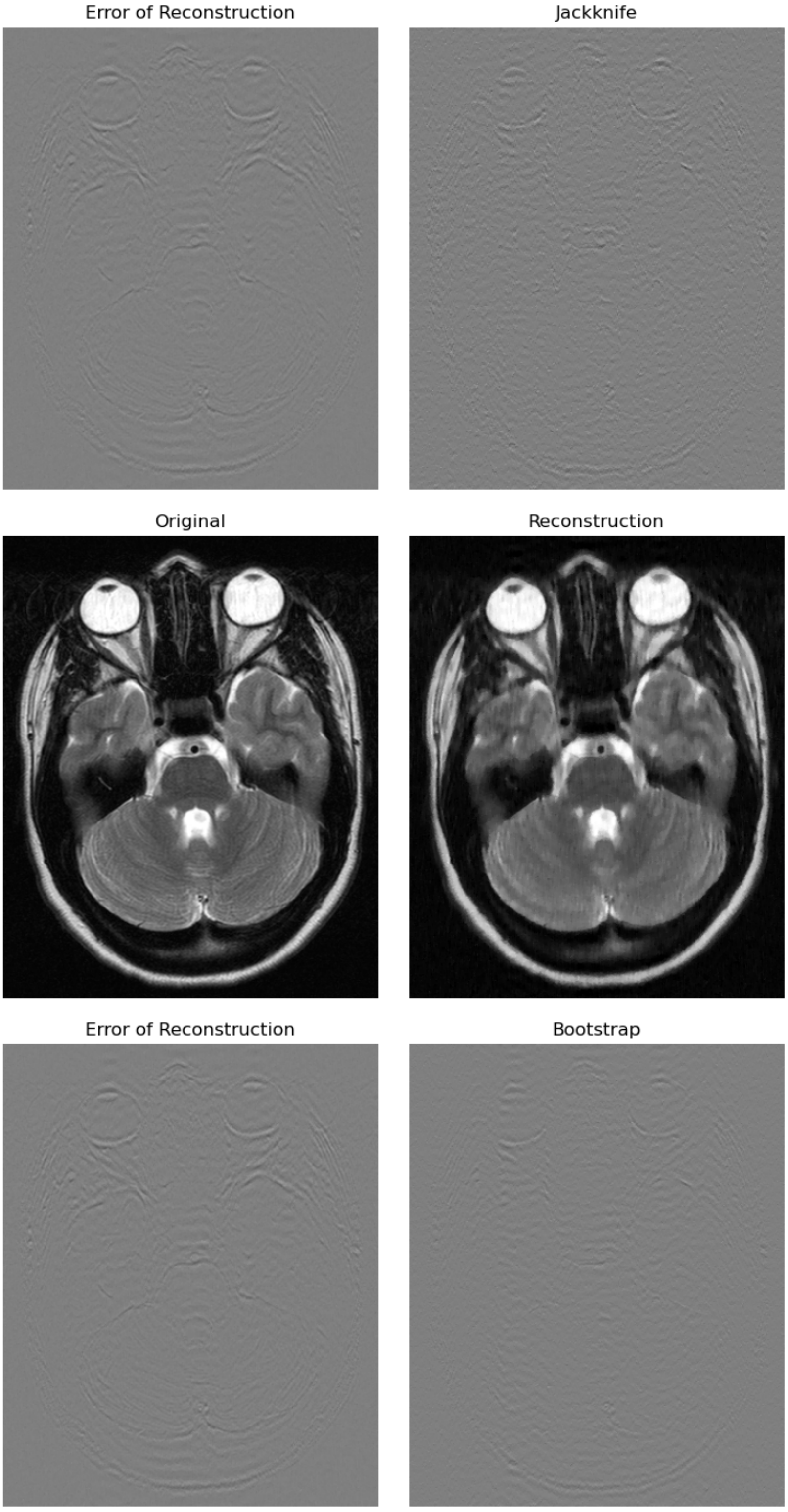

Figure 41: Horizontally retained sampling — slice 6

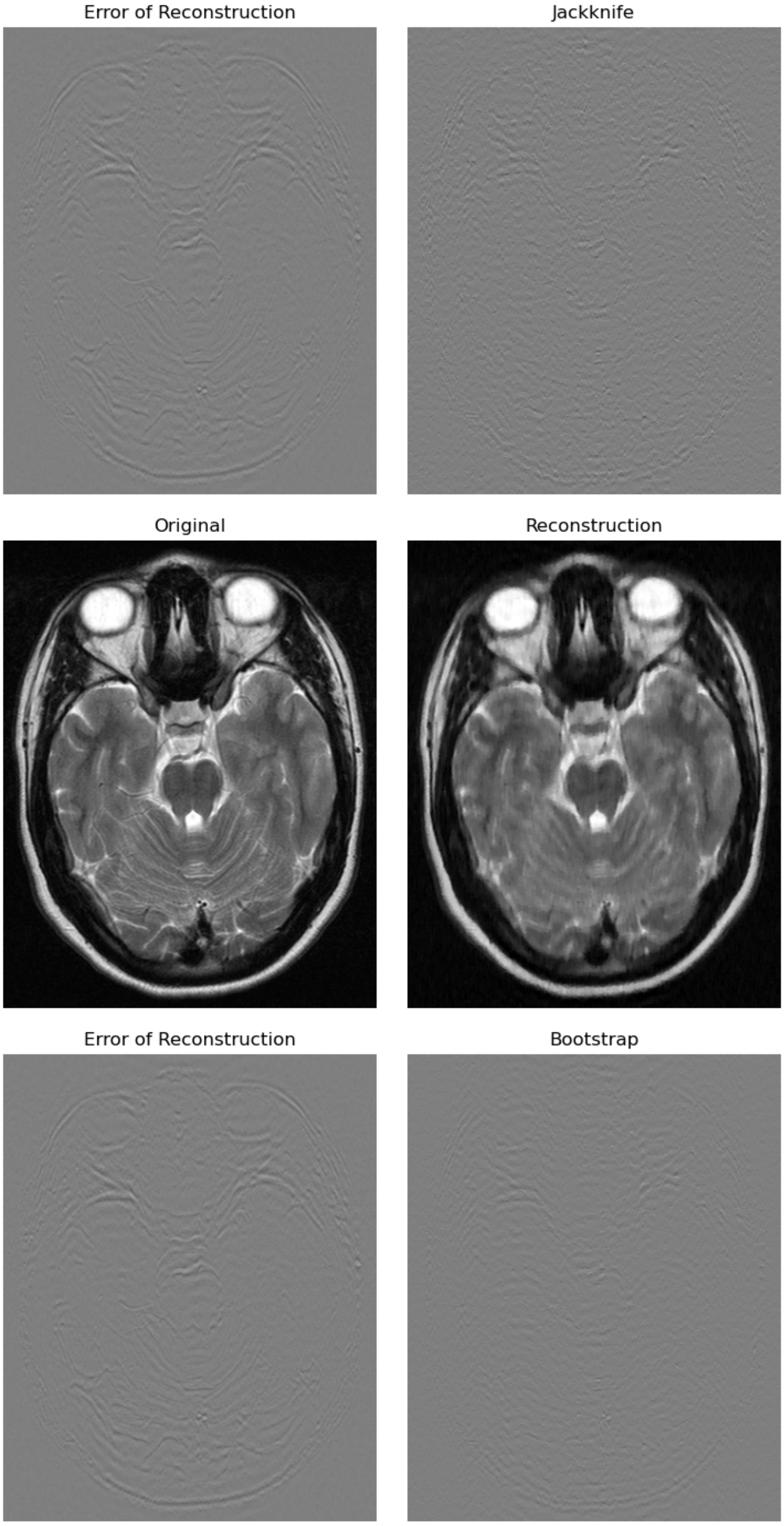

Figure 42: Horizontally retained sampling — slice 7

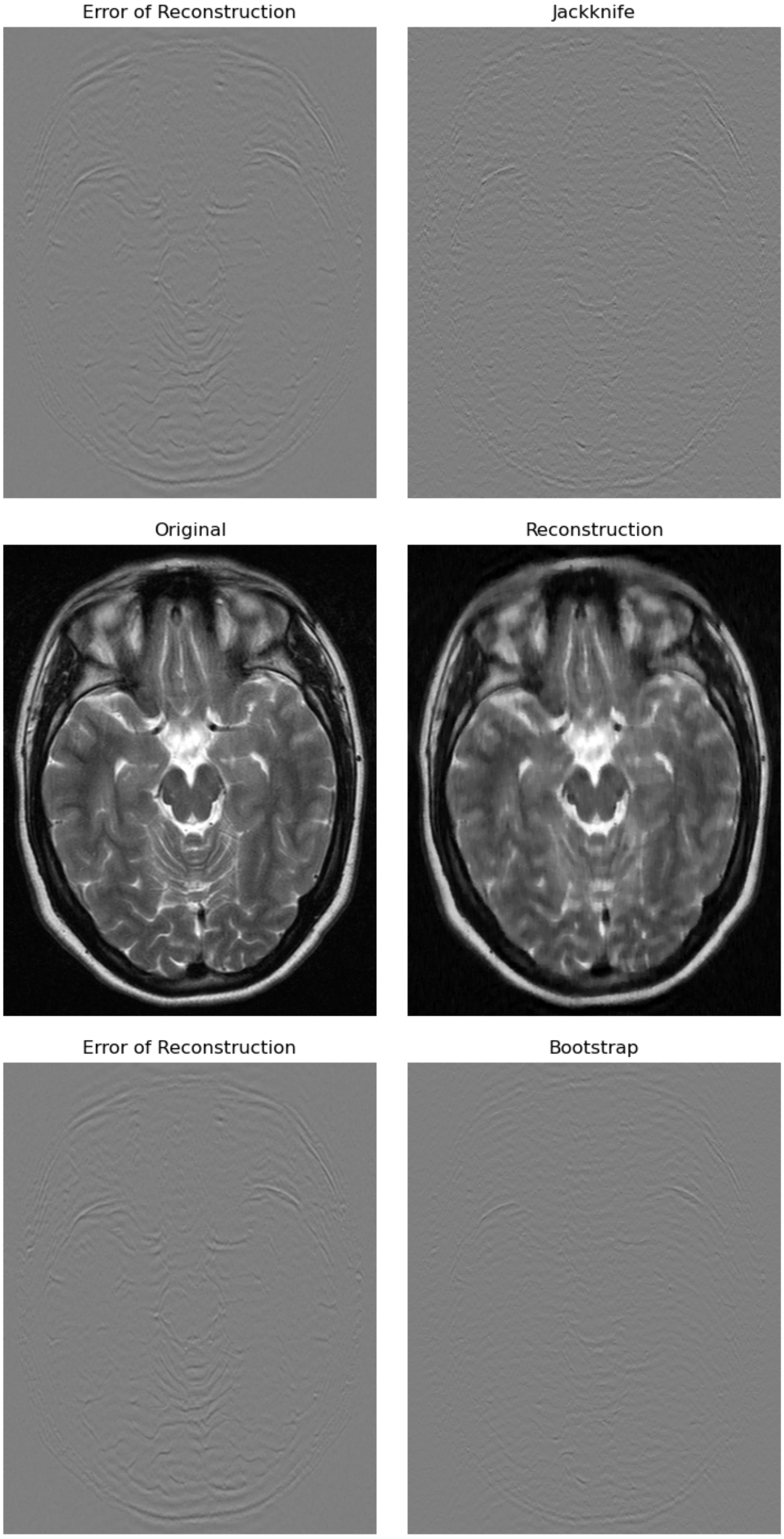

Figure 43: Horizontally retained sampling — slice 8

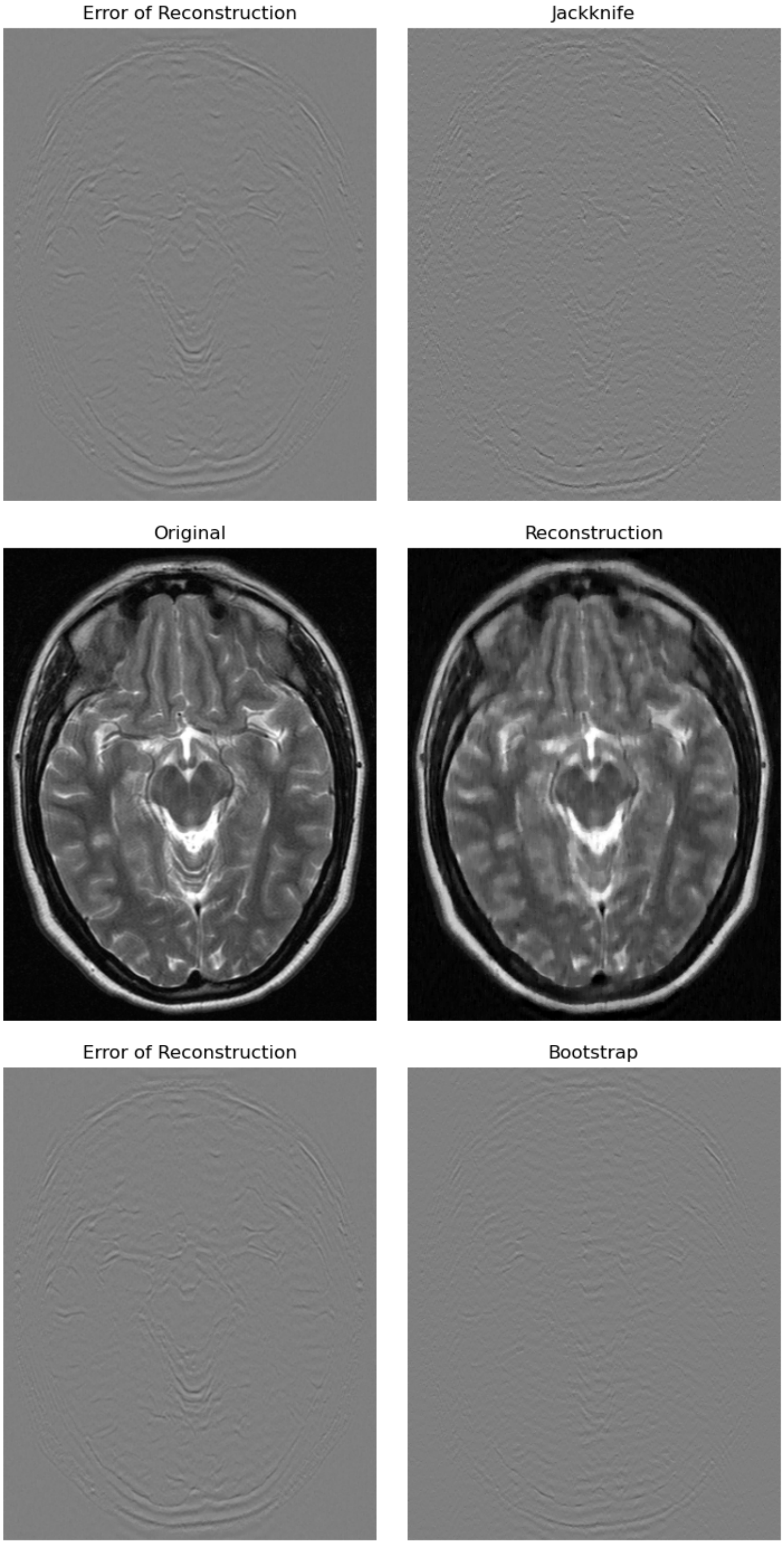

Figure 44: Horizontally retained sampling — slice 9

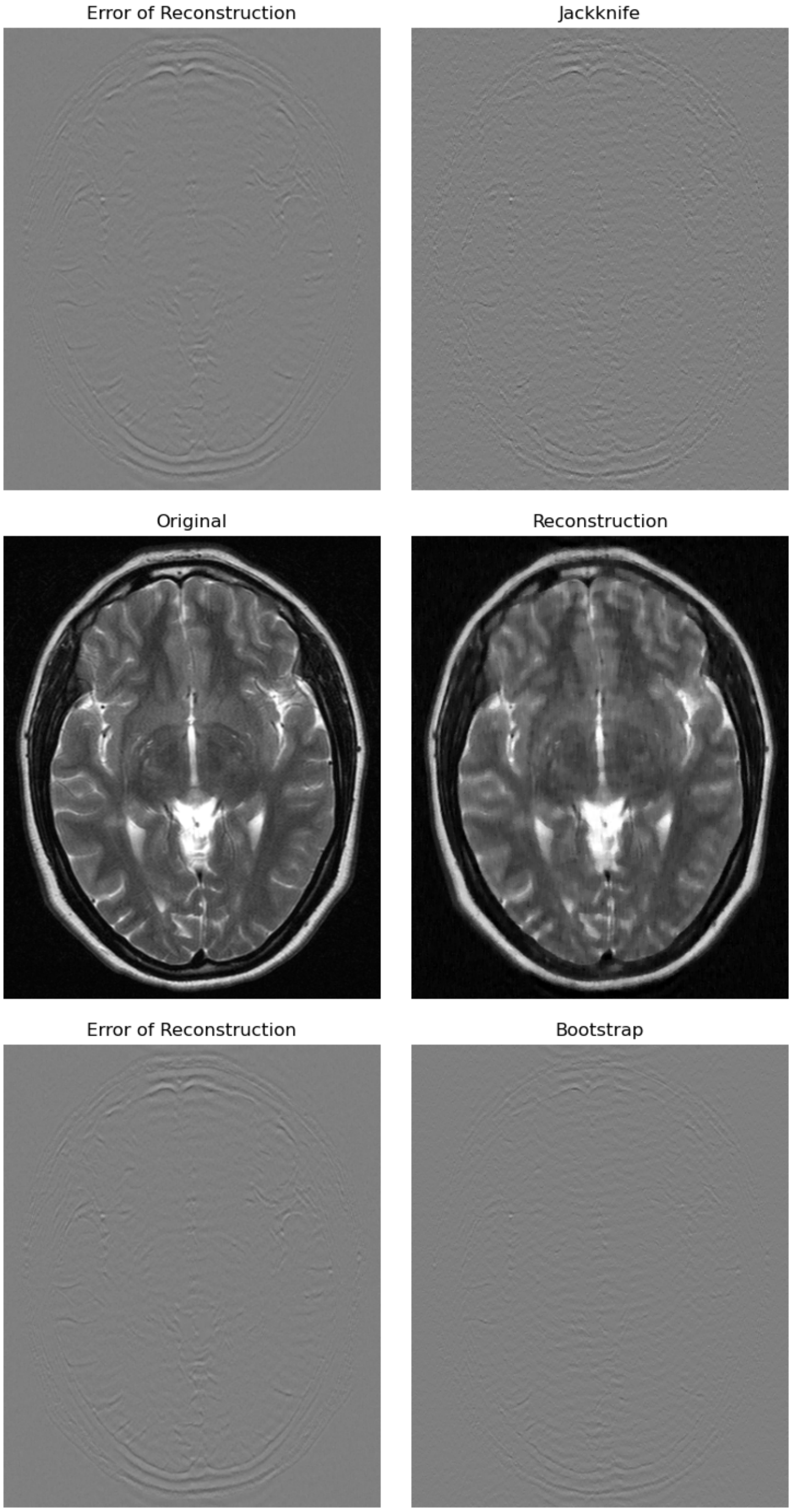

Figure 45: Horizontally retained sampling — slice 10

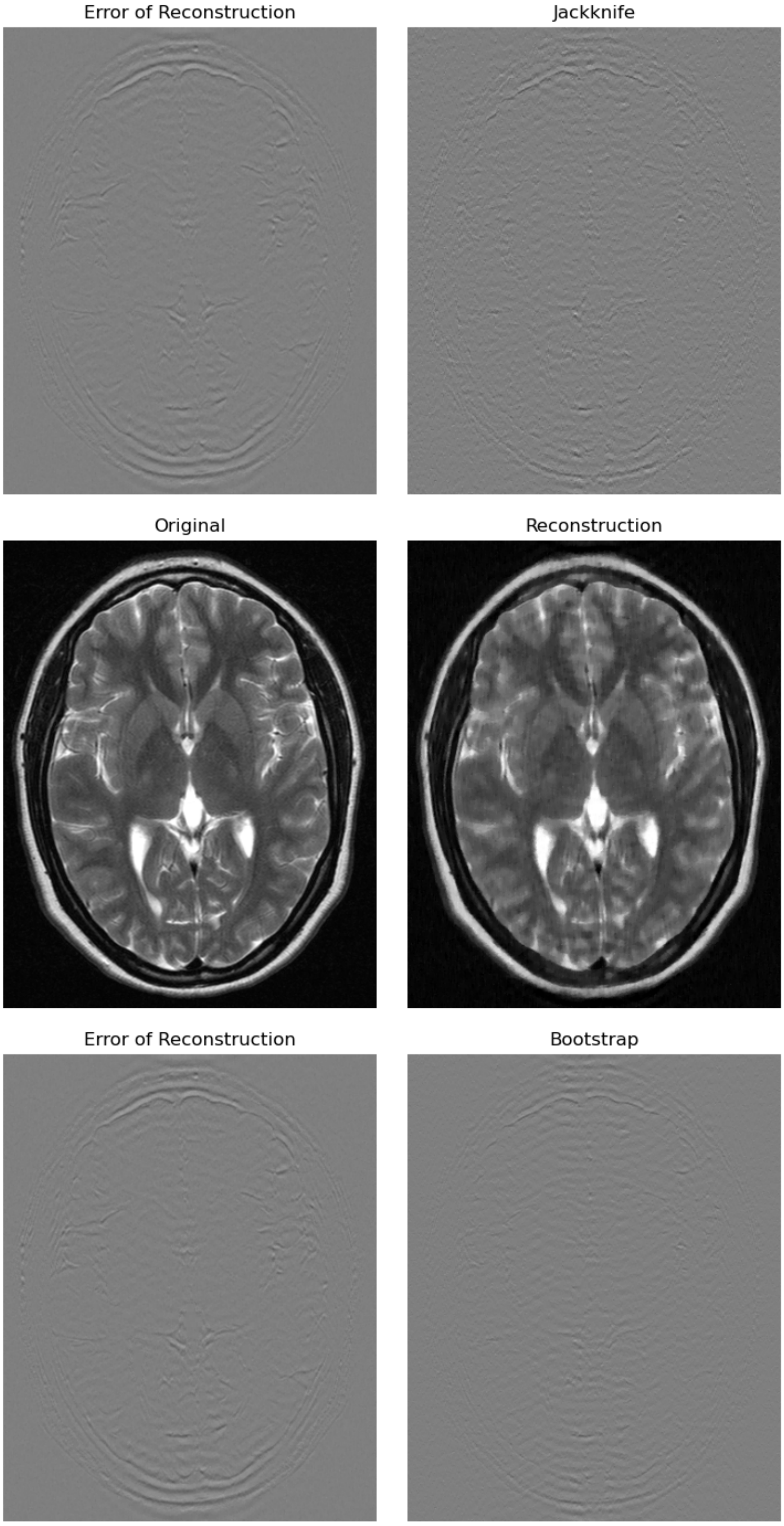

Figure 46: Horizontally retained sampling — slice 11

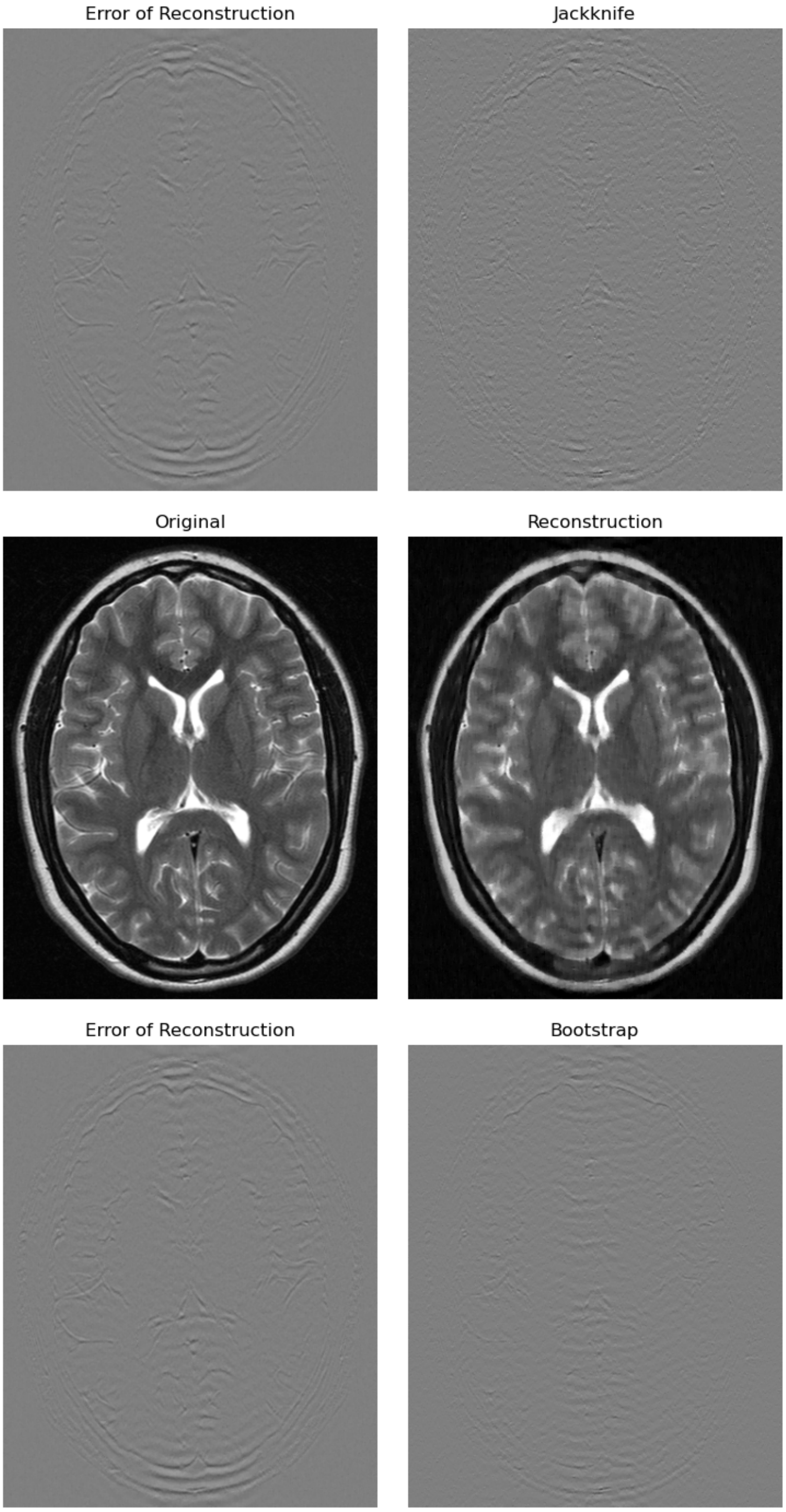

Figure 47: Horizontally retained sampling — slice 12

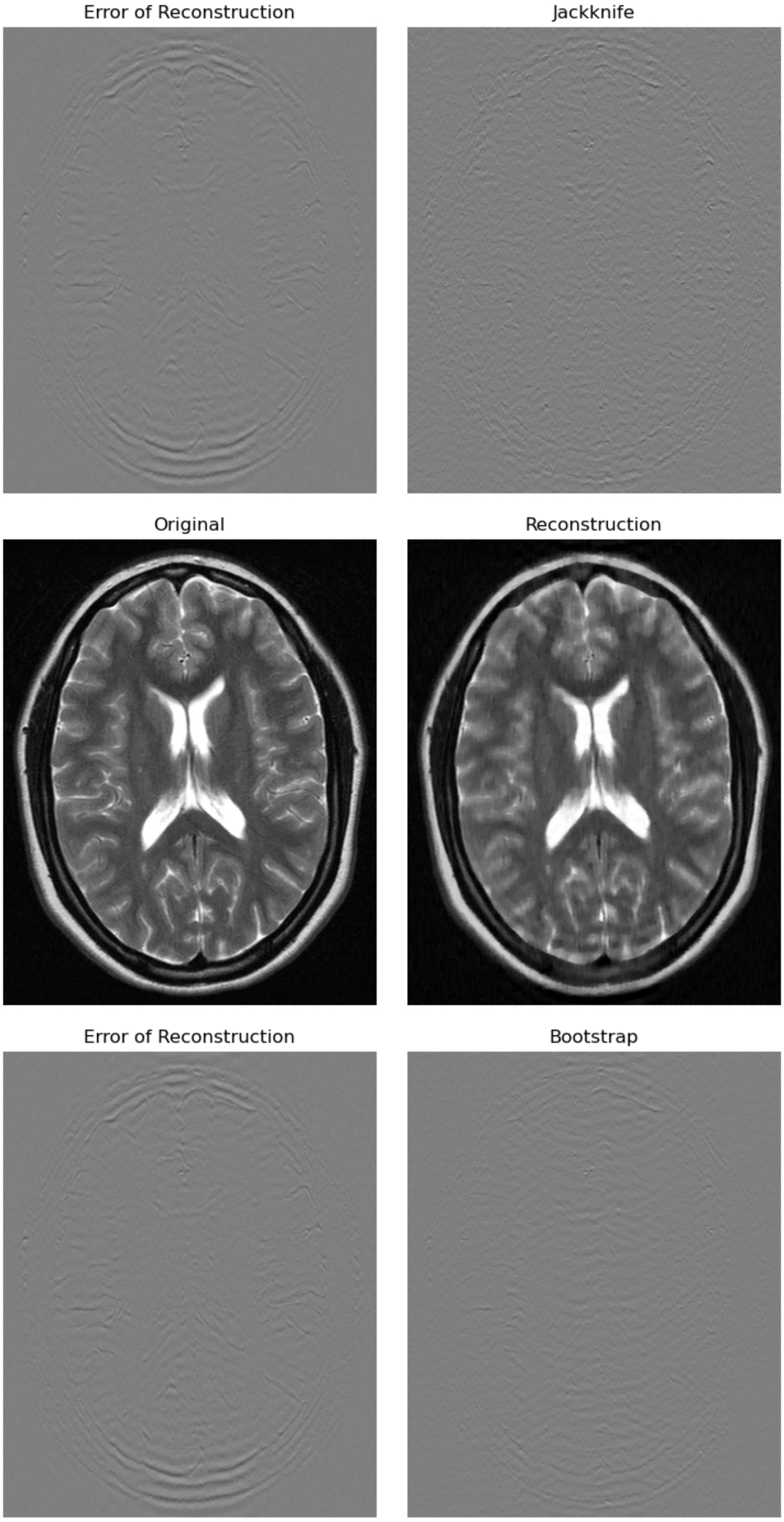

Figure 48: Horizontally retained sampling — slice 13

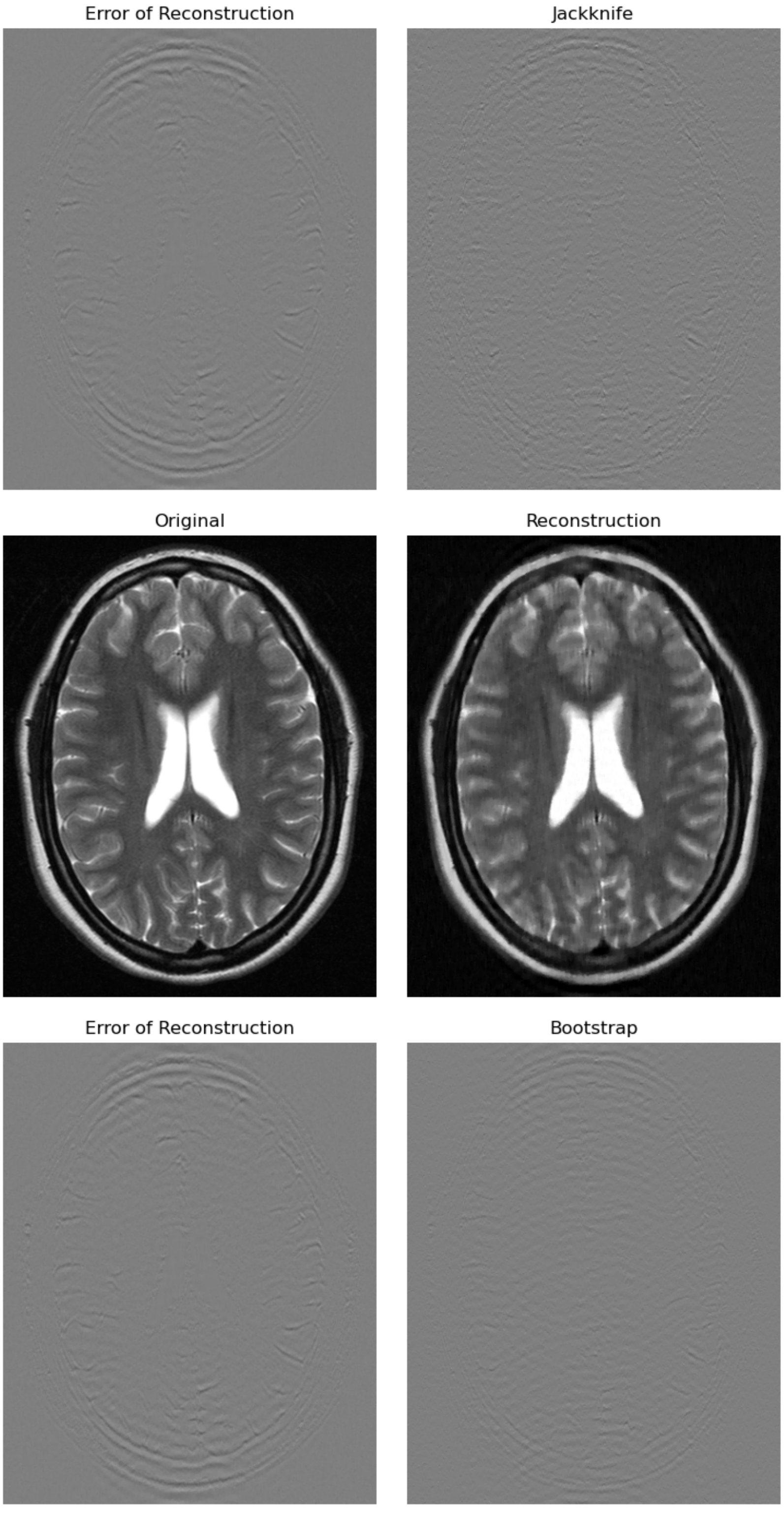

Figure 49: Horizontally retained sampling — slice 14

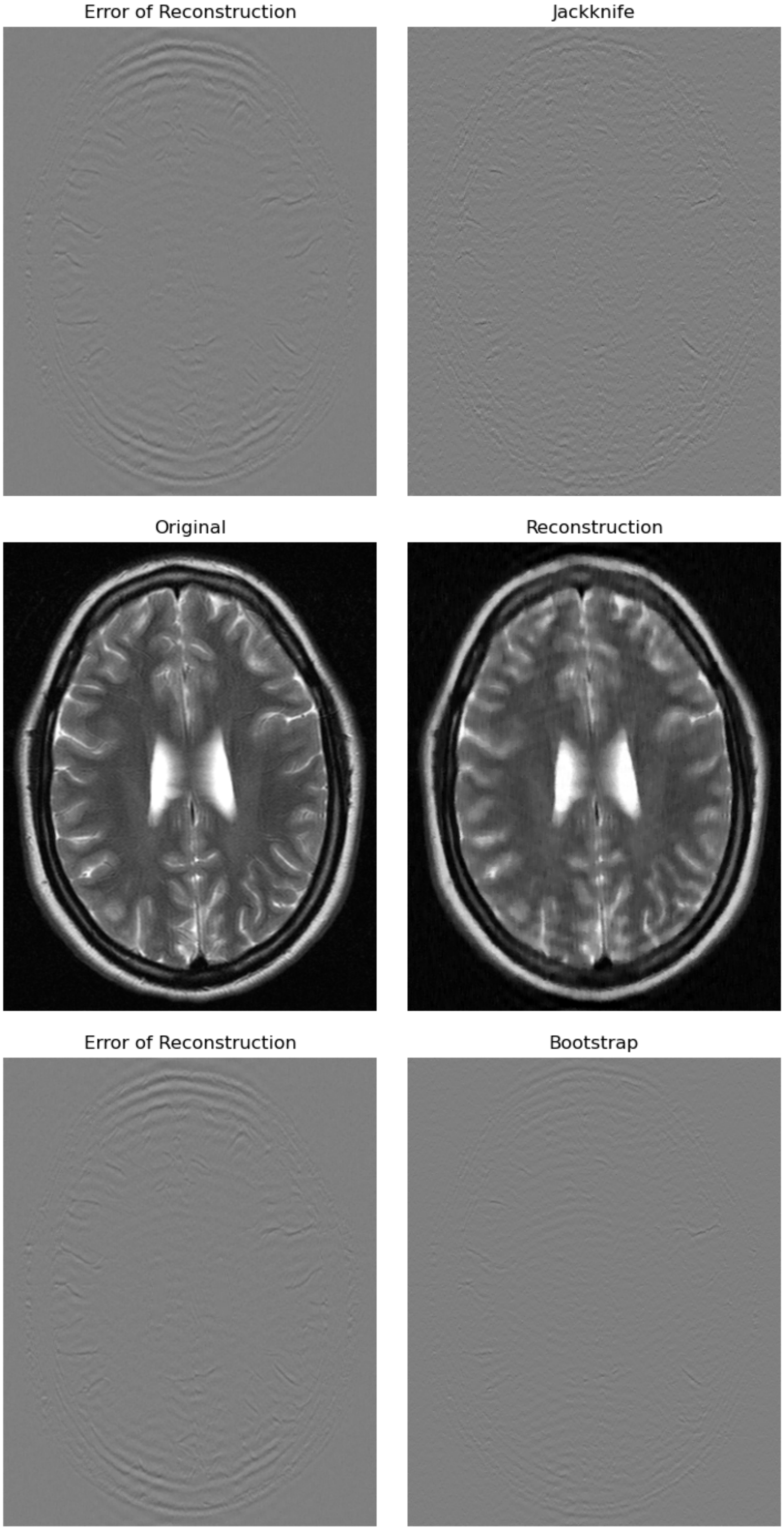

Figure 50: Horizontally retained sampling — slice 15

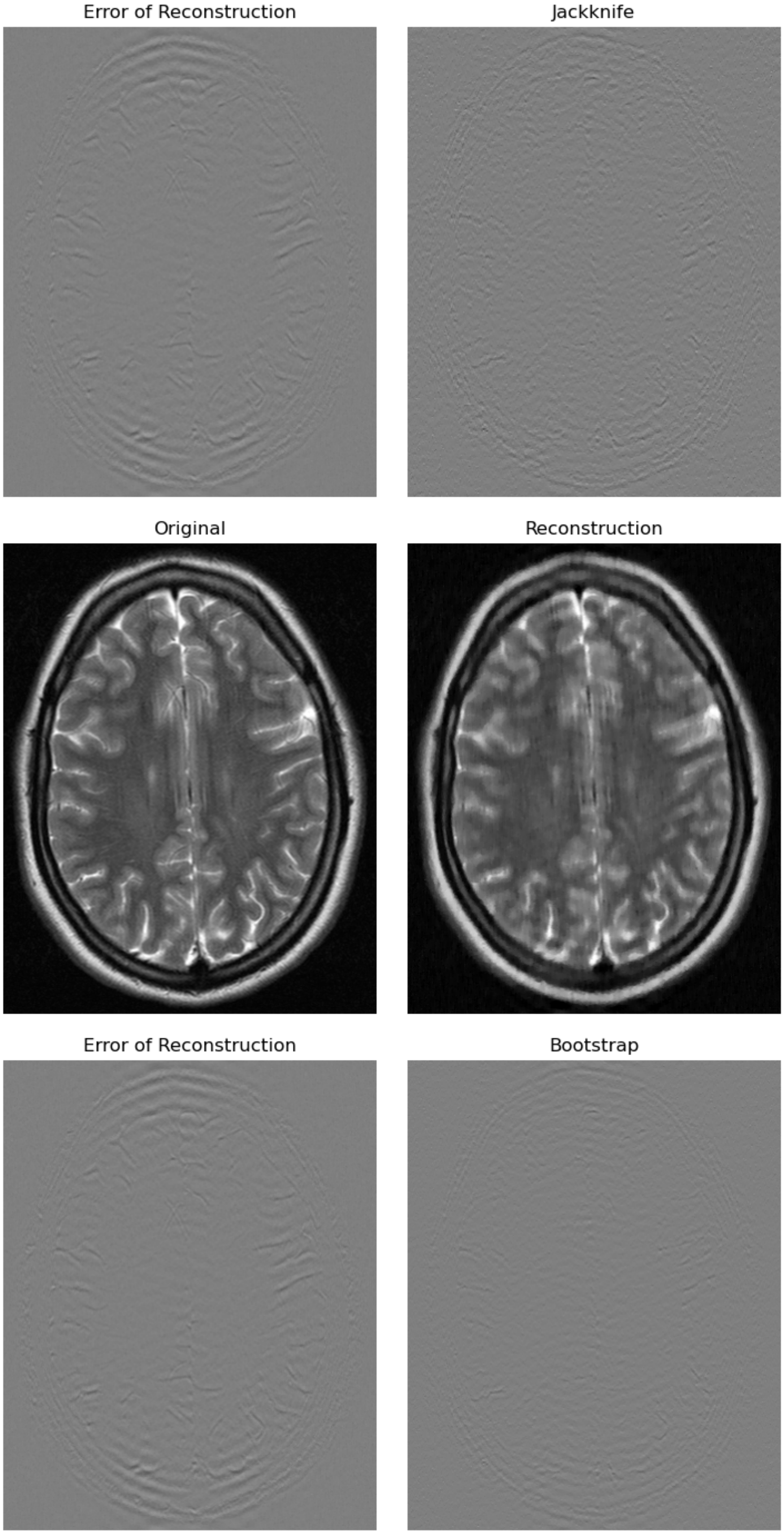

Figure 51: Horizontally retained sampling — slice 16

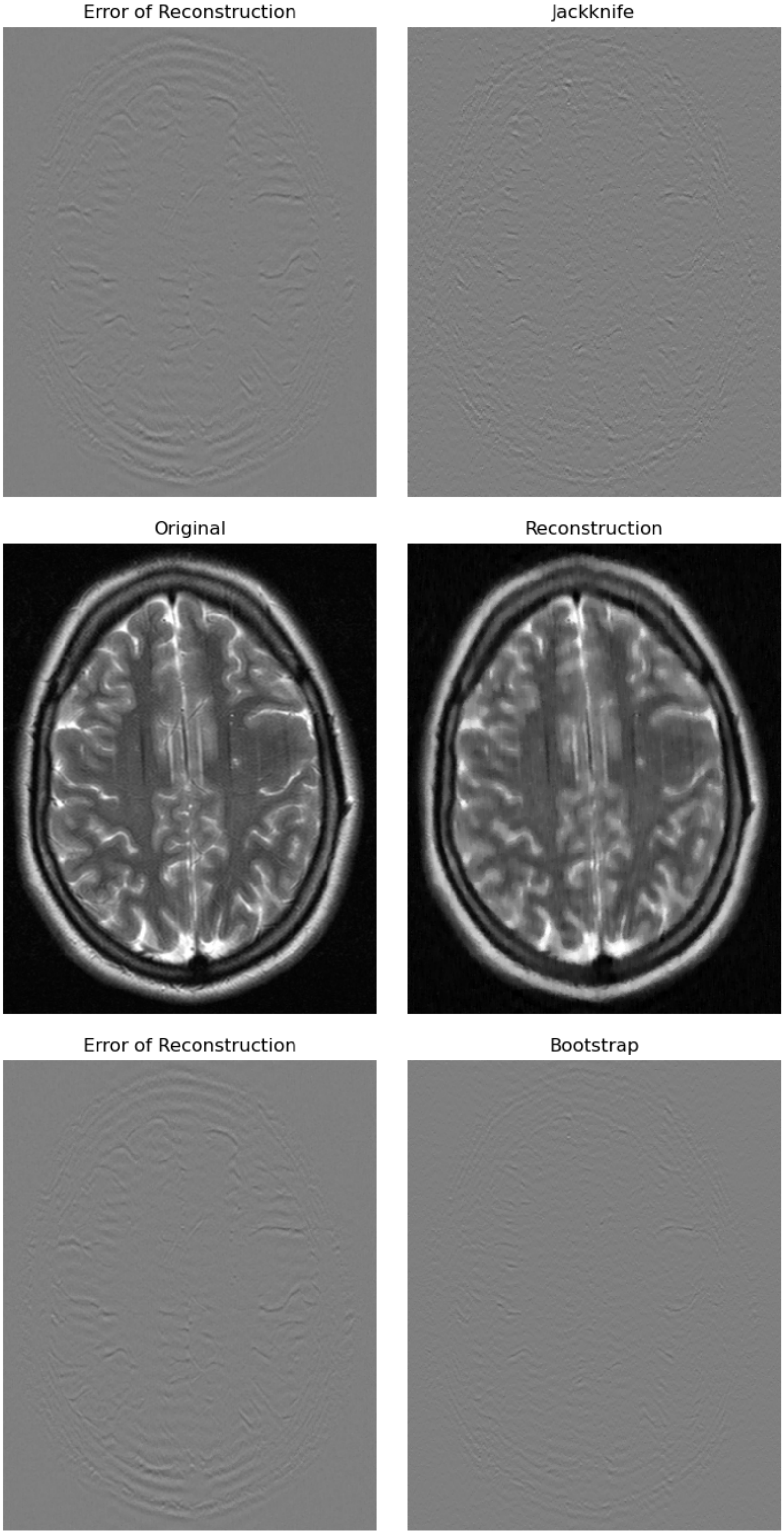

Figure 52: Horizontally retained sampling — slice 17

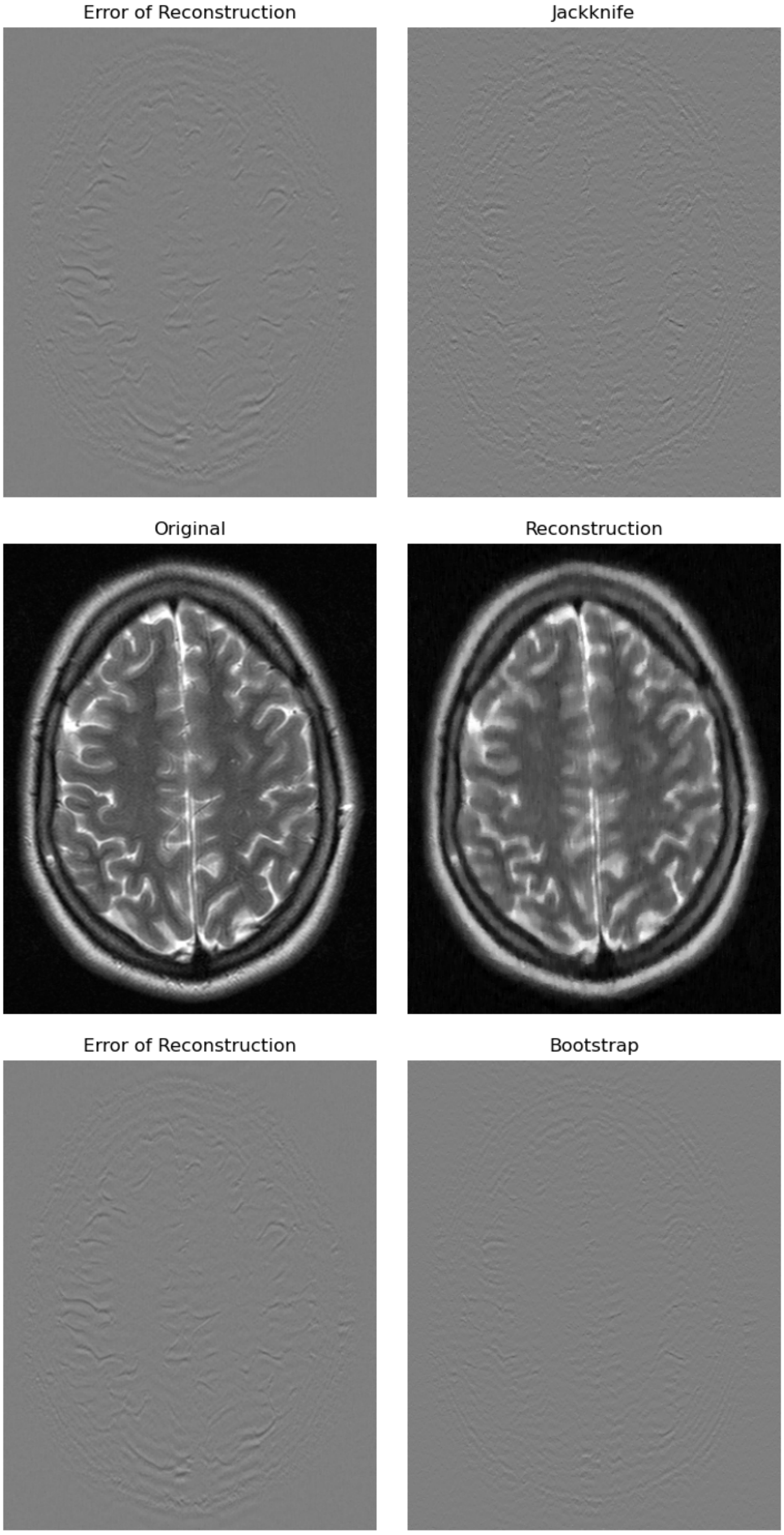

Figure 53: Horizontally retained sampling — slice 18

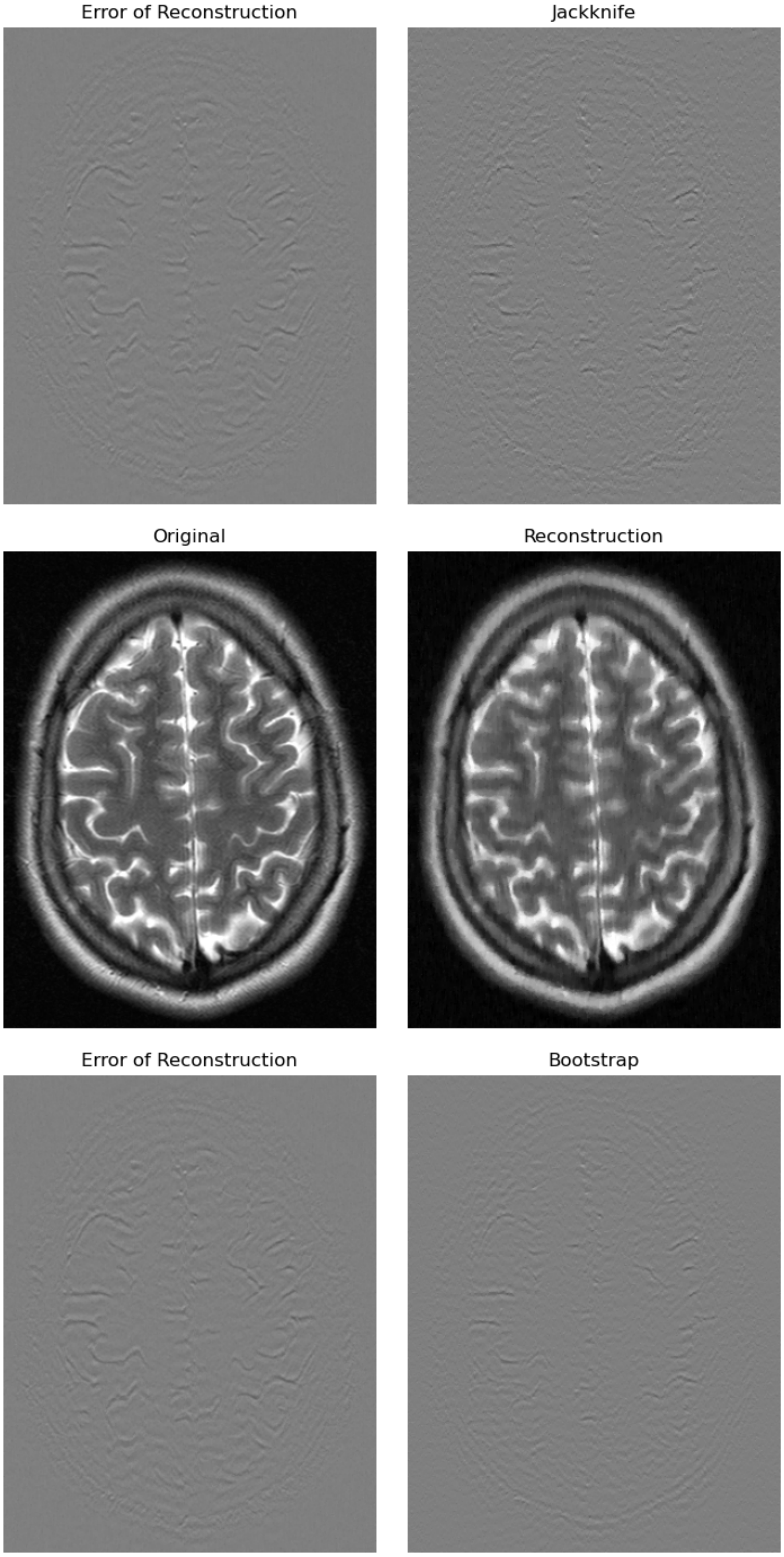

Figure 54: Horizontally retained sampling — slice 19

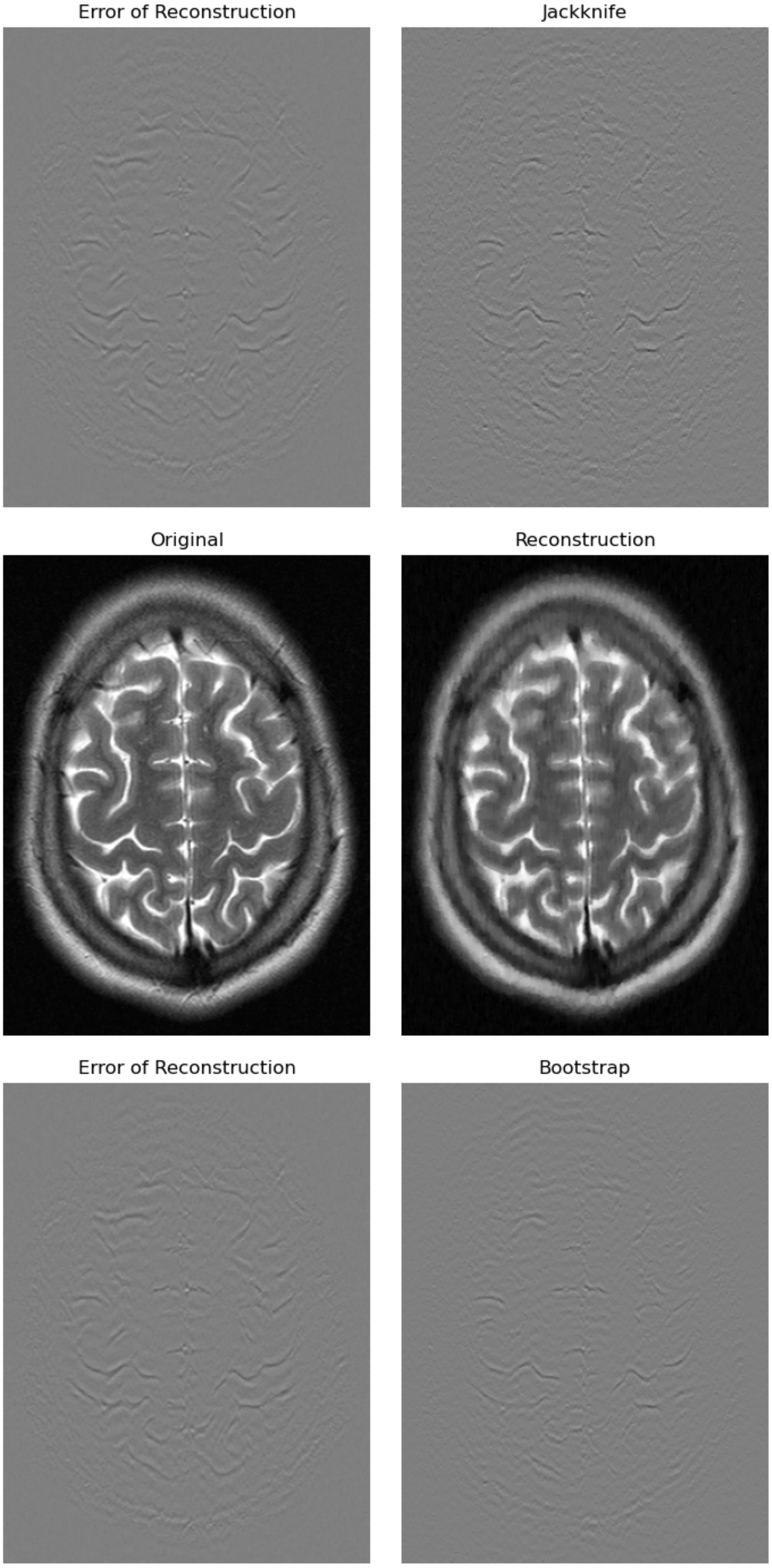

Figure 55: Horizontally retained sampling — slice 20

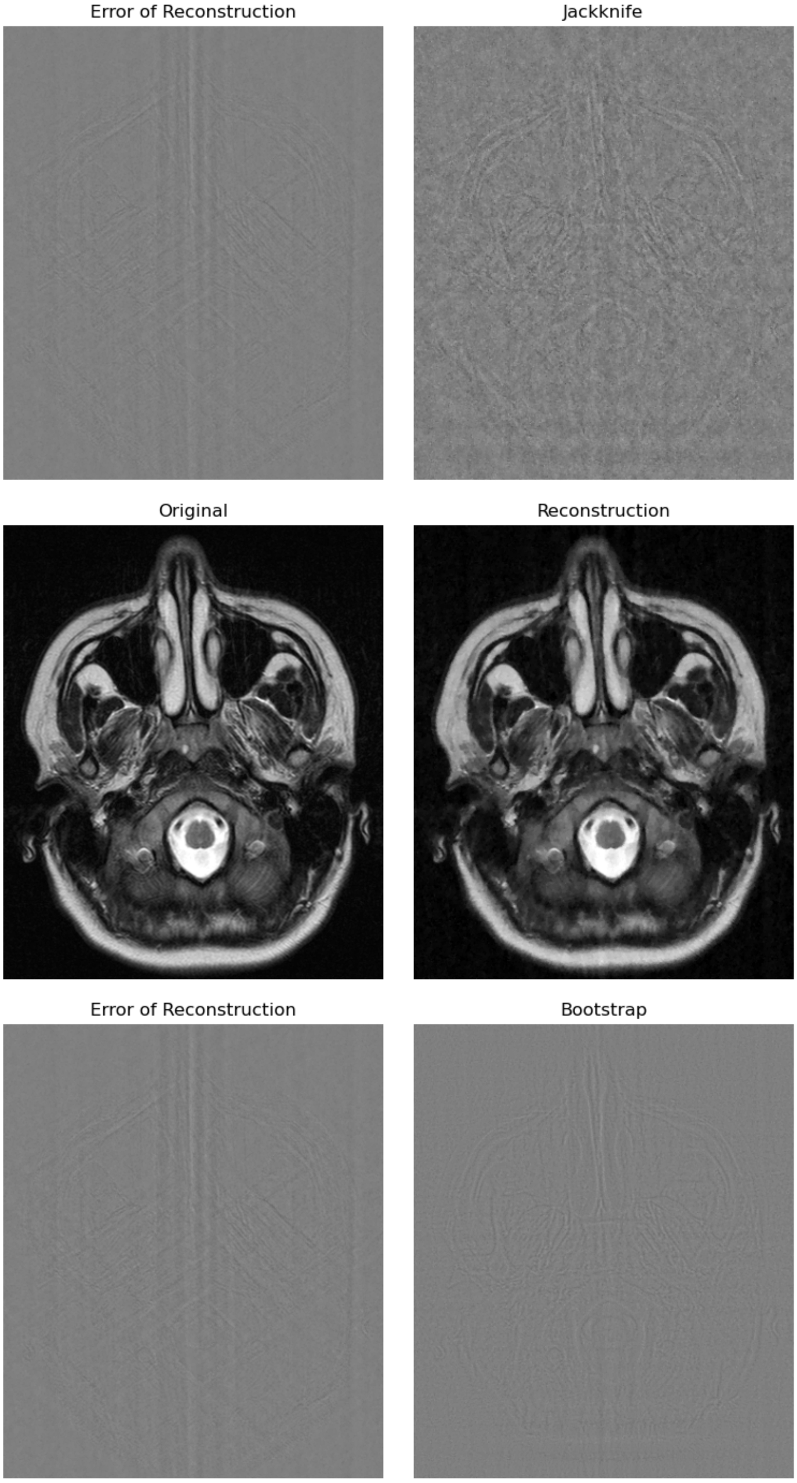

Figure 56: 2× radially retained sampling — slice 1

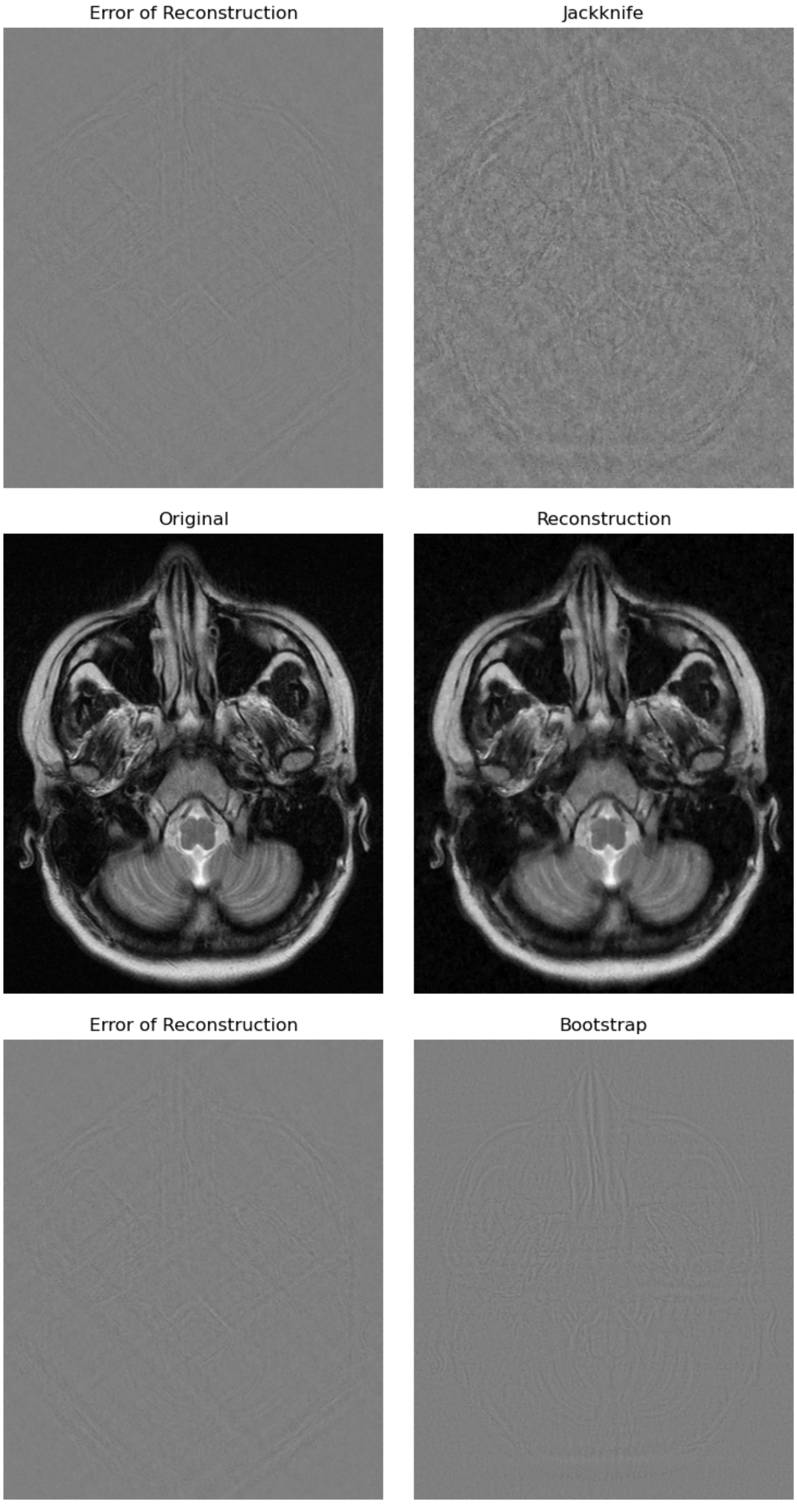

Figure 57: 2× radially retained sampling — slice 2

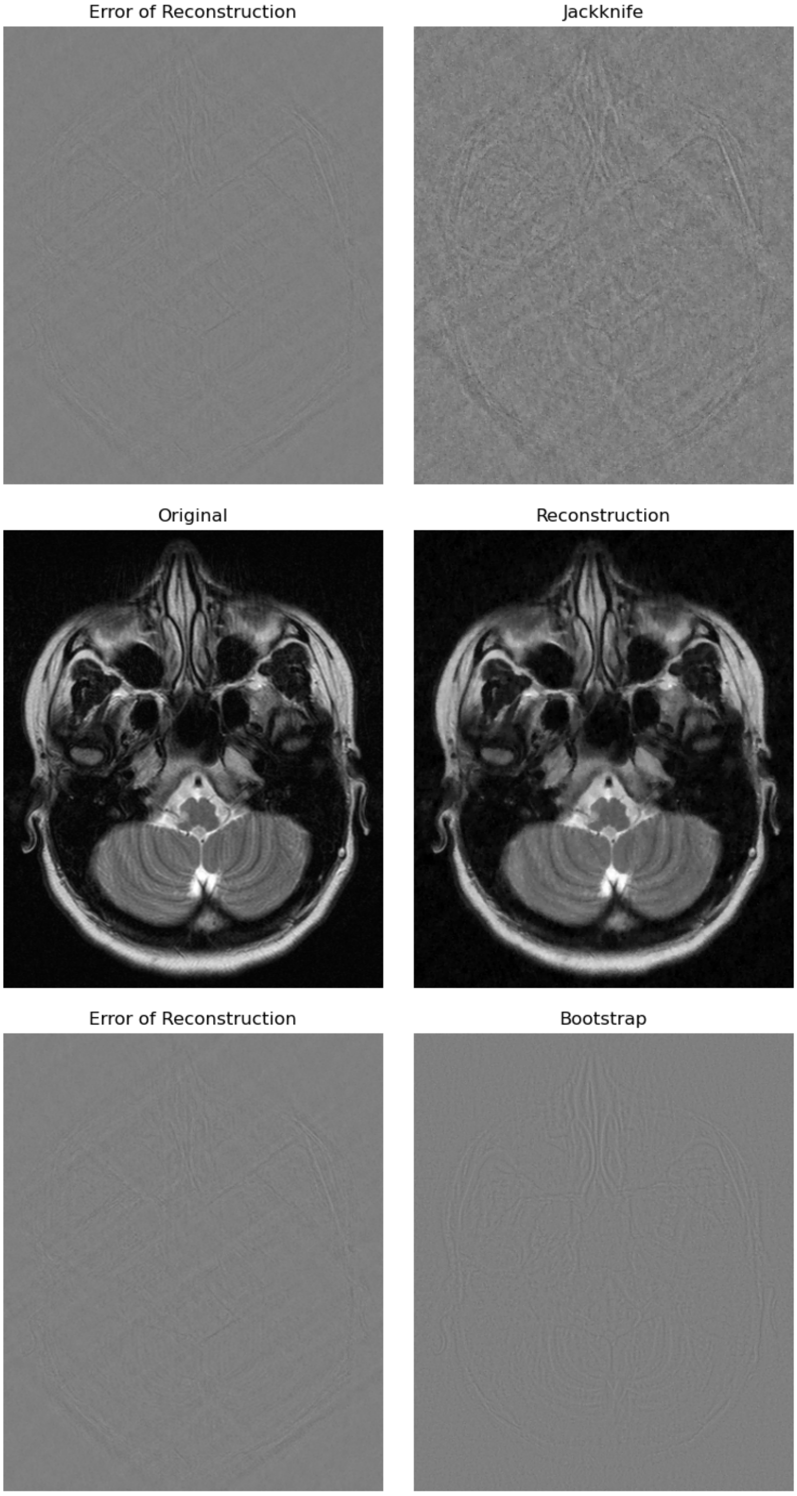

Figure 58: 2× radially retained sampling — slice 3

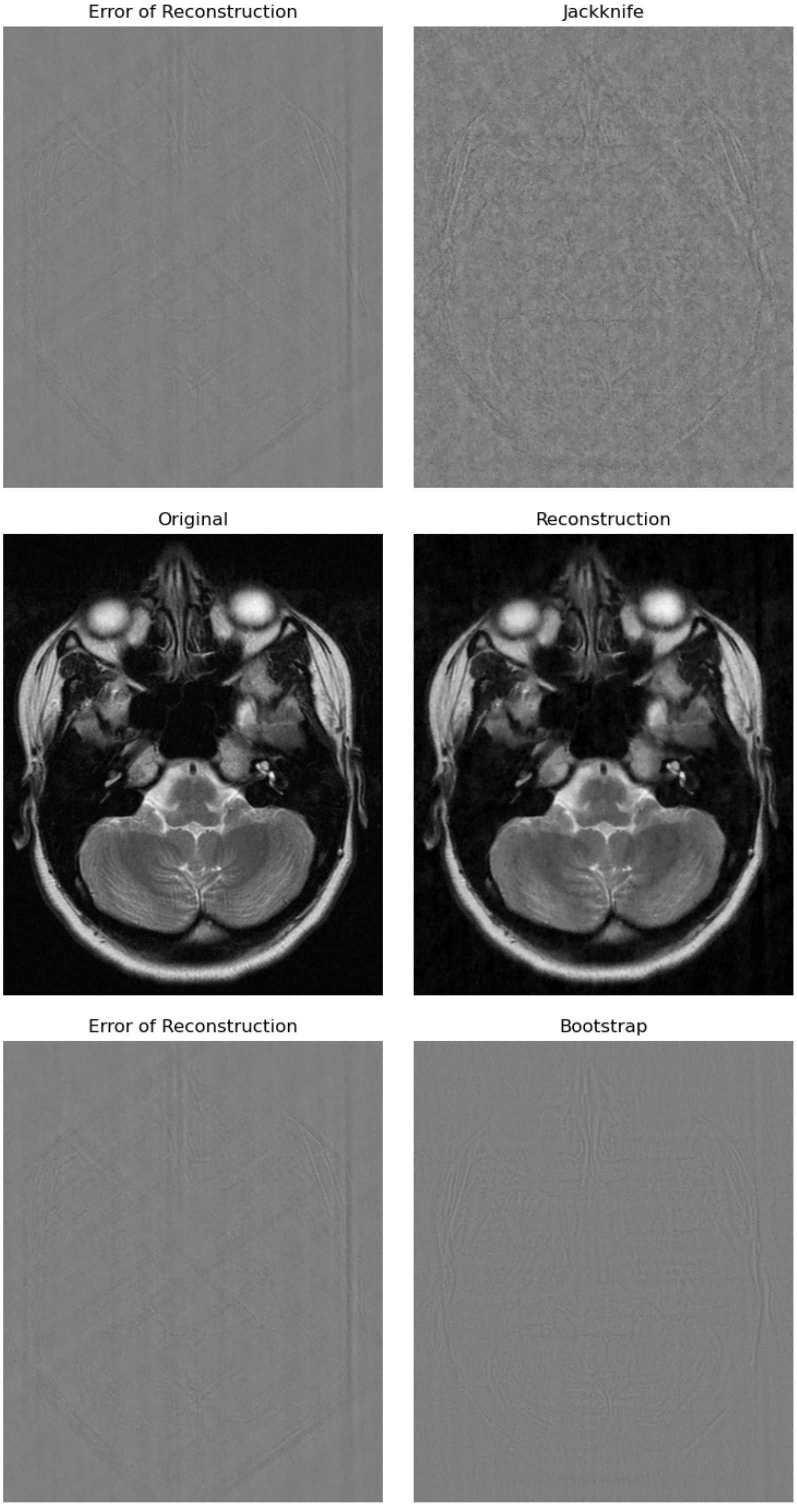

Figure 59: 2× radially retained sampling — slice 4

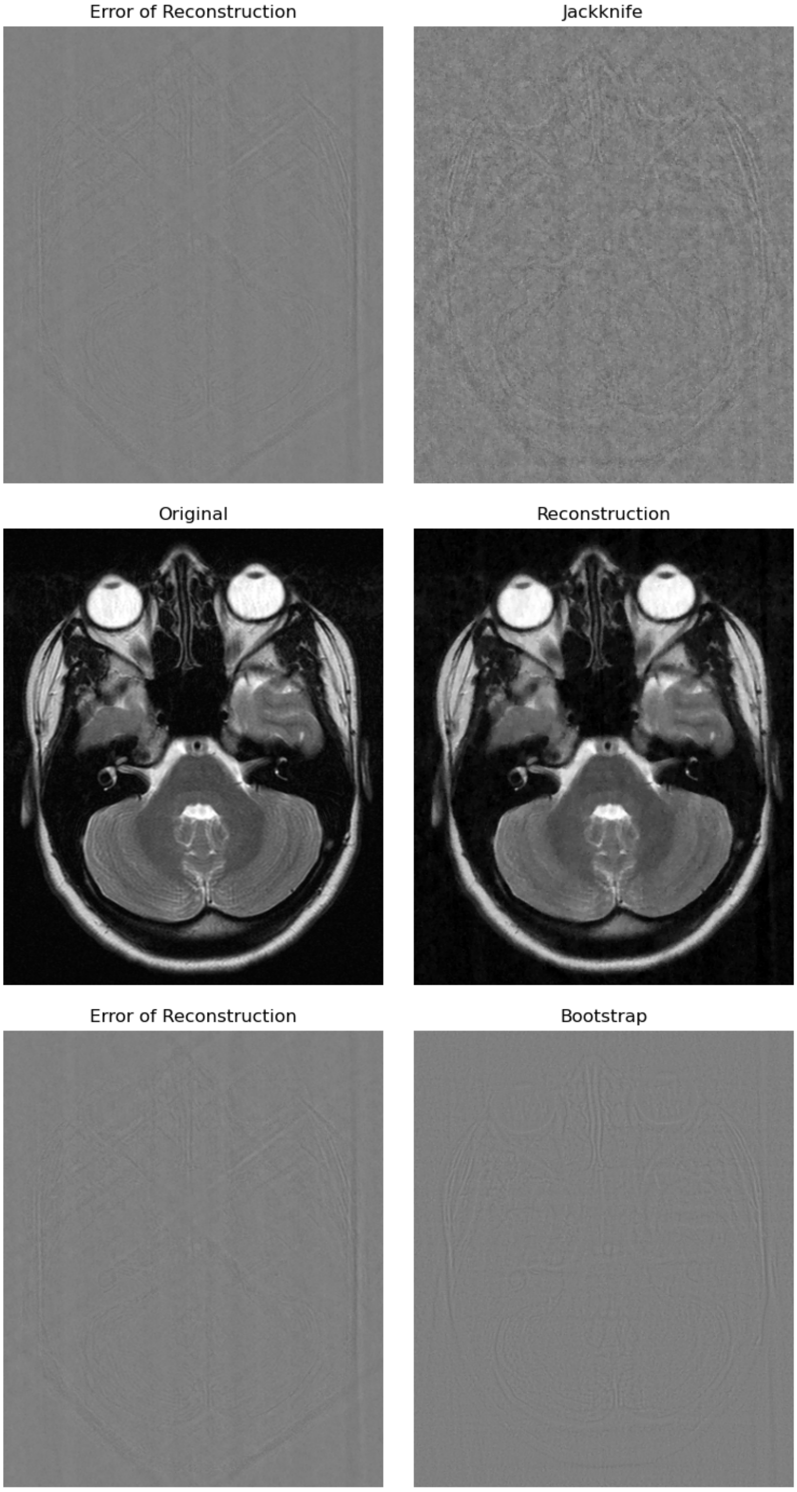

Figure 60: 2× radially retained sampling — slice 5

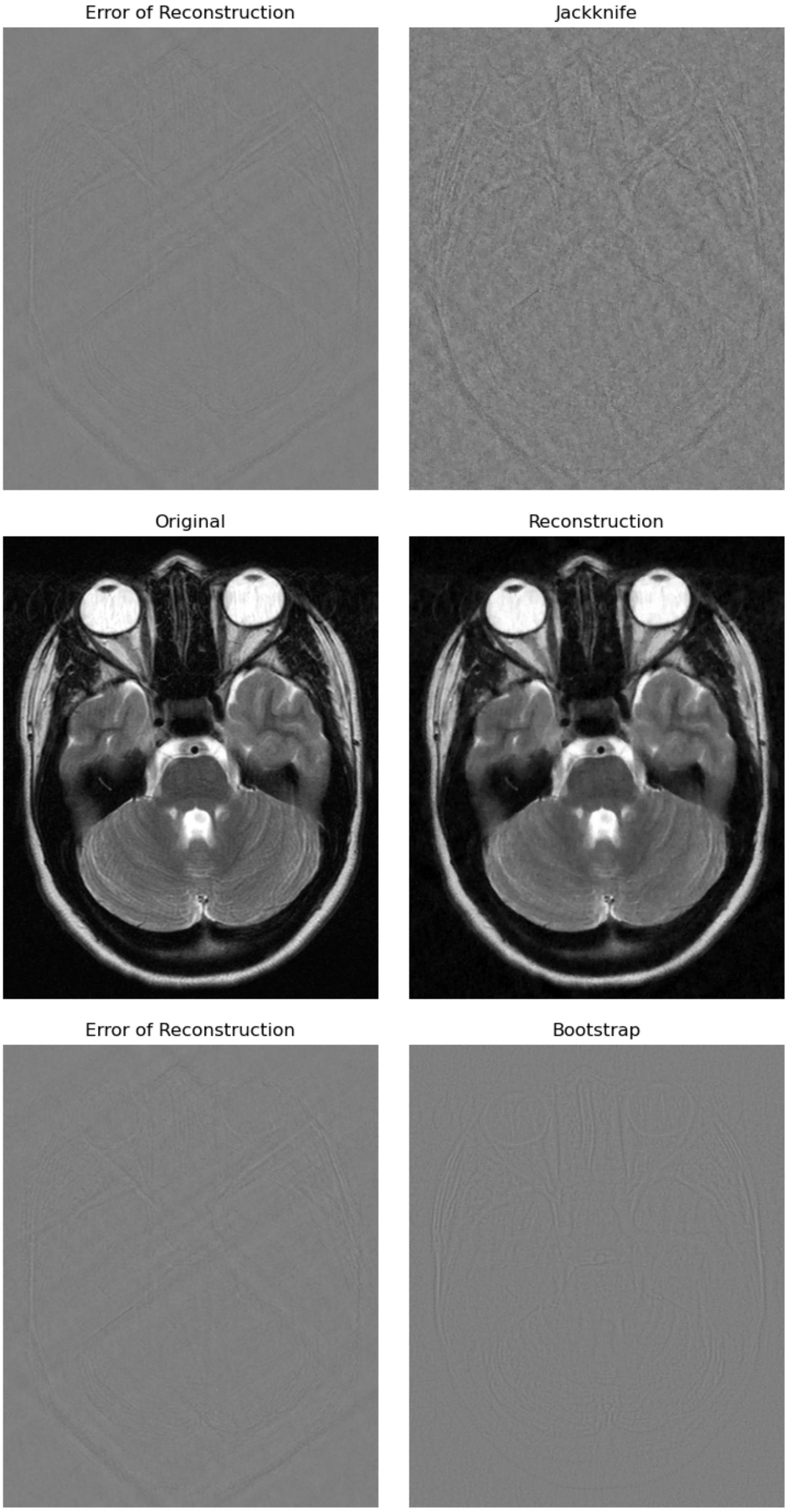

Figure 61: 2× radially retained sampling — slice 6

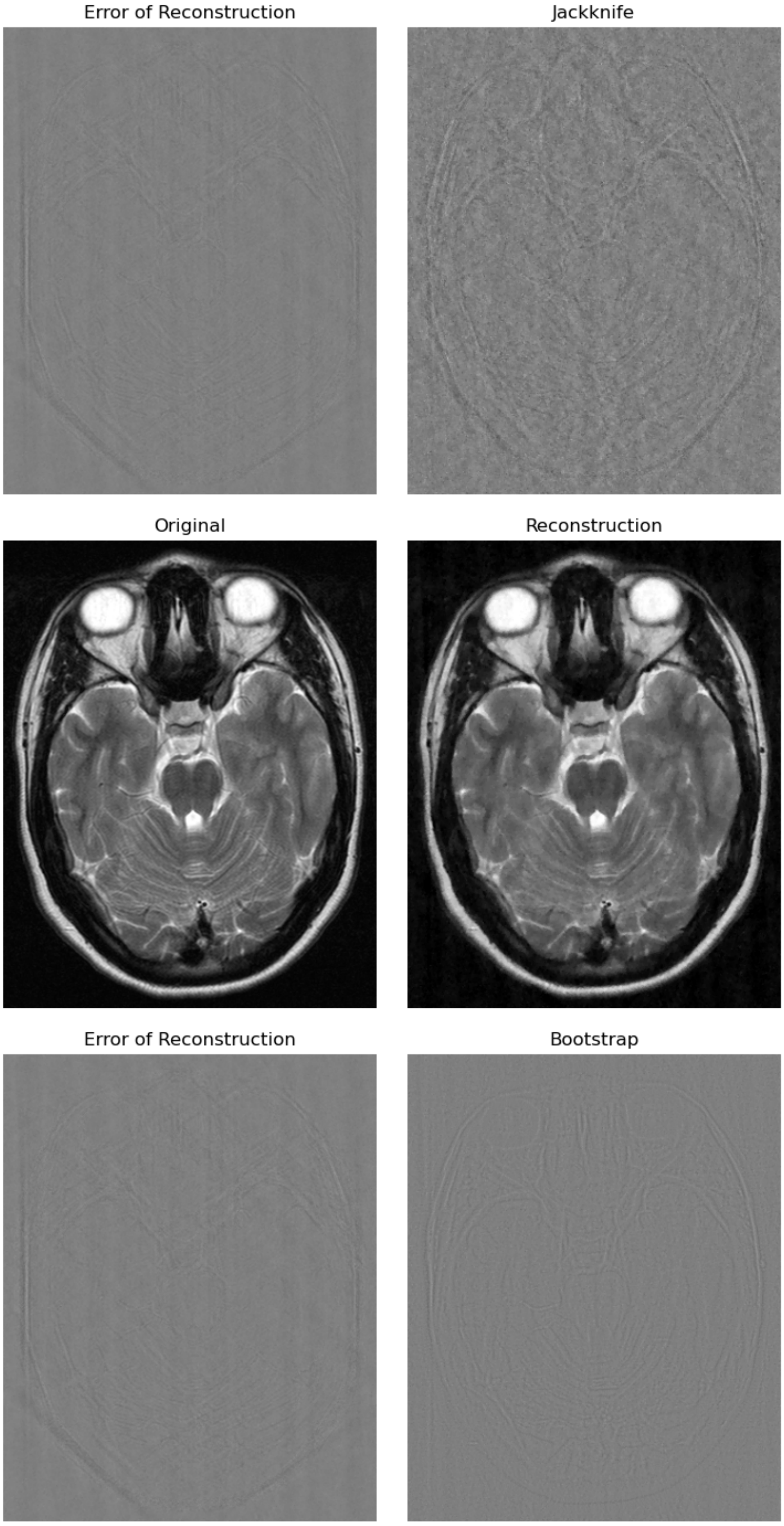

Figure 62: 2× radially retained sampling — slice 7

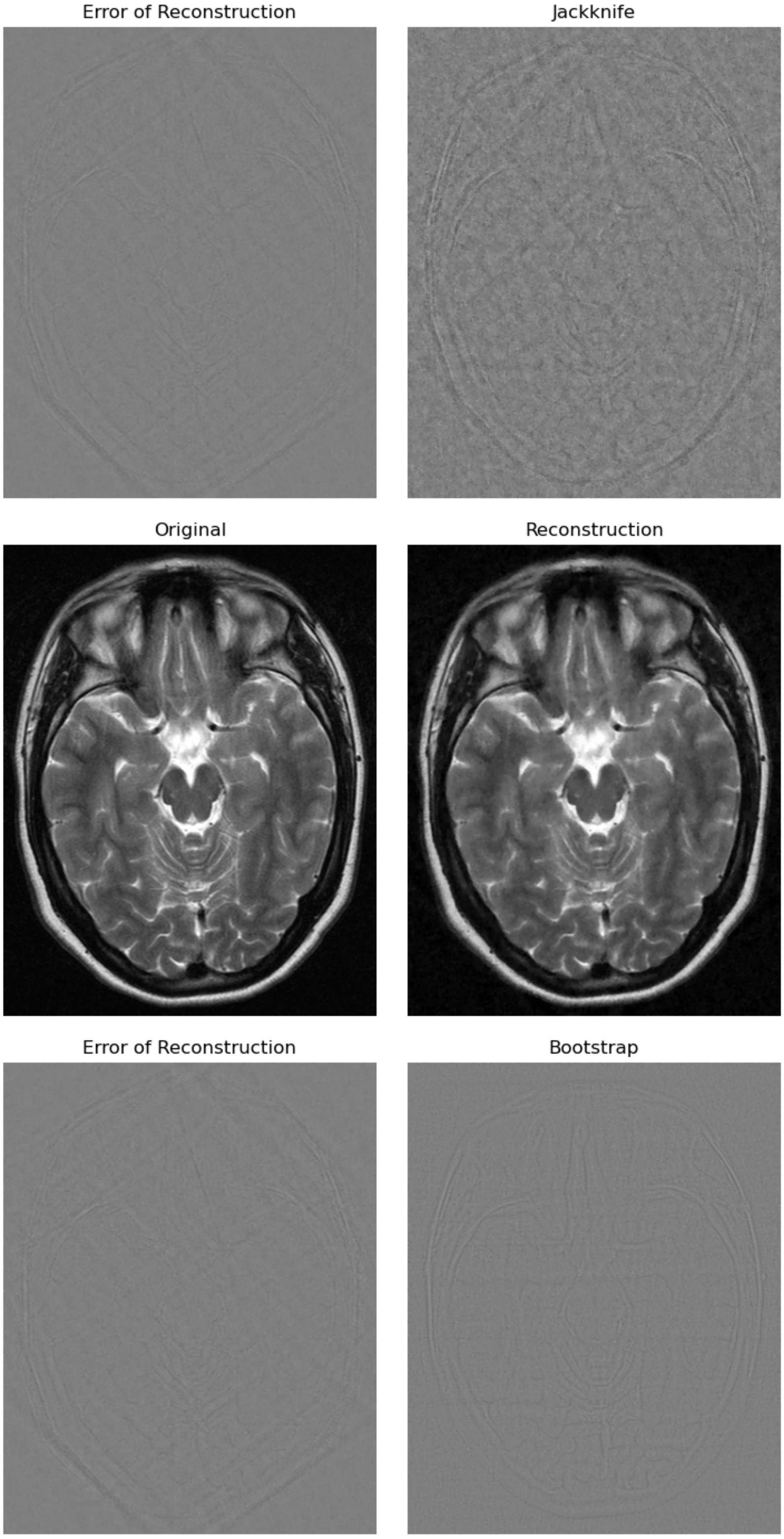

Figure 63: 2× radially retained sampling — slice 8

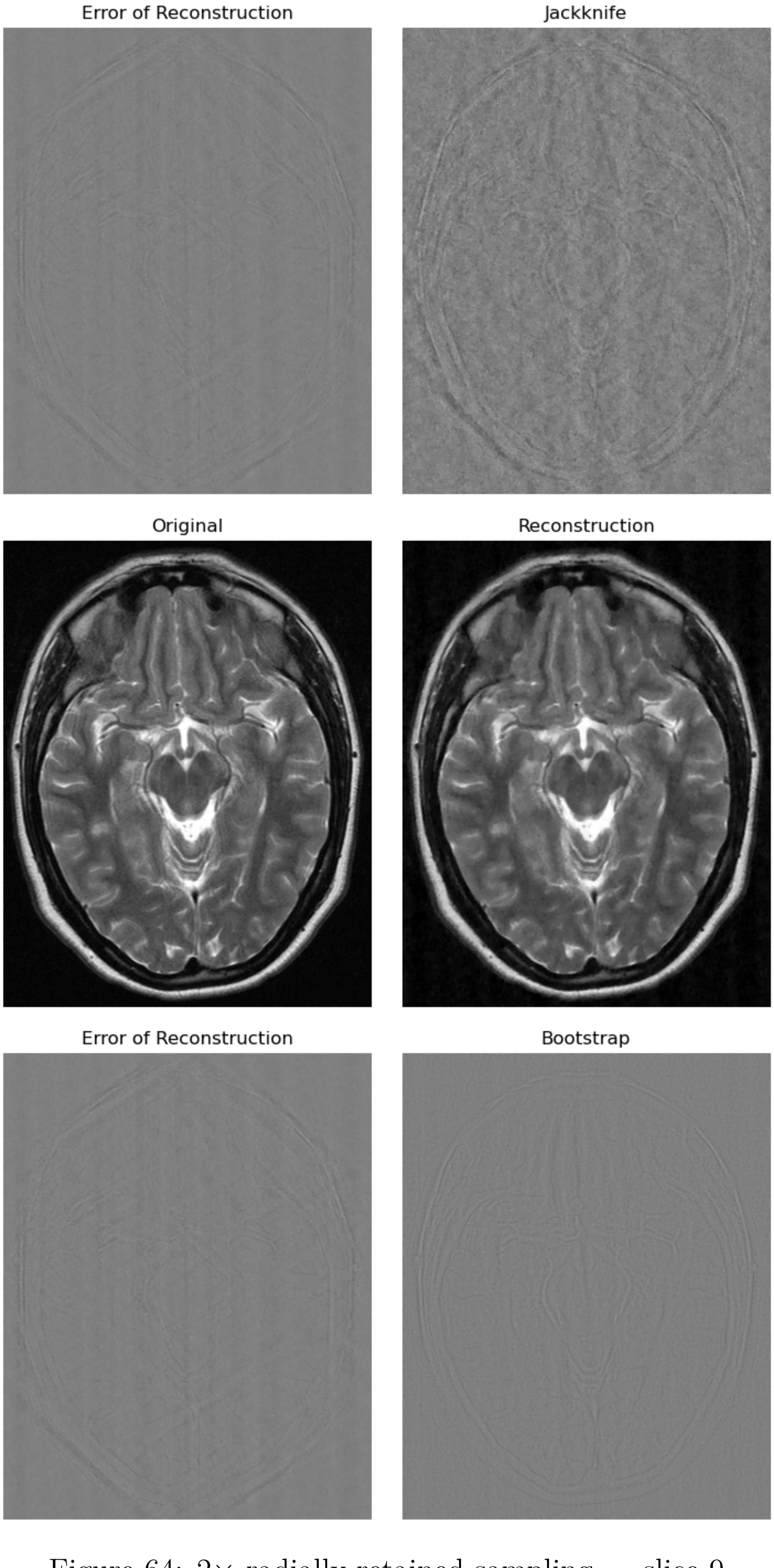

Figure 64: 2× radially retained sampling — slice 9

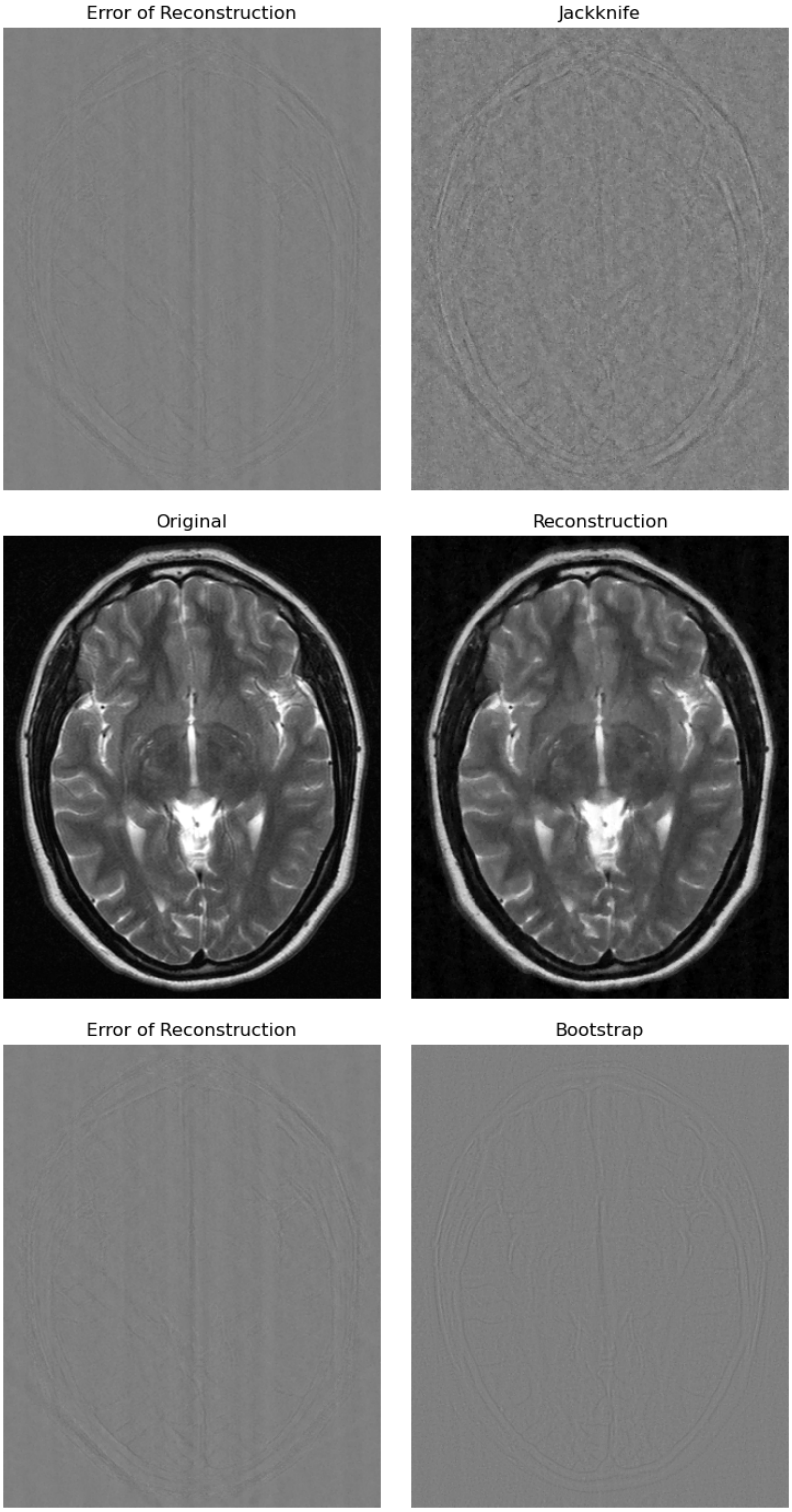

Figure 65: 2× radially retained sampling — slice 10

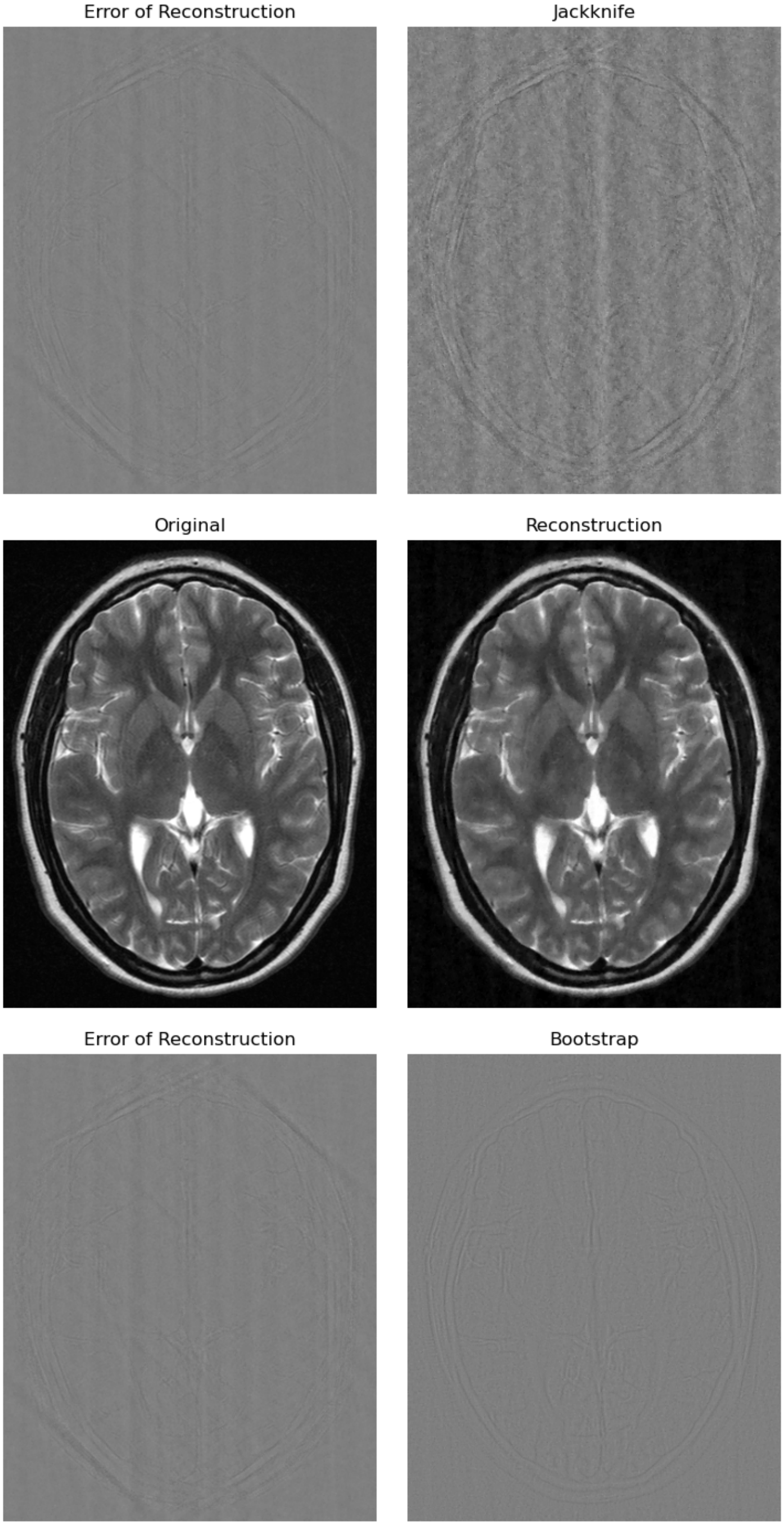

Figure 66: 2× radially retained sampling — slice 11

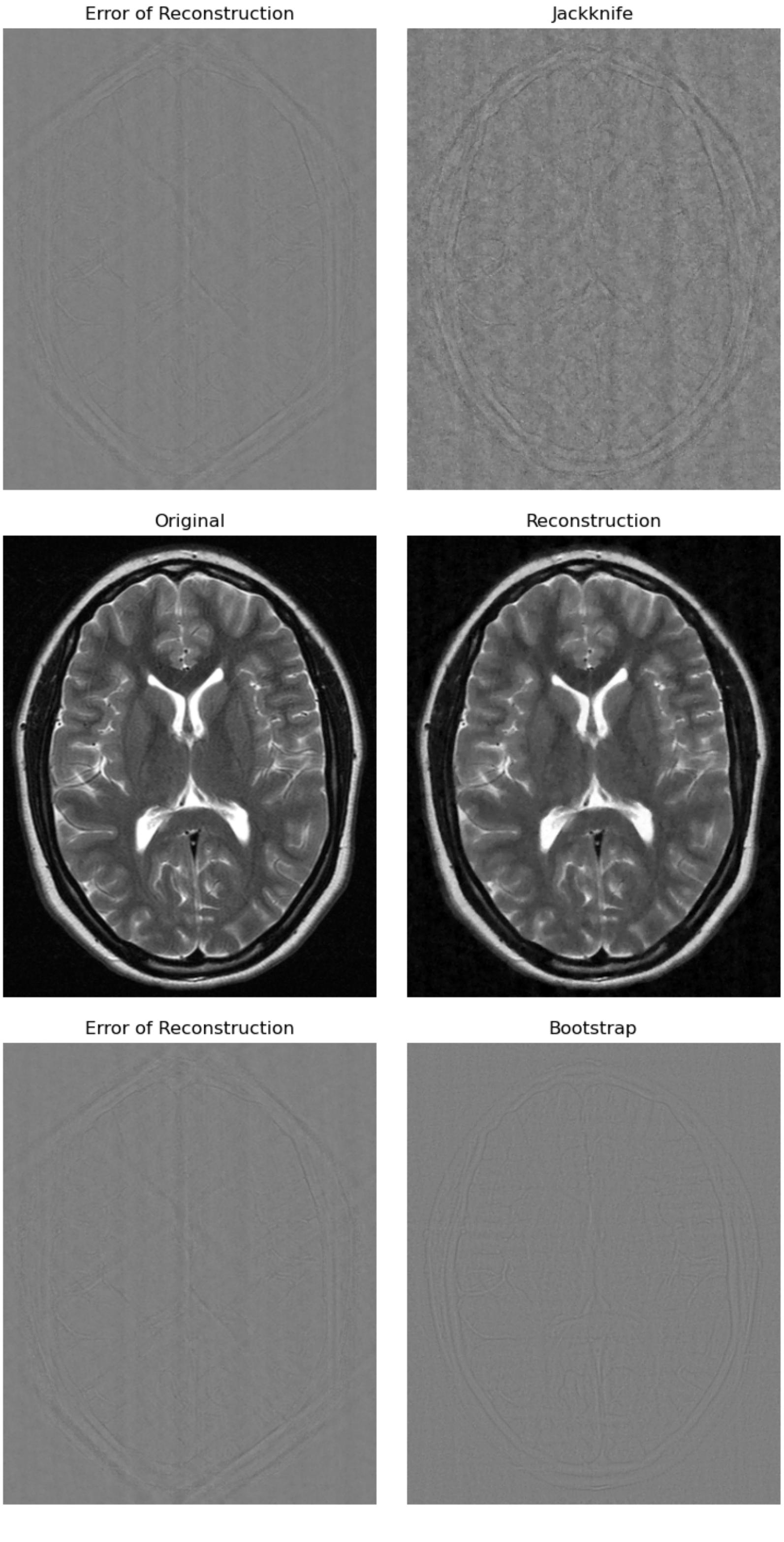

Figure 67: 2× radially retained sampling — slice 12

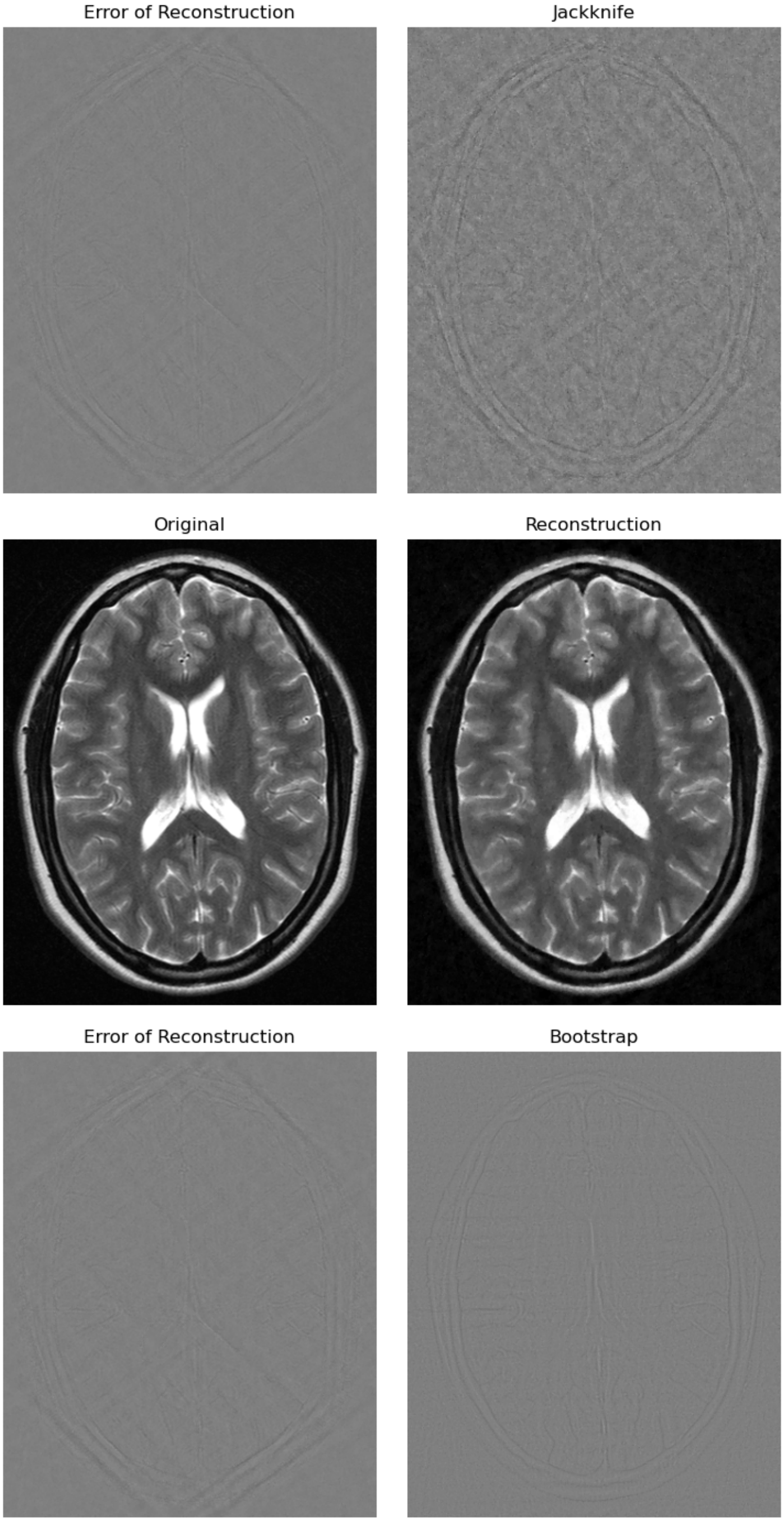

Figure 68: 2× radially retained sampling — slice 13

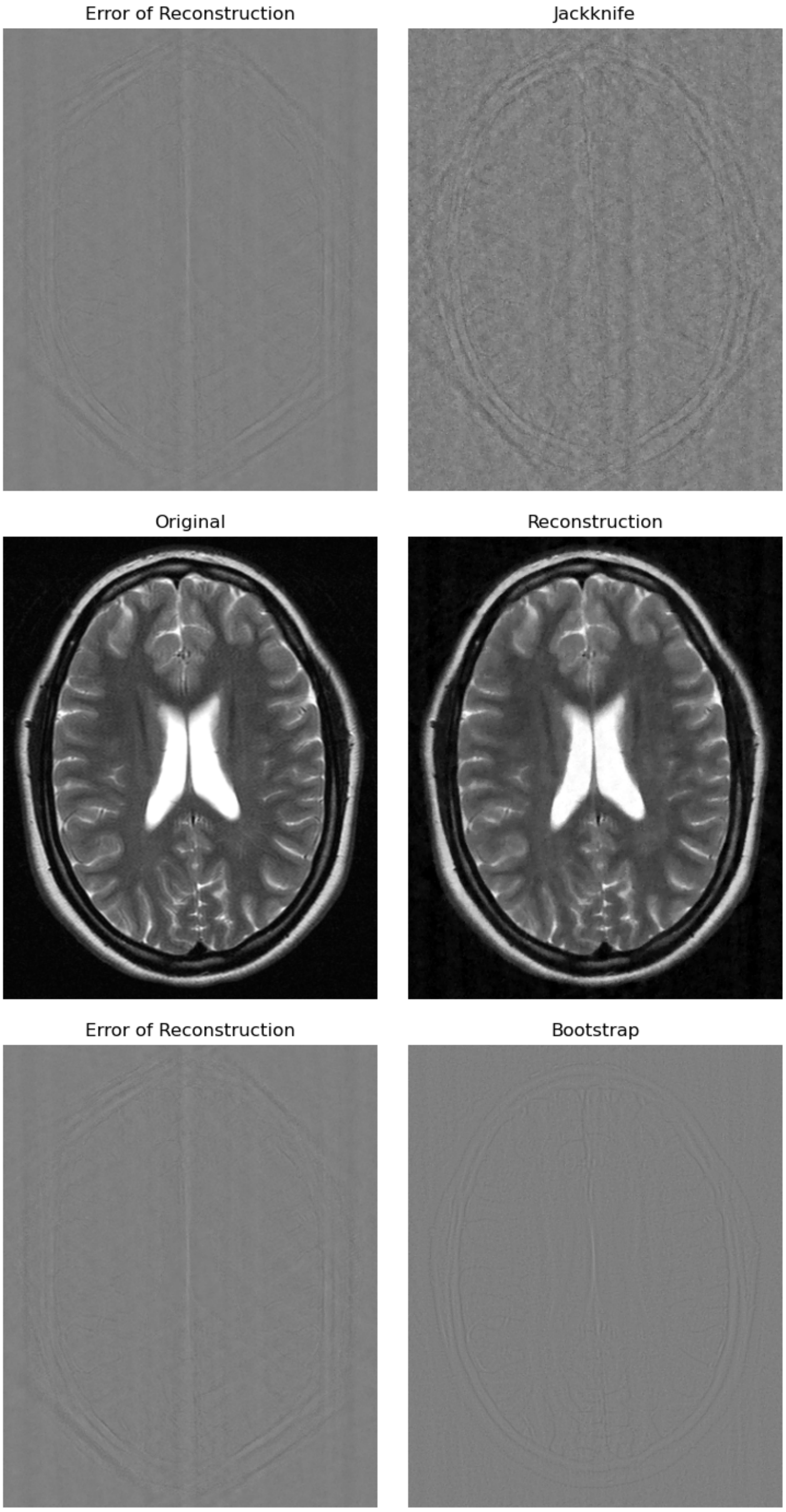

Figure 69: 2× radially retained sampling — slice 14

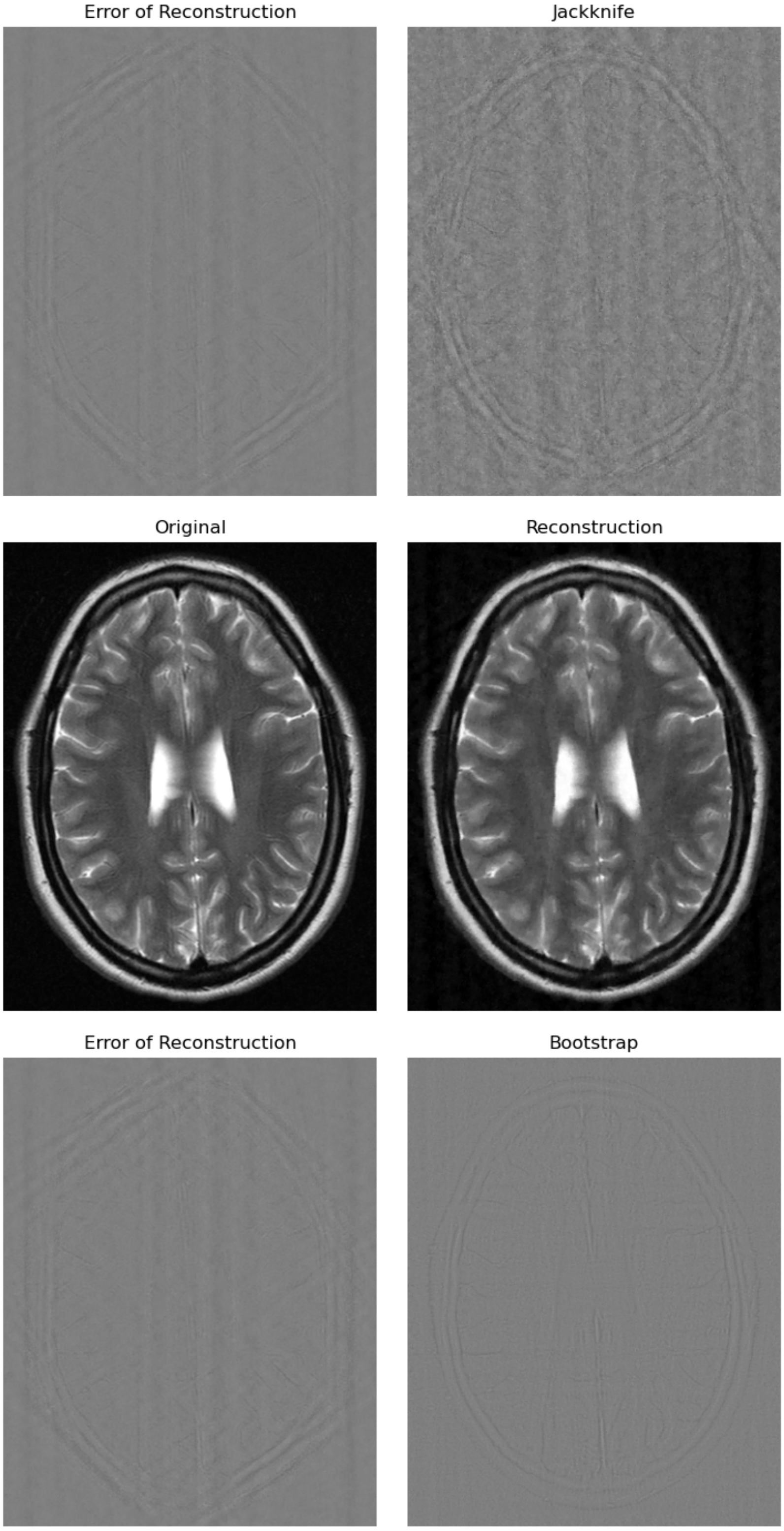

Figure 70: 2× radially retained sampling — slice 15

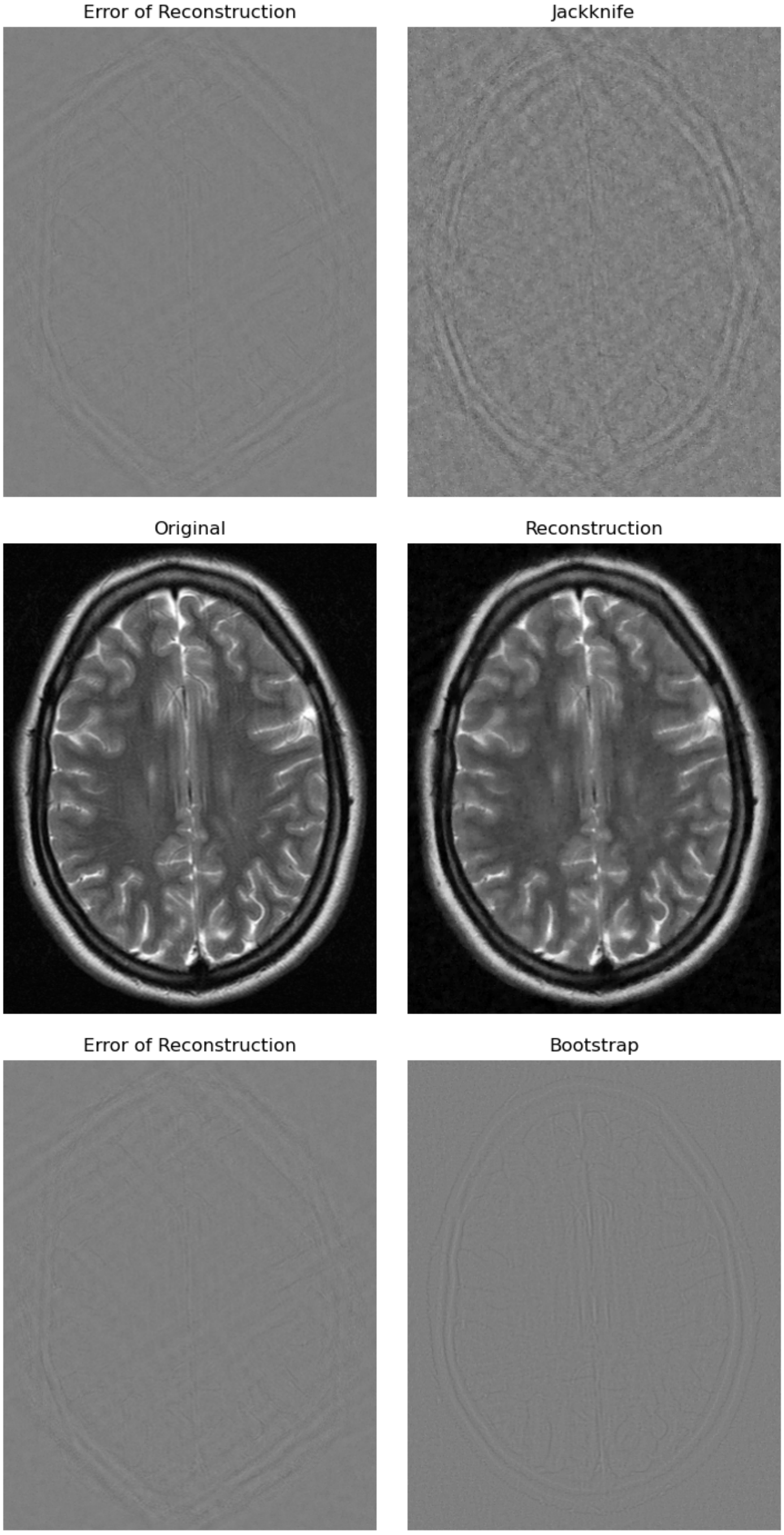

Figure 71: 2× radially retained sampling — slice 16

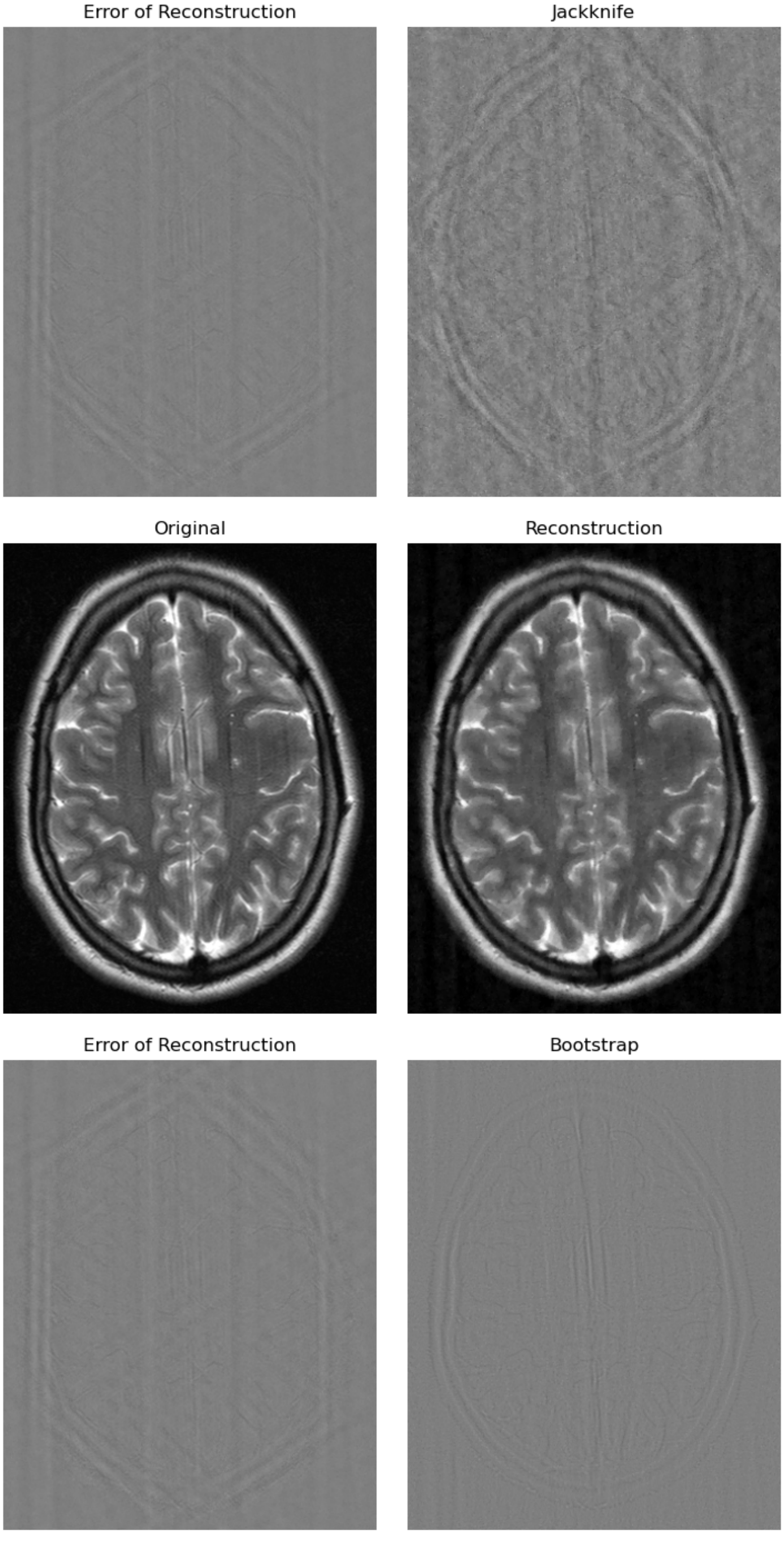

Figure 72: 2× radially retained sampling — slice 17

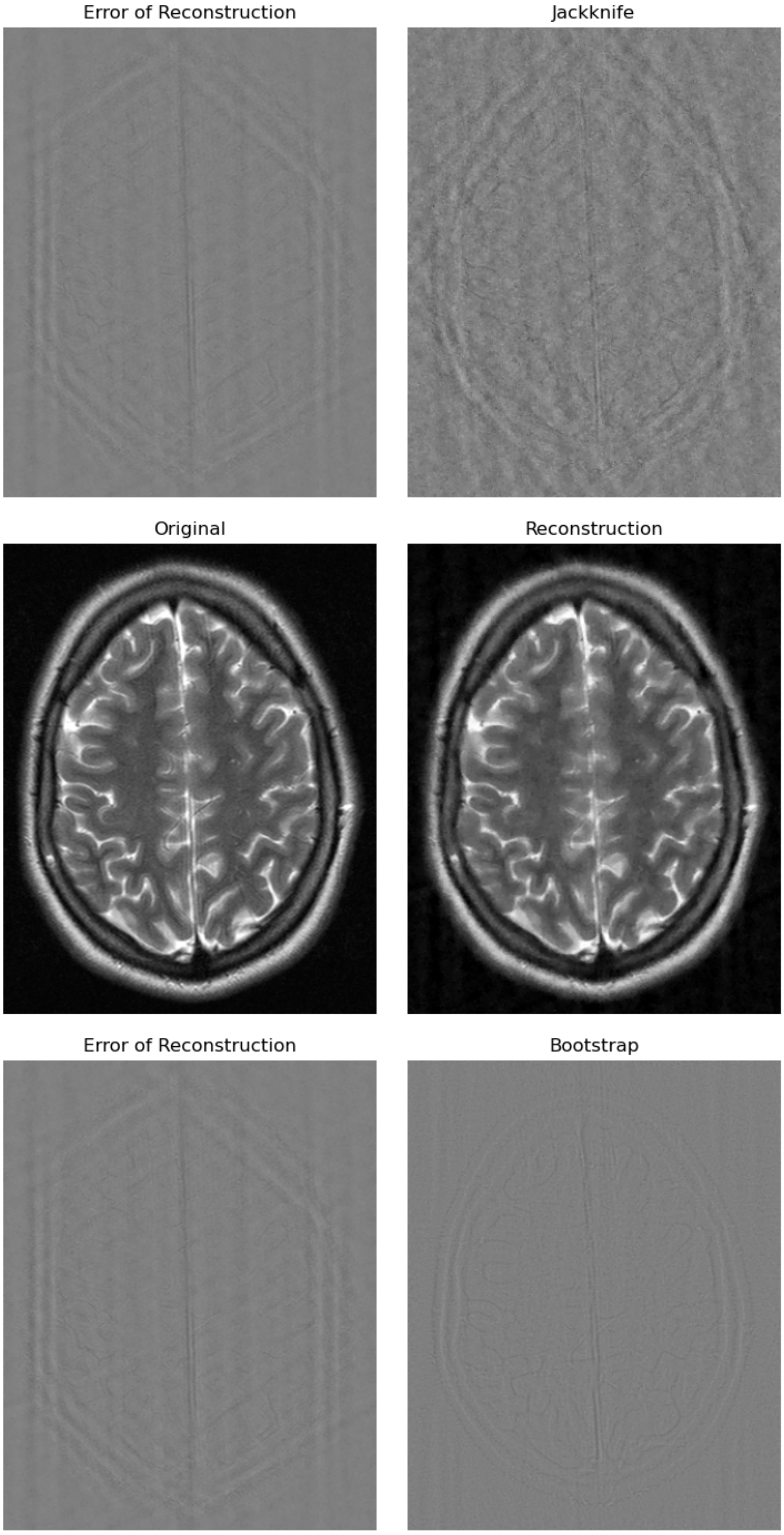

Figure 73: 2× radially retained sampling — slice 18

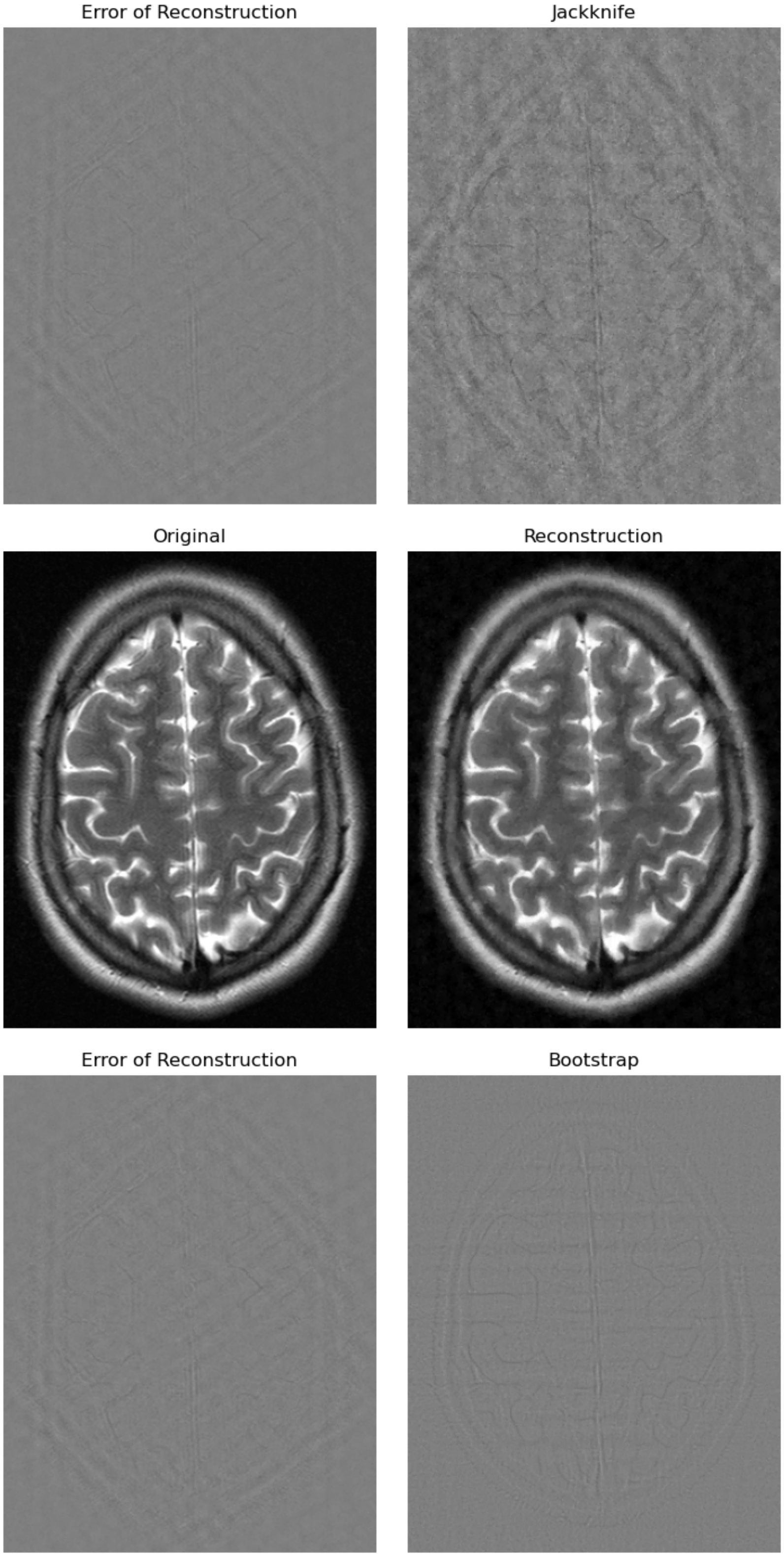

Figure 74: 2× radially retained sampling — slice 19

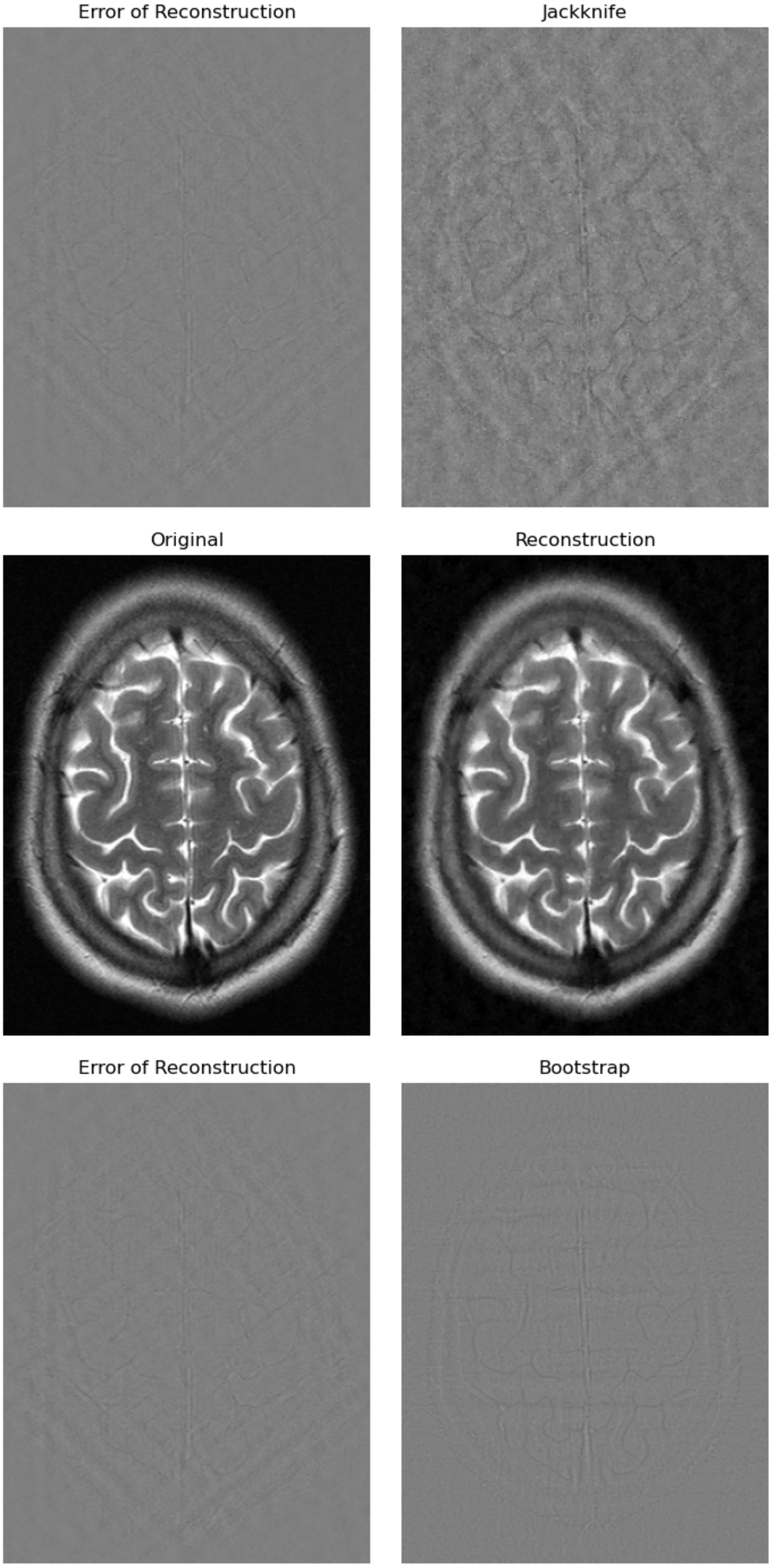

Figure 75: 2× radially retained sampling — slice 20

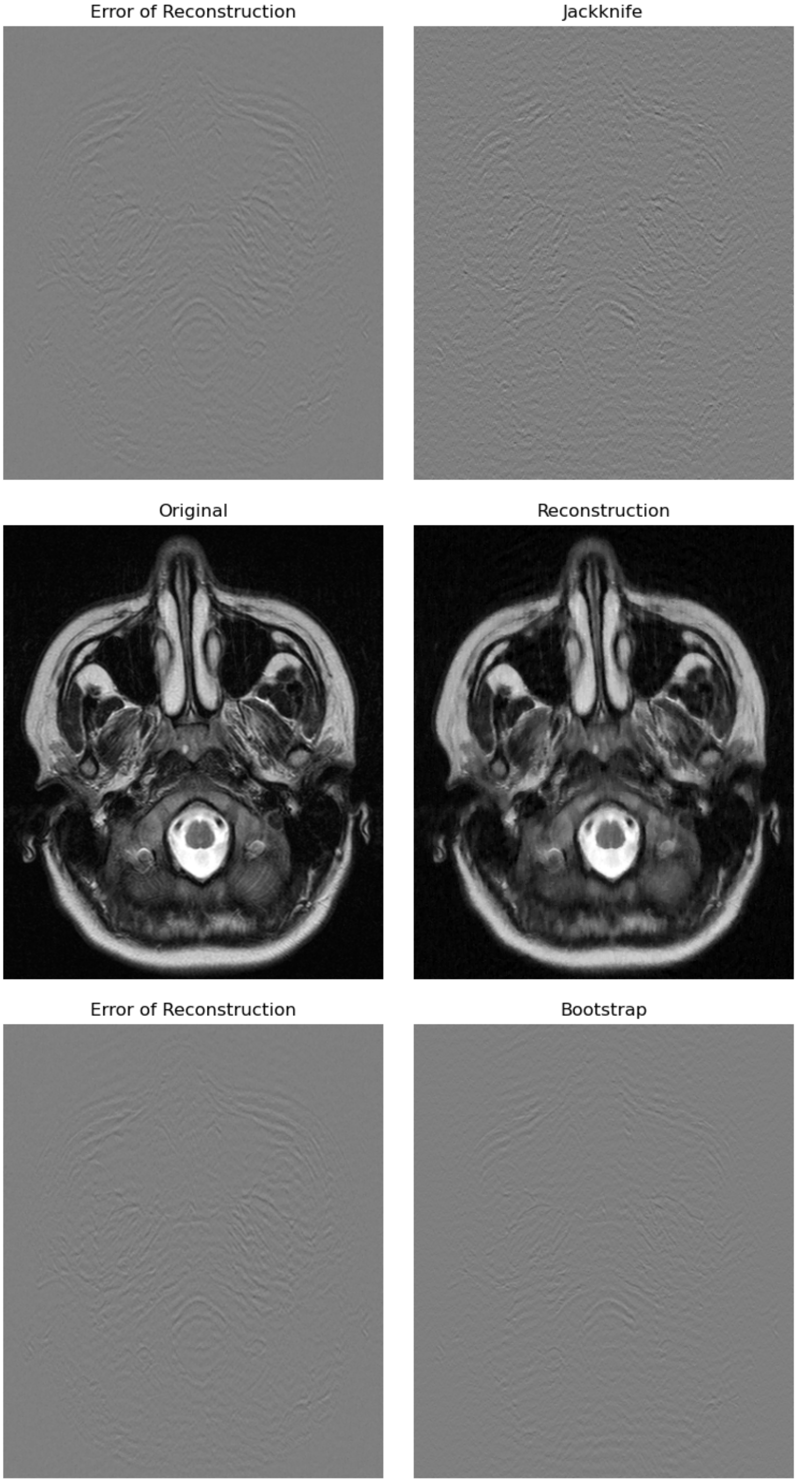

Figure 76: 2× horizontally retained sampling — slice 1

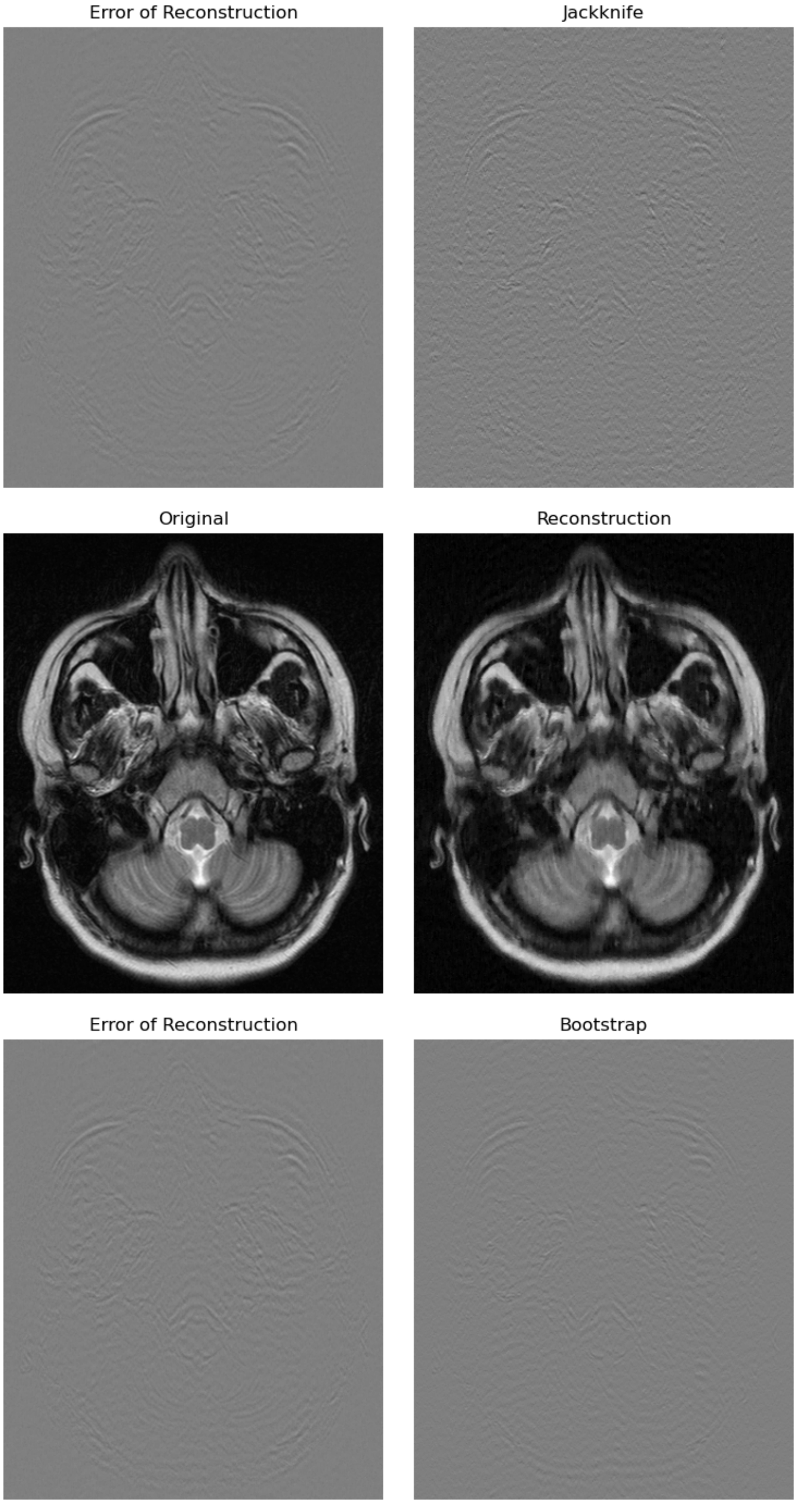

Figure 77: 2× horizontally retained sampling — slice 2

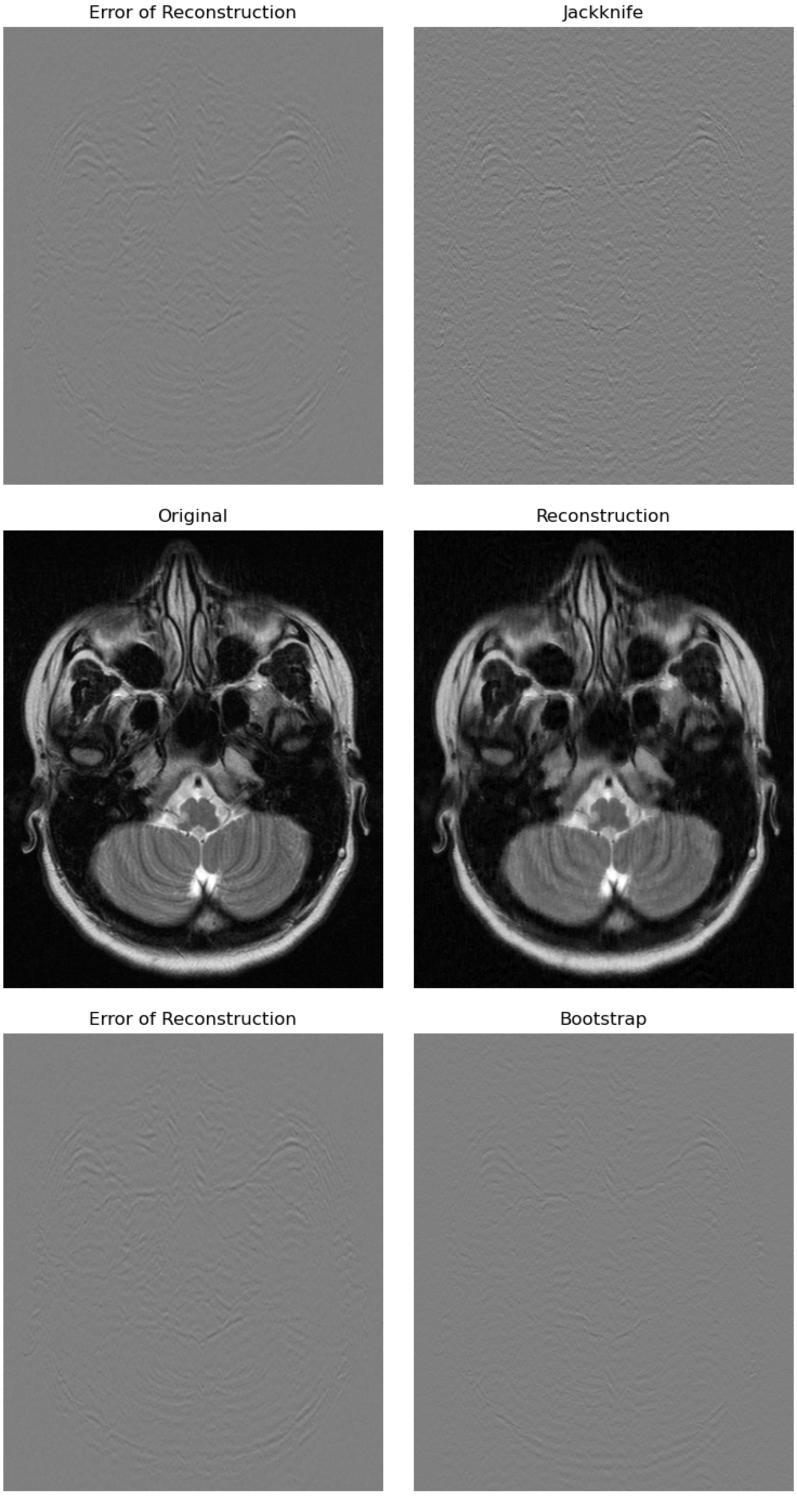

Figure 78: 2× horizontally retained sampling — slice 3

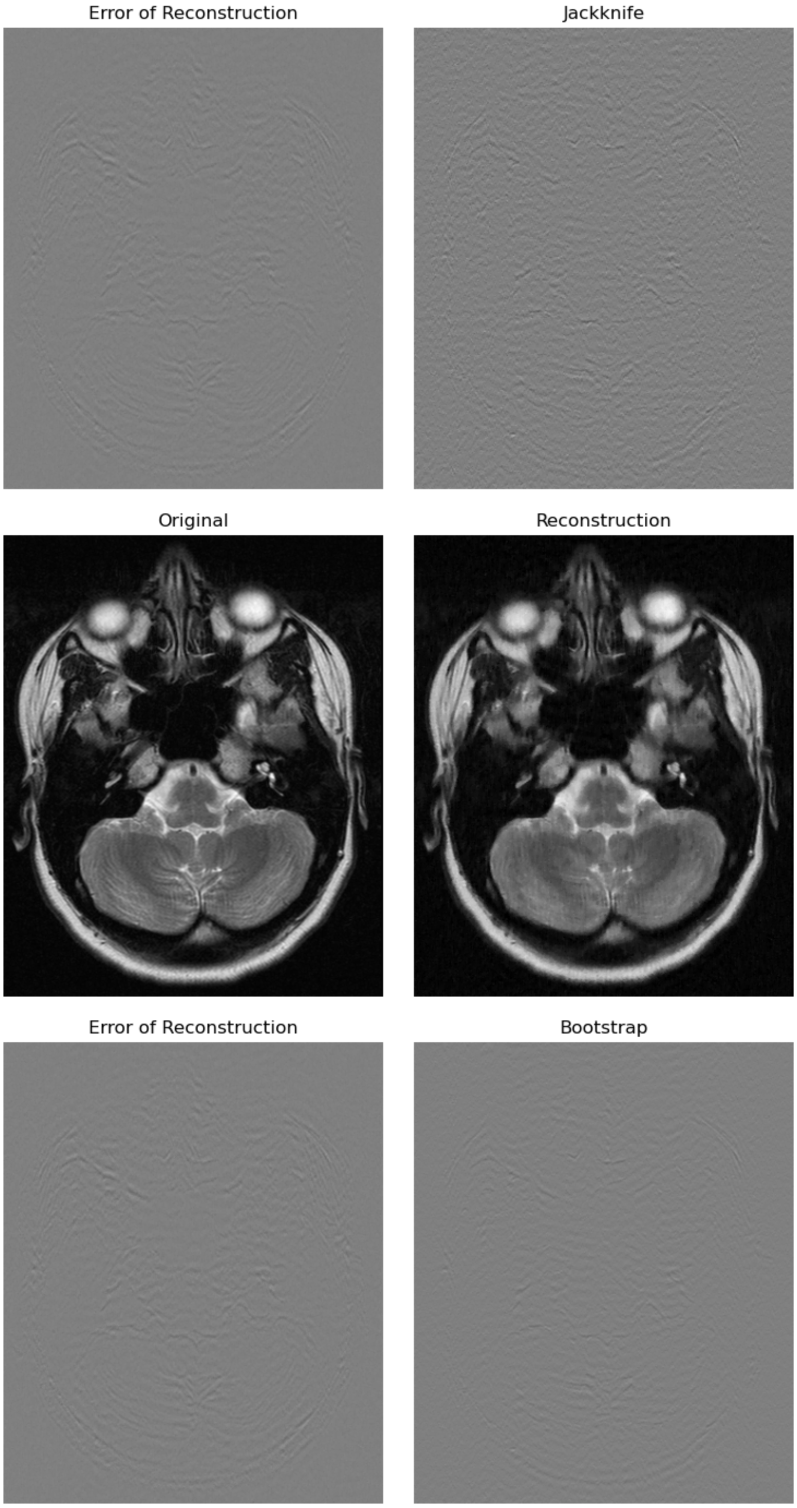

Figure 79: 2× horizontally retained sampling — slice 4

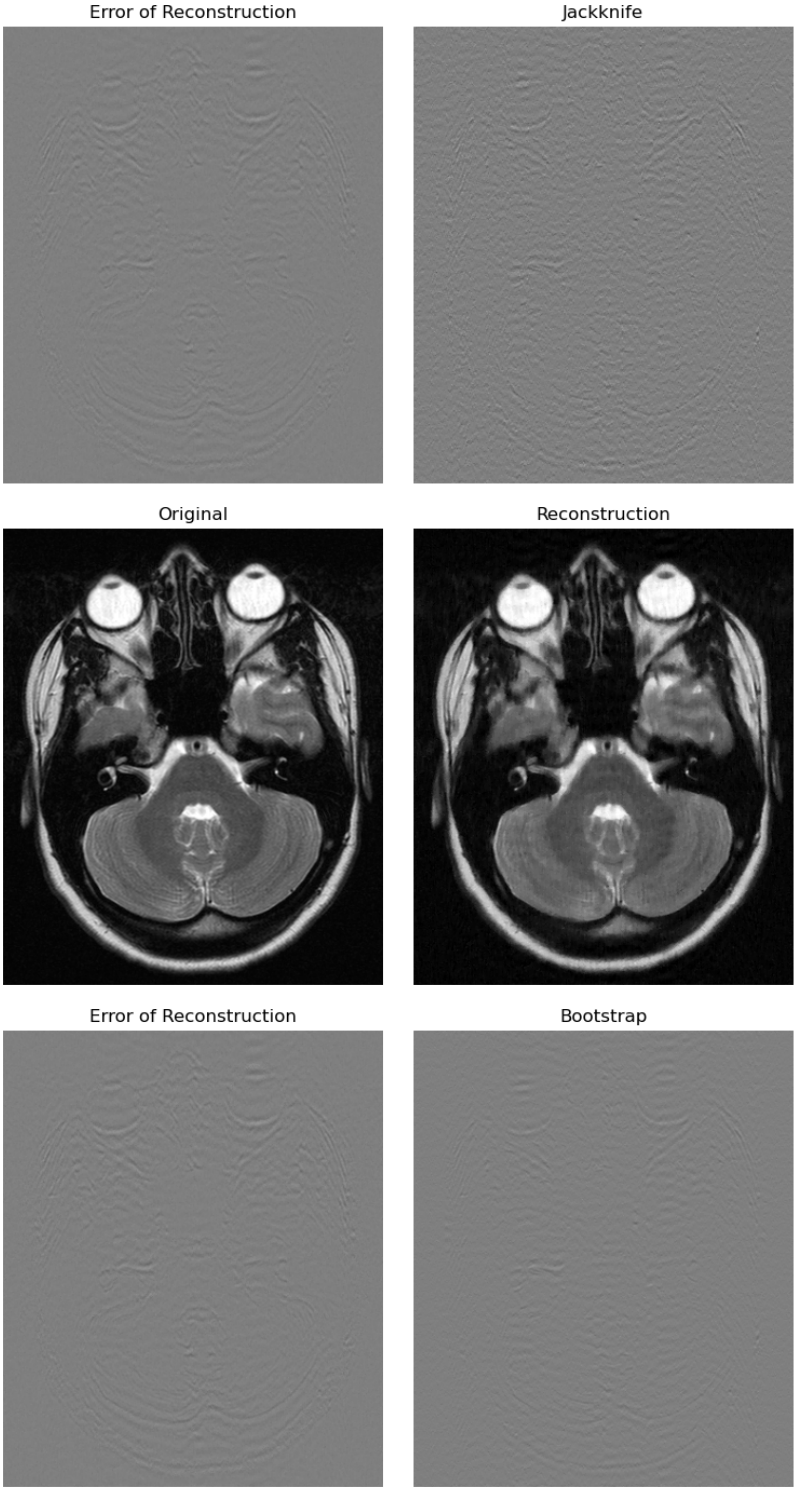

Figure 80: 2× horizontally retained sampling — slice 5

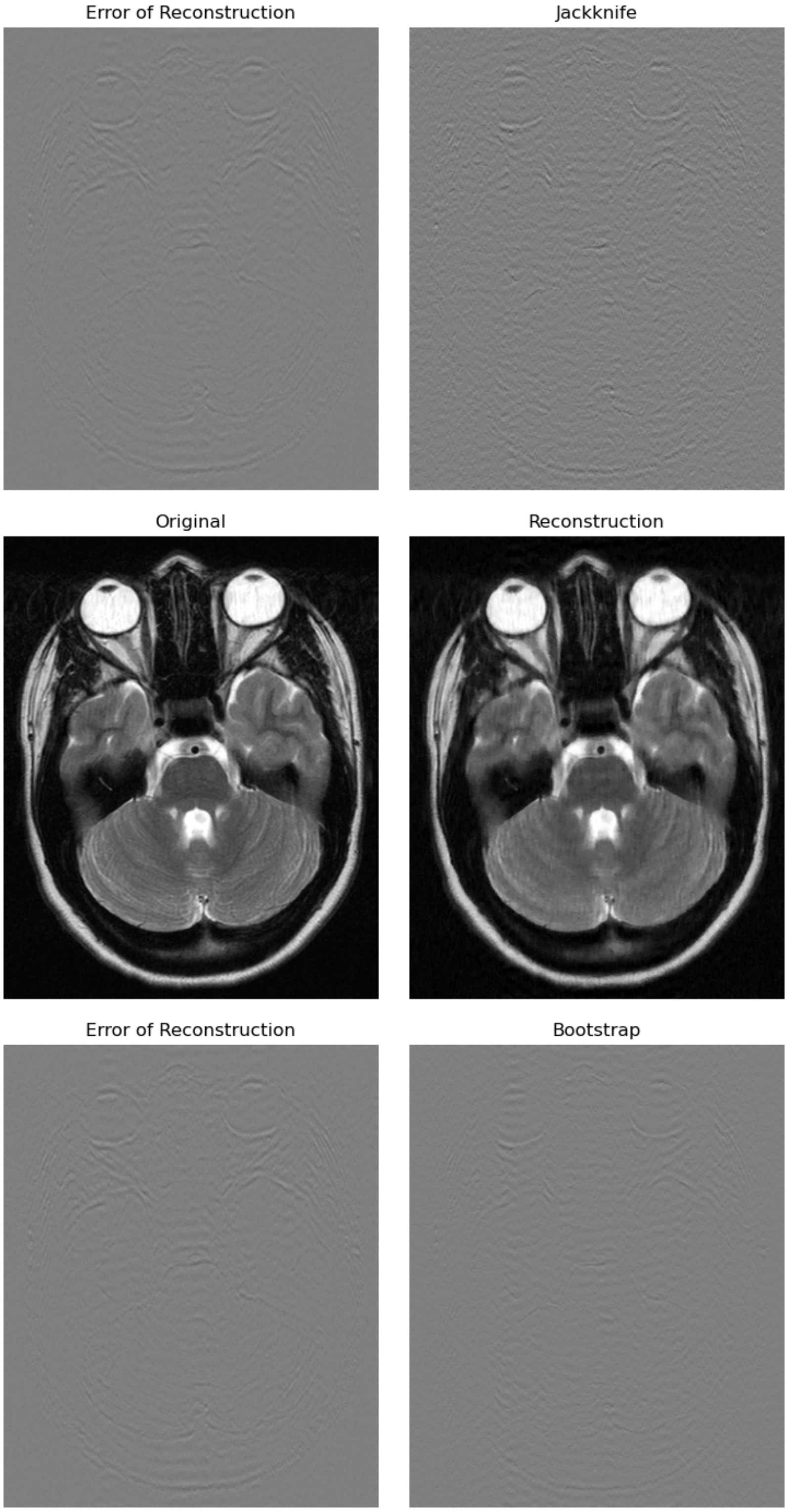

Figure 81: 2× horizontally retained sampling — slice 6

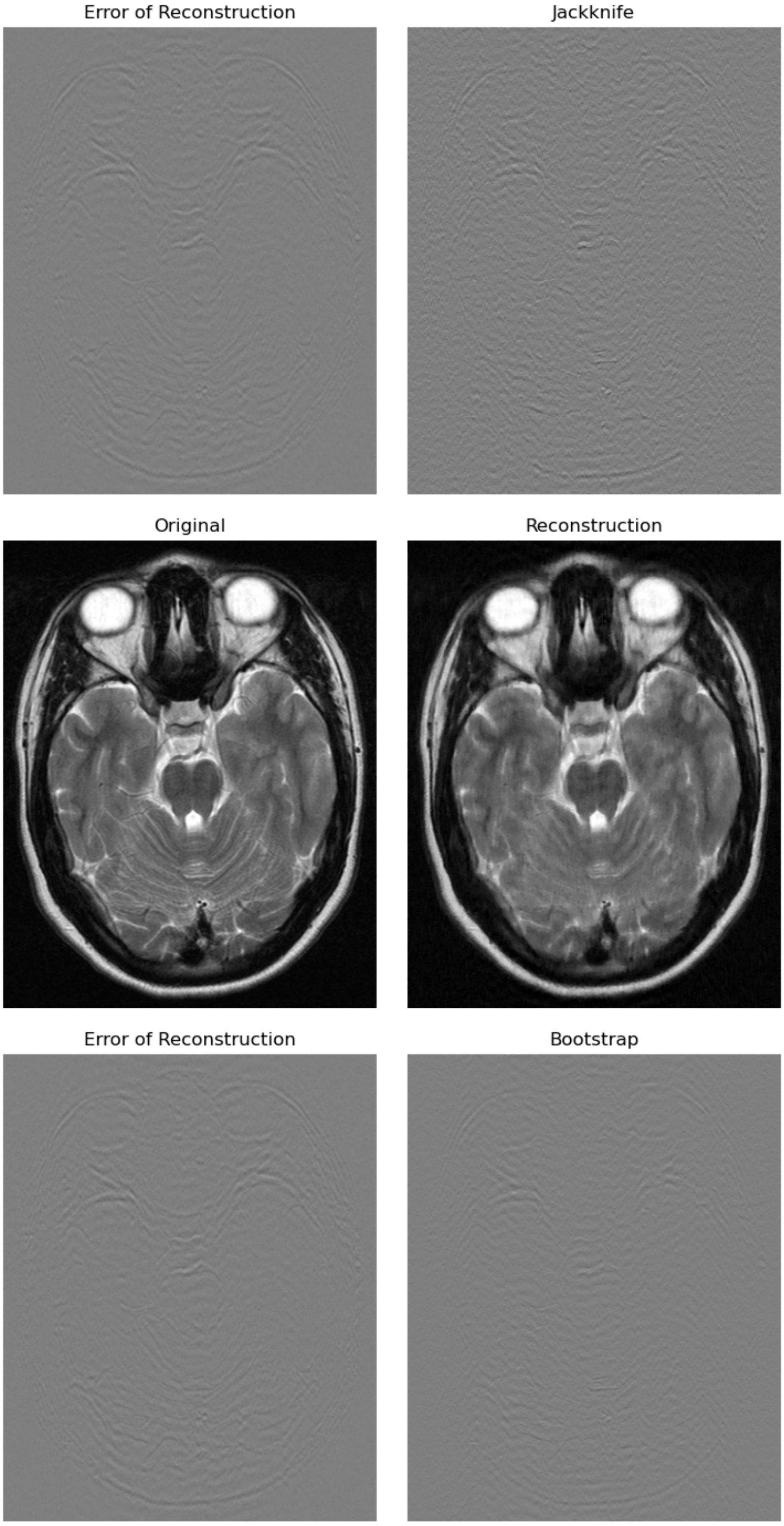

Figure 82: 2× horizontally retained sampling — slice 7

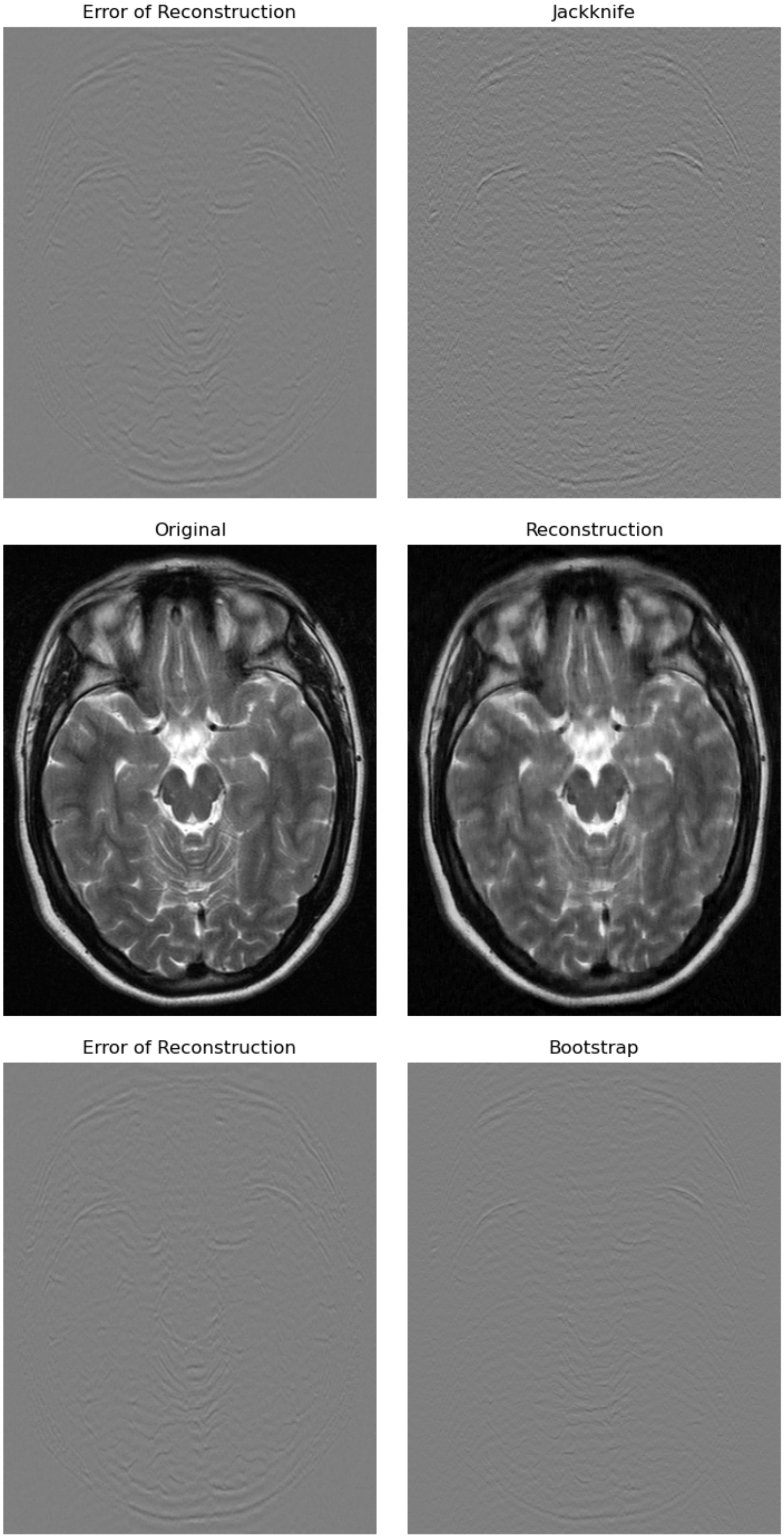

Figure 83: 2× horizontally retained sampling — slice 8

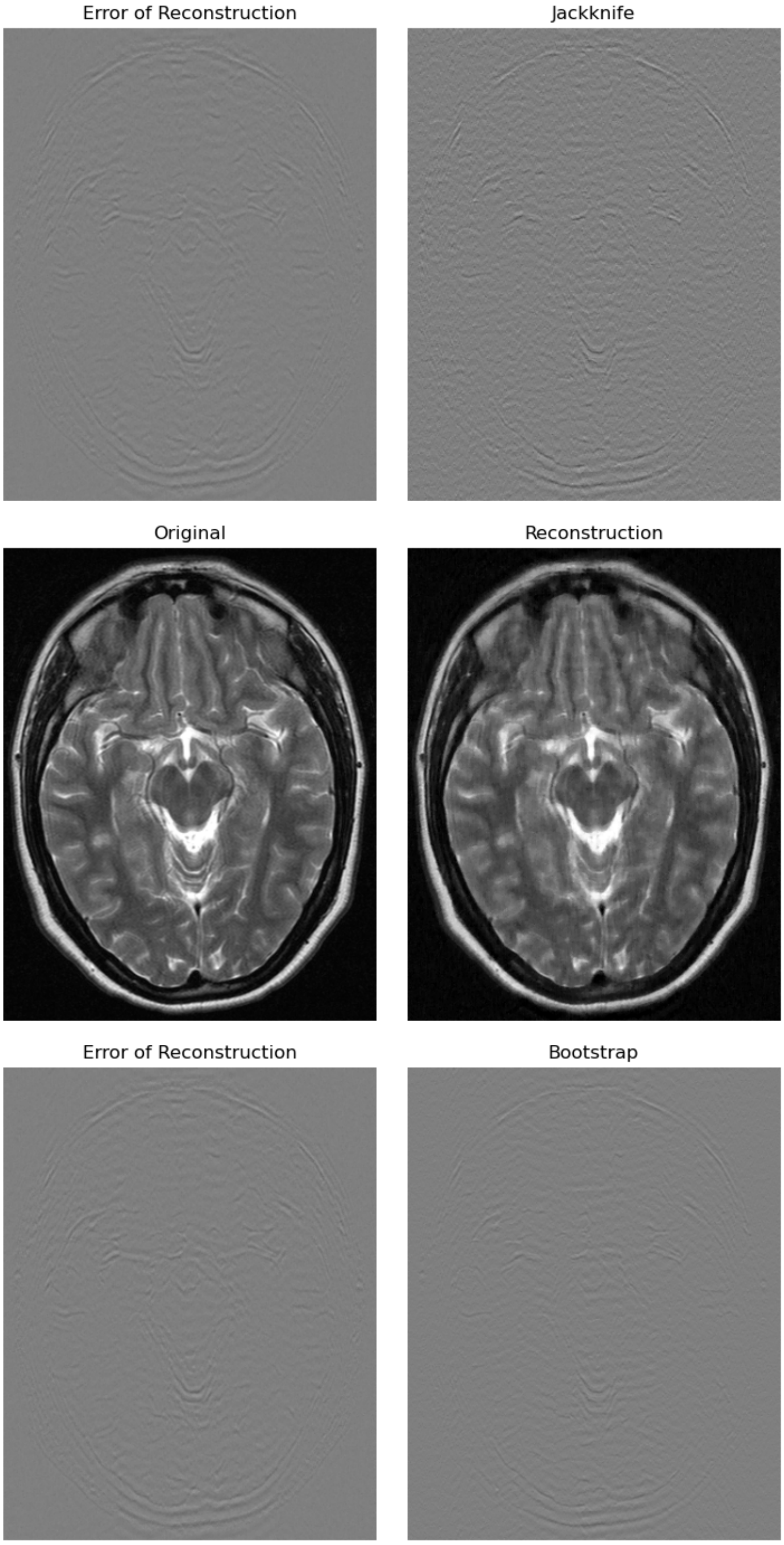

Figure 84: 2× horizontally retained sampling — slice 9

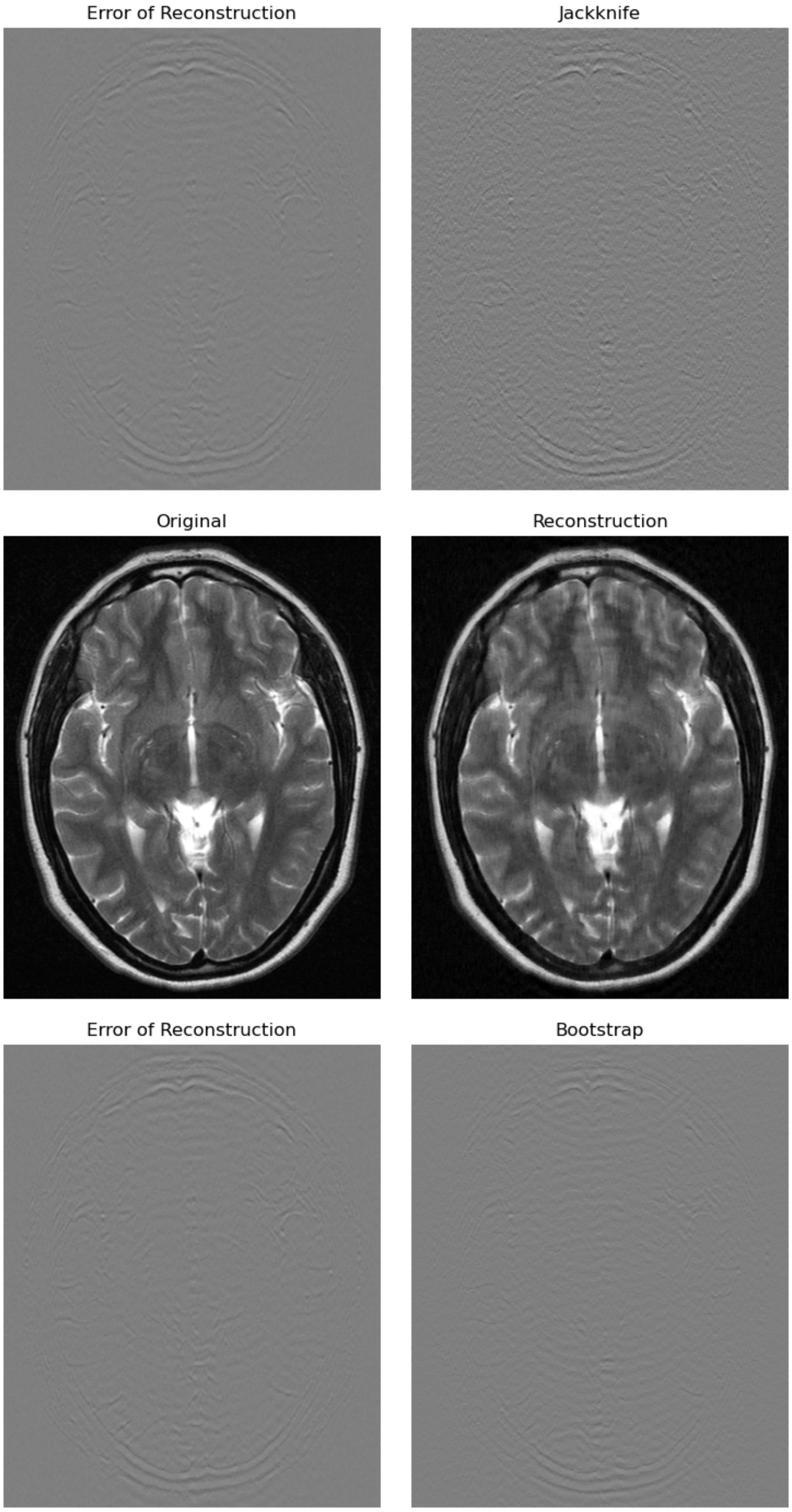

Figure 85: 2× horizontally retained sampling — slice 10

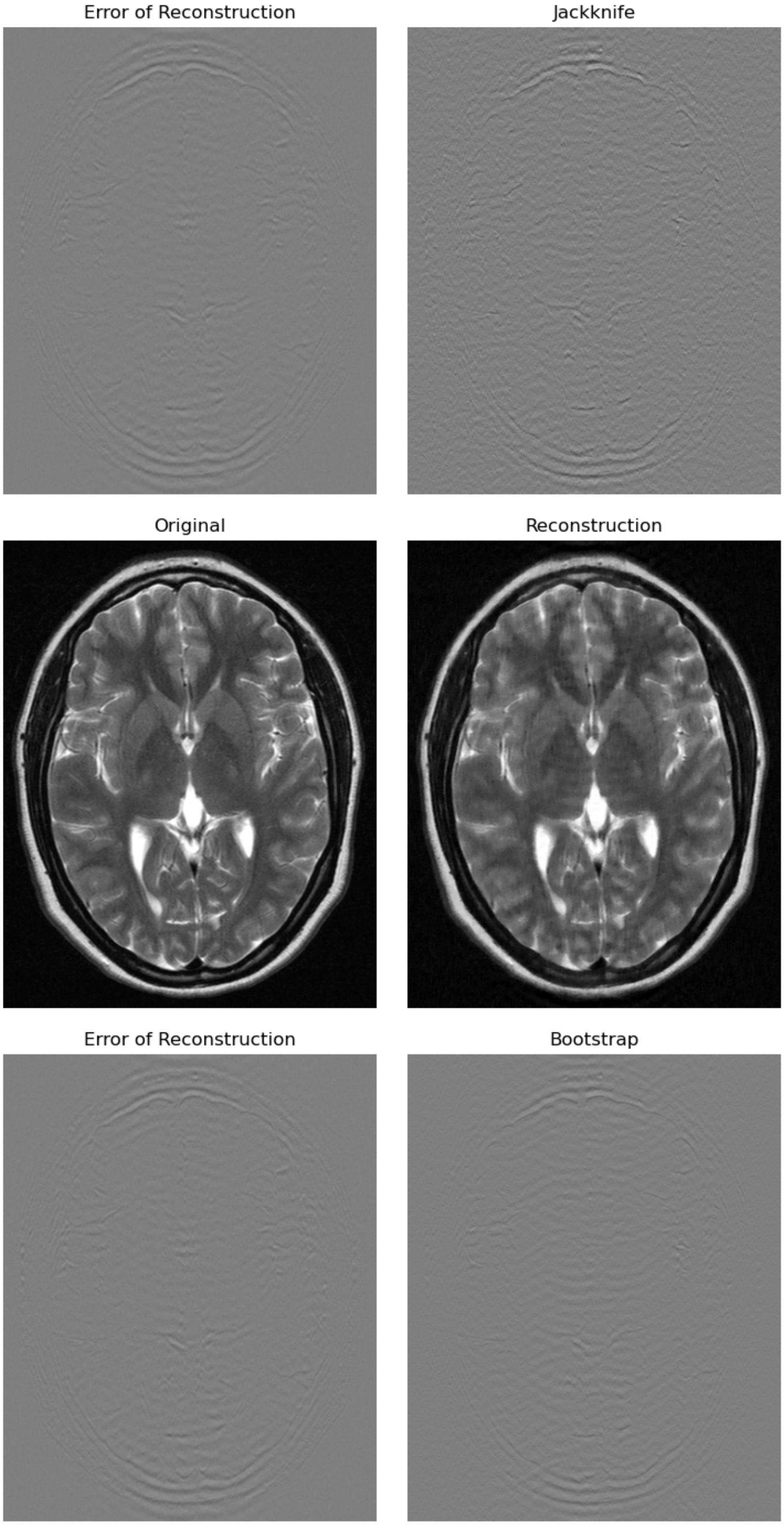

Figure 86: 2× horizontally retained sampling — slice 11

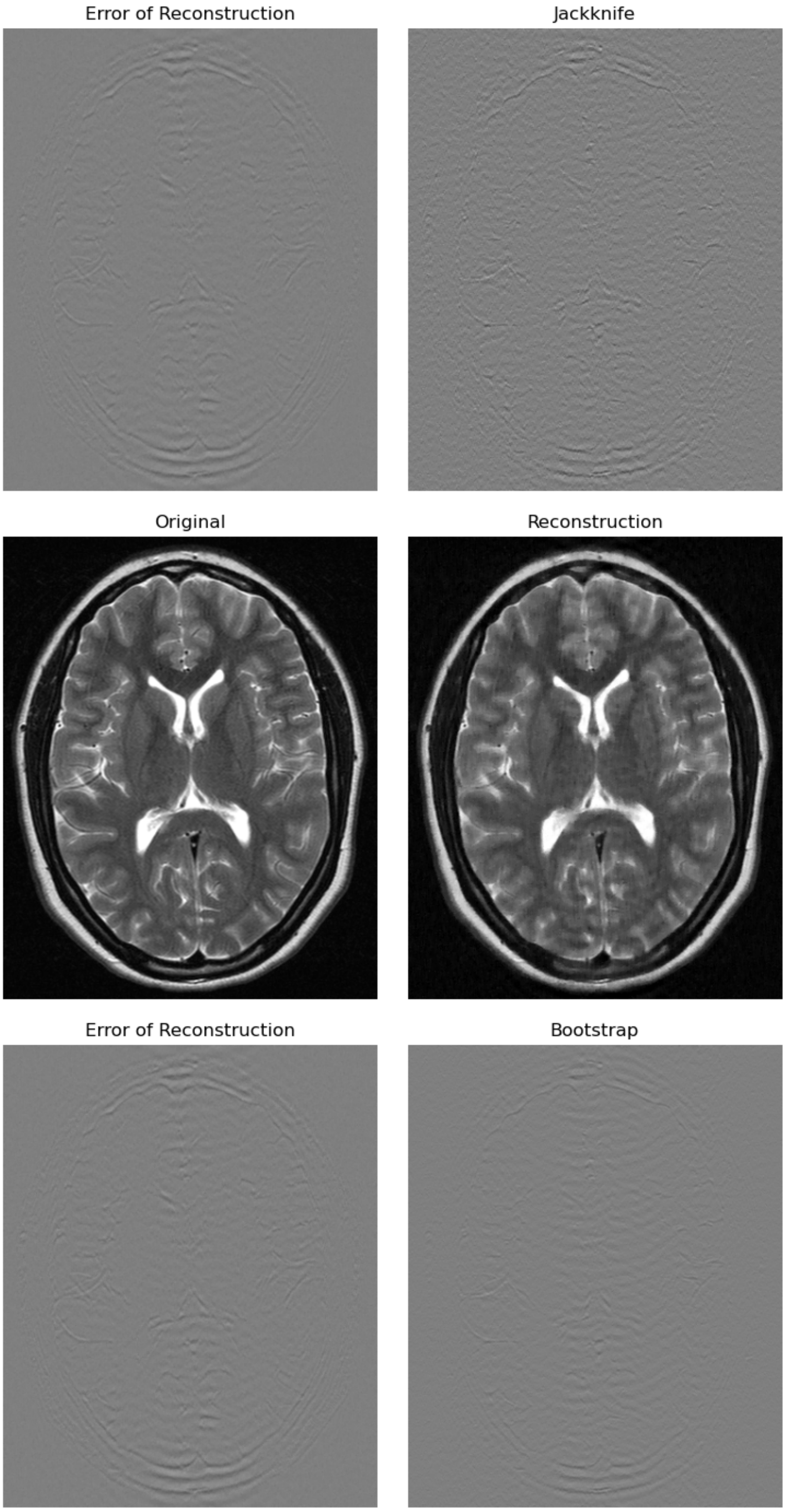

Figure 87: 2× horizontally retained sampling — slice 12

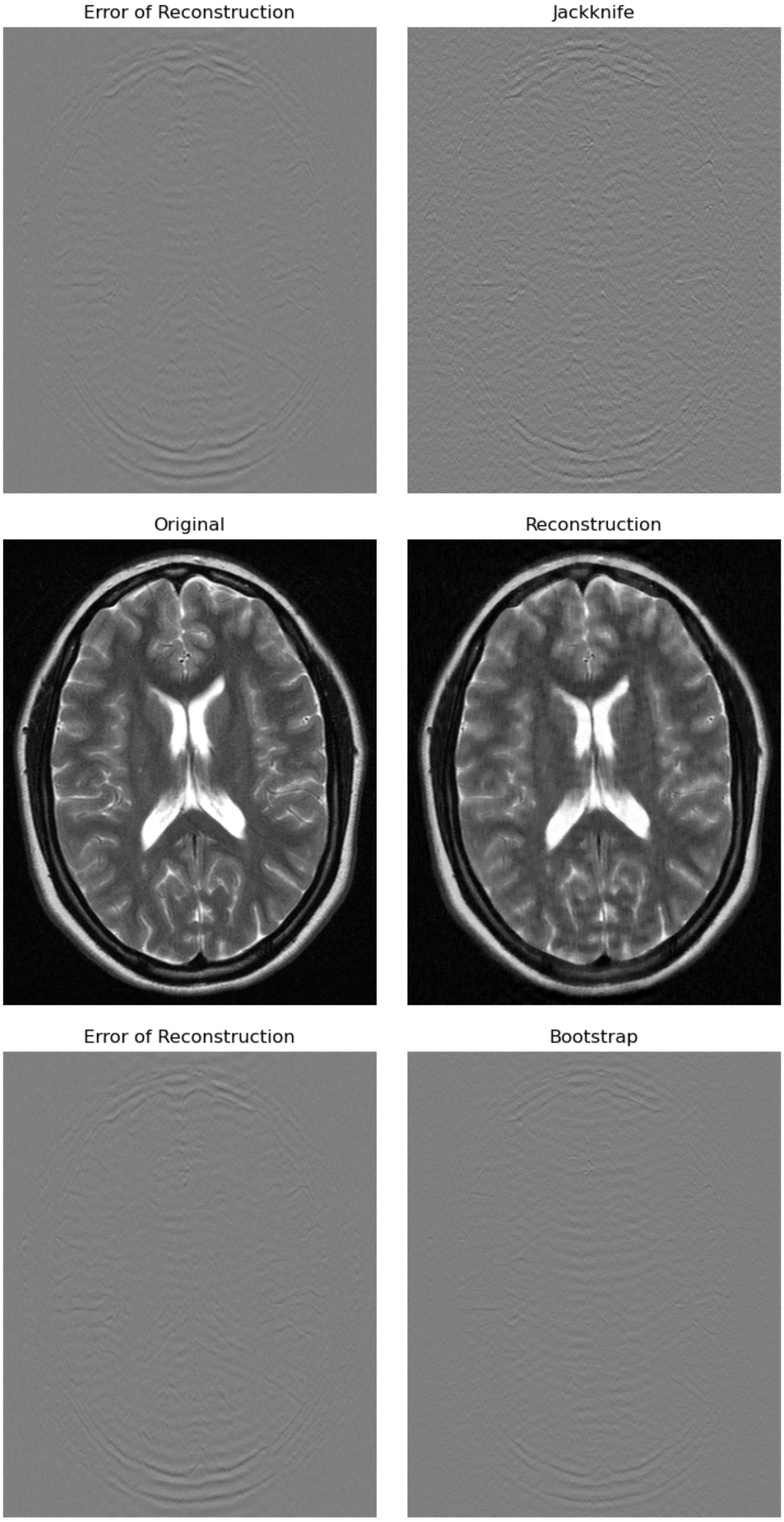

Figure 88: 2× horizontally retained sampling — slice 13

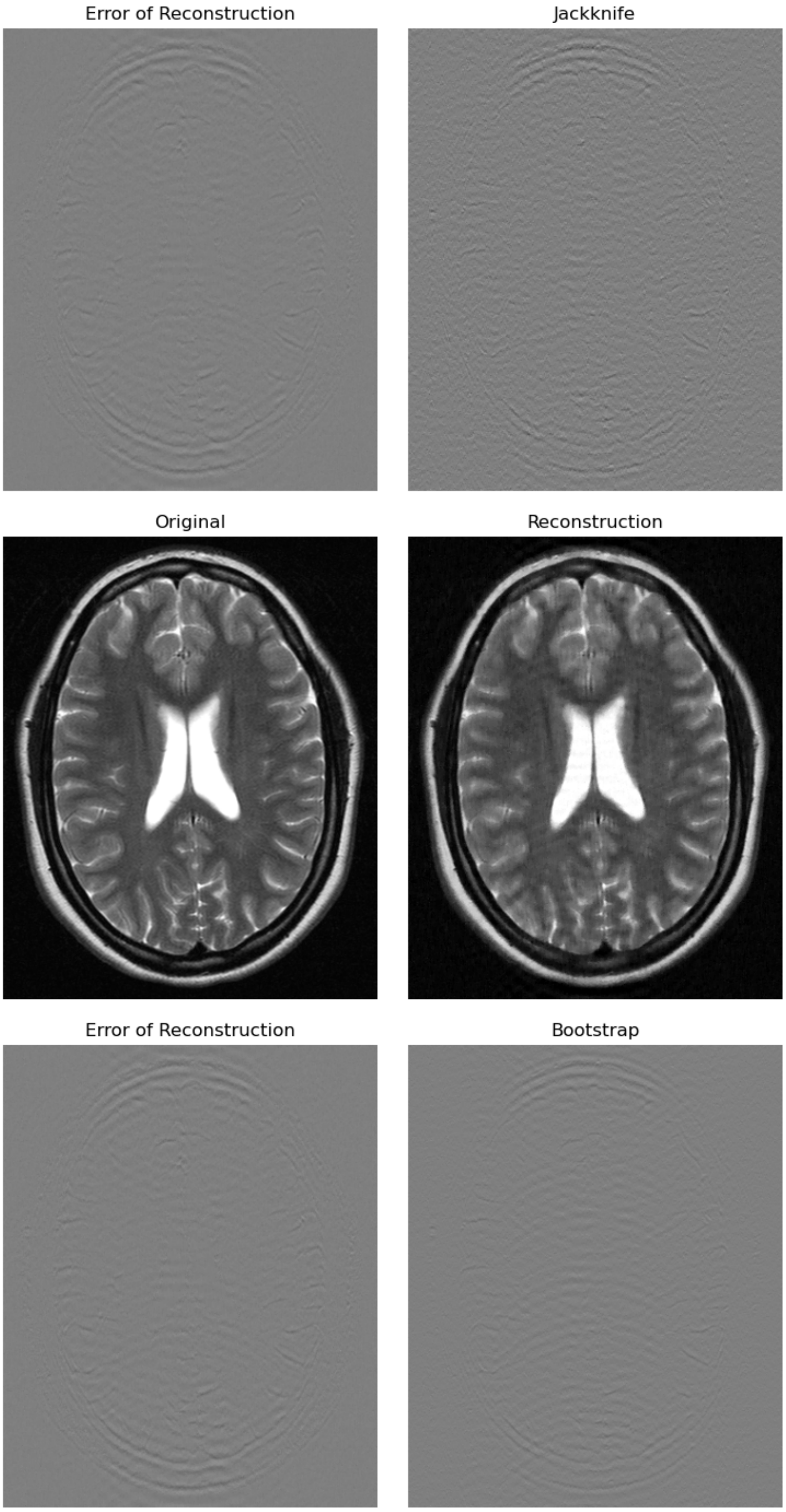

Figure 89: 2× horizontally retained sampling — slice 14

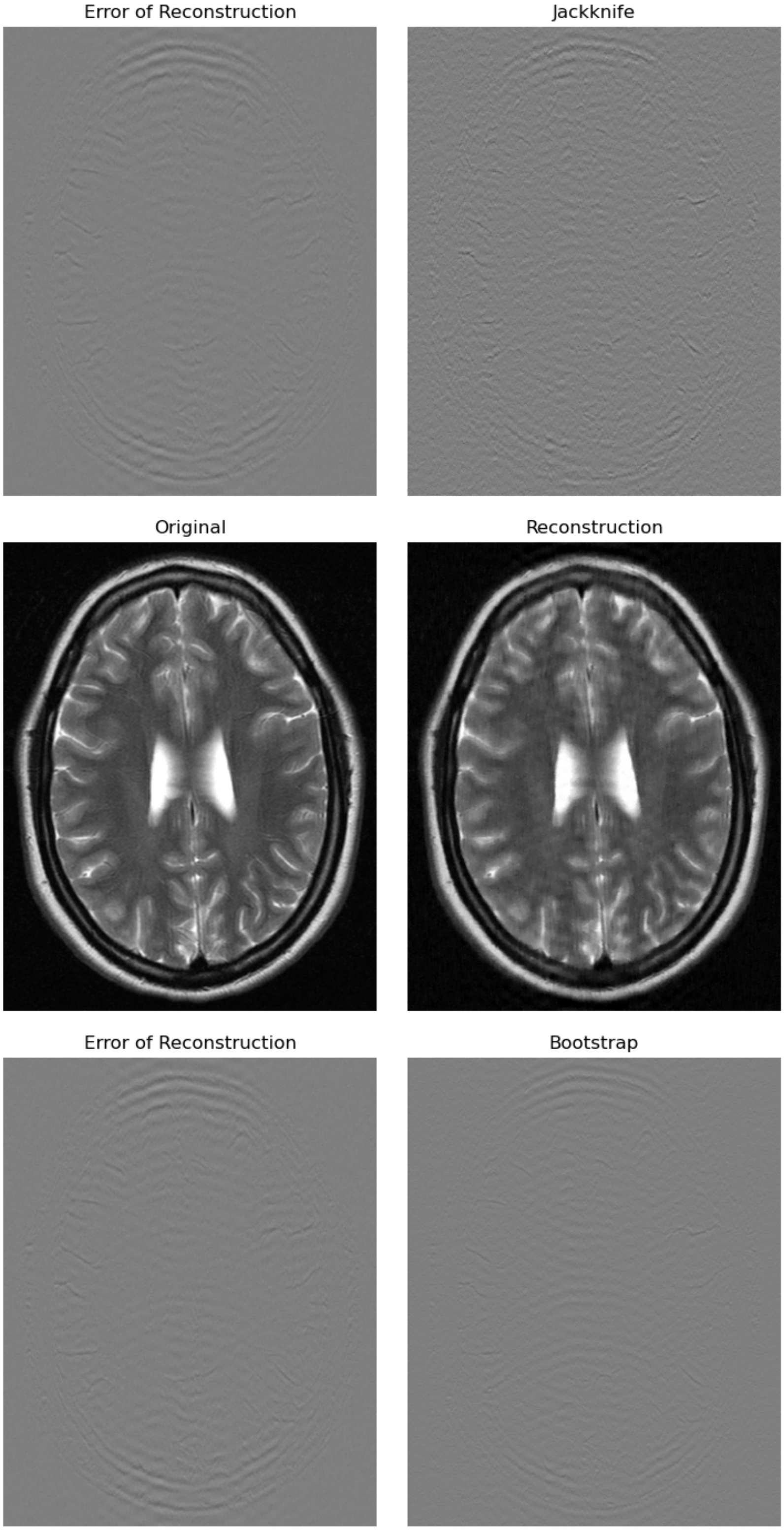

Figure 90: 2× horizontally retained sampling — slice 15

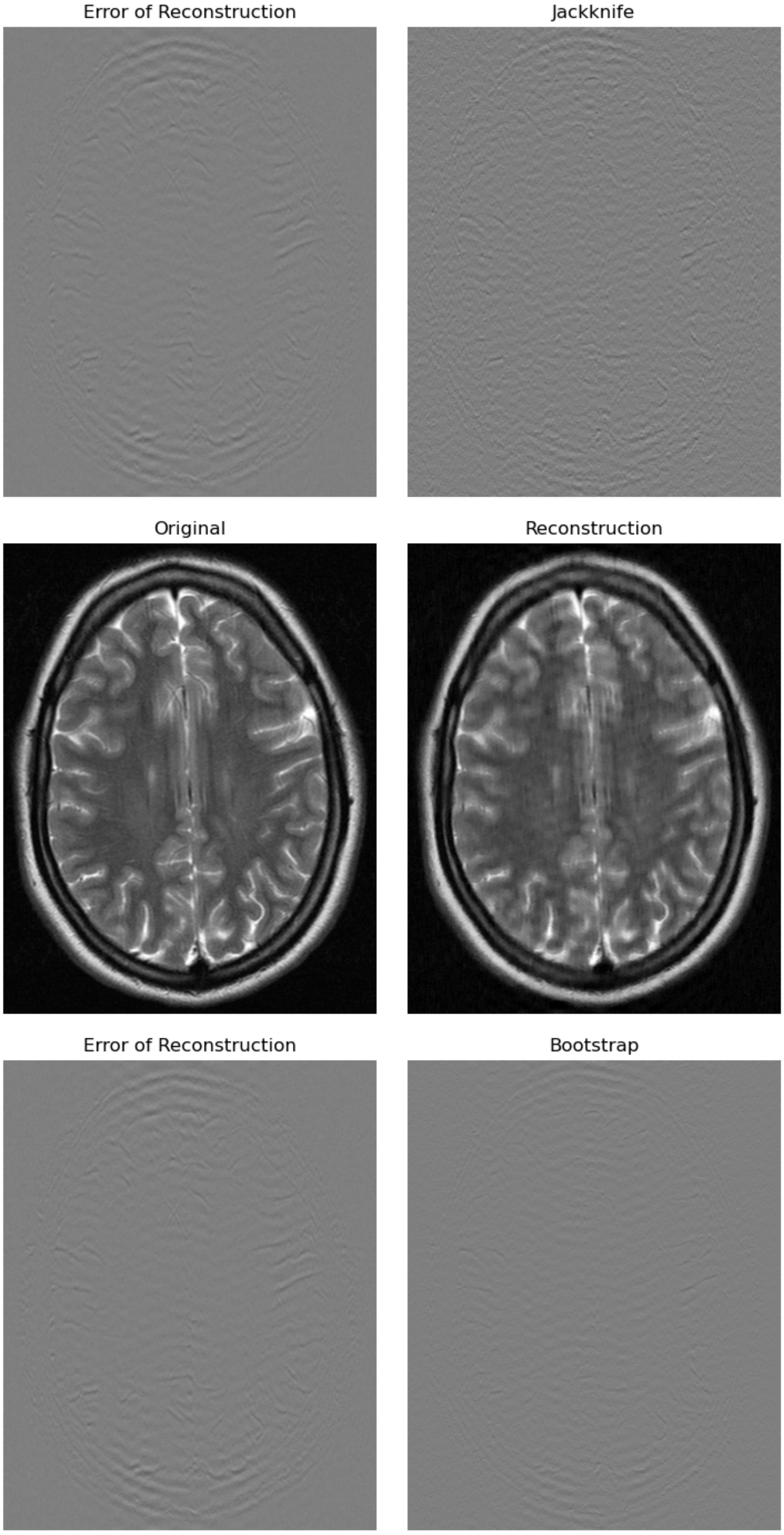

Figure 91: 2× horizontally retained sampling — slice 16

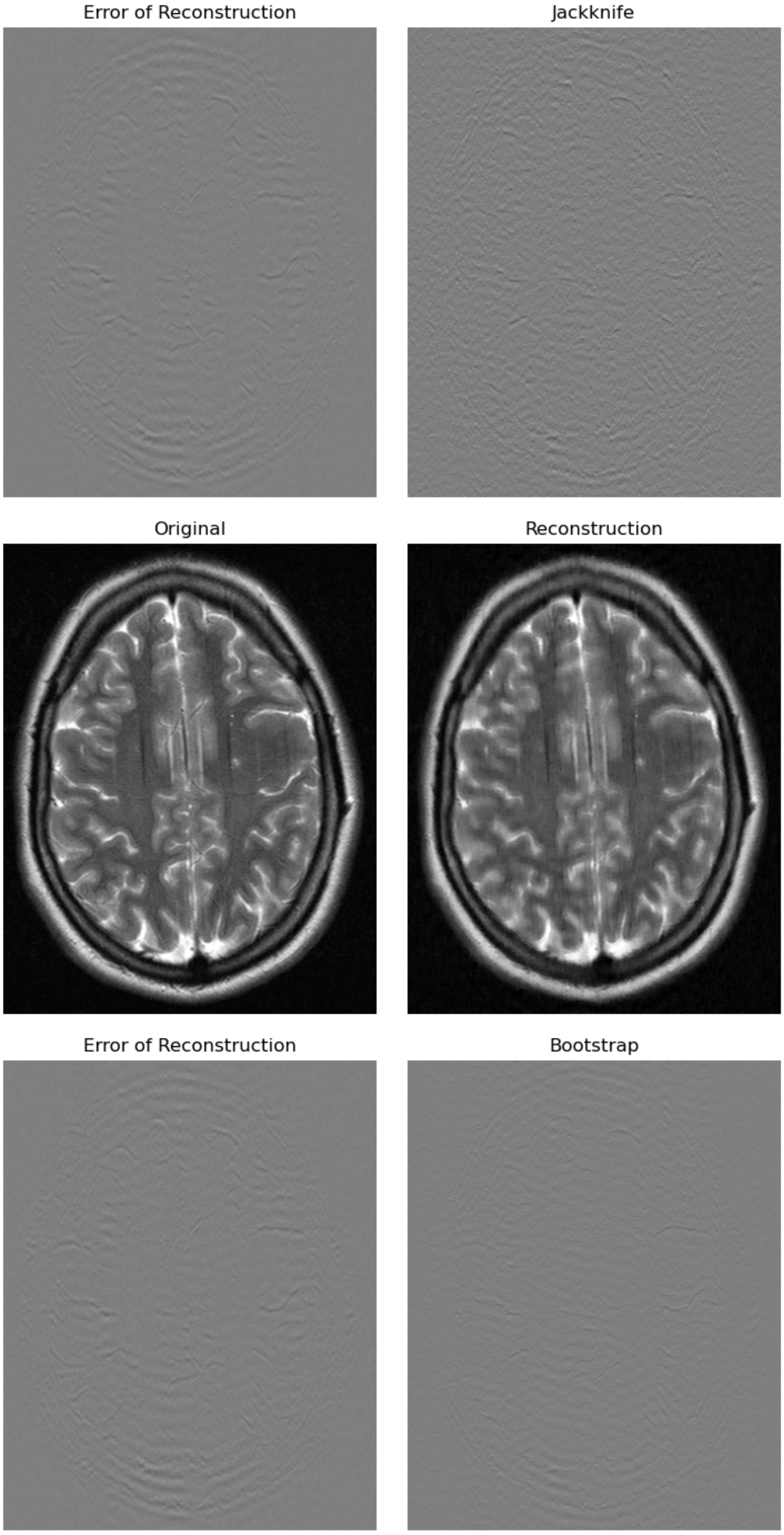

Figure 92: 2× horizontally retained sampling — slice 17

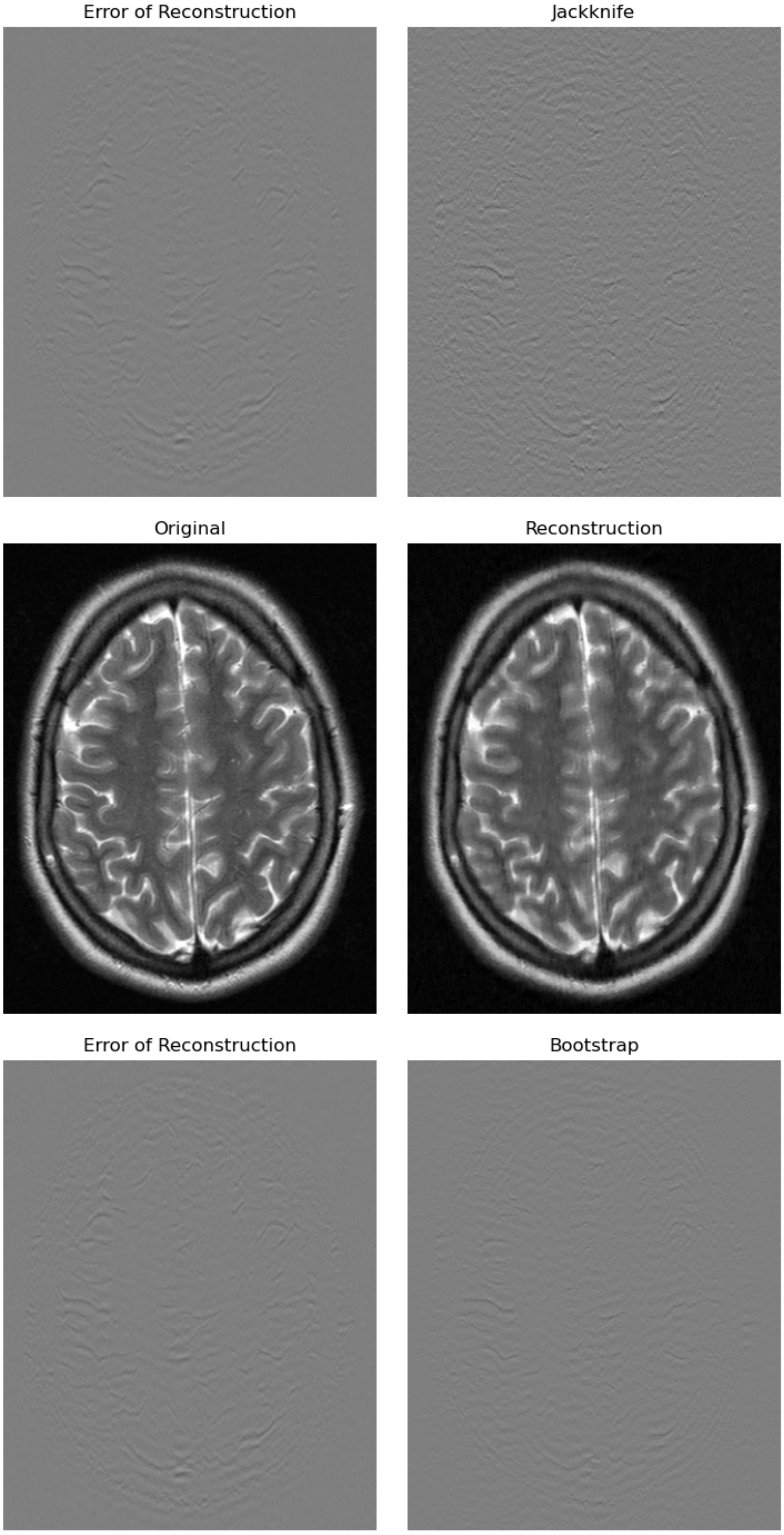

Figure 93: 2× horizontally retained sampling — slice 18

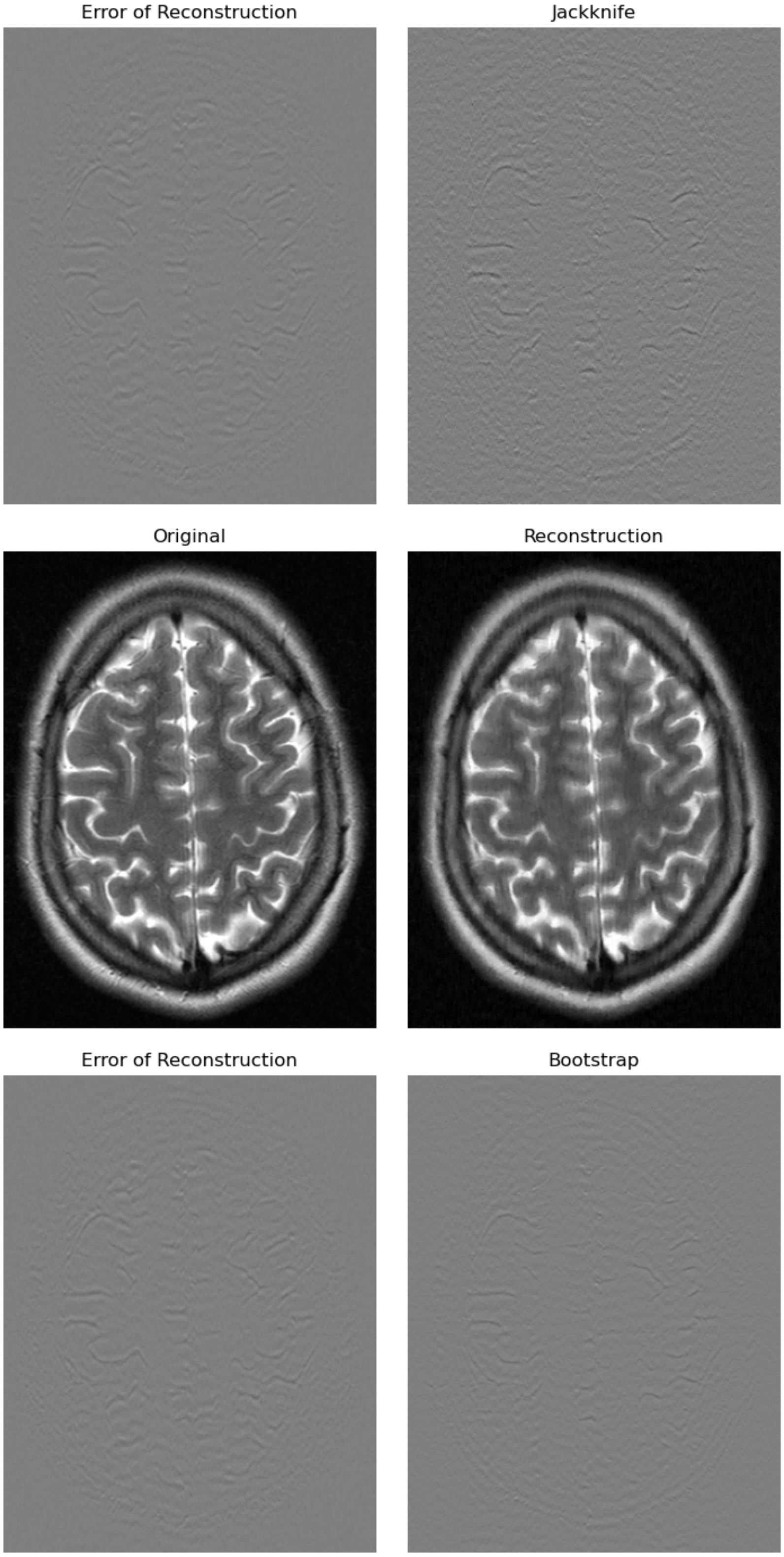

Figure 94: 2× horizontally retained sampling — slice 19

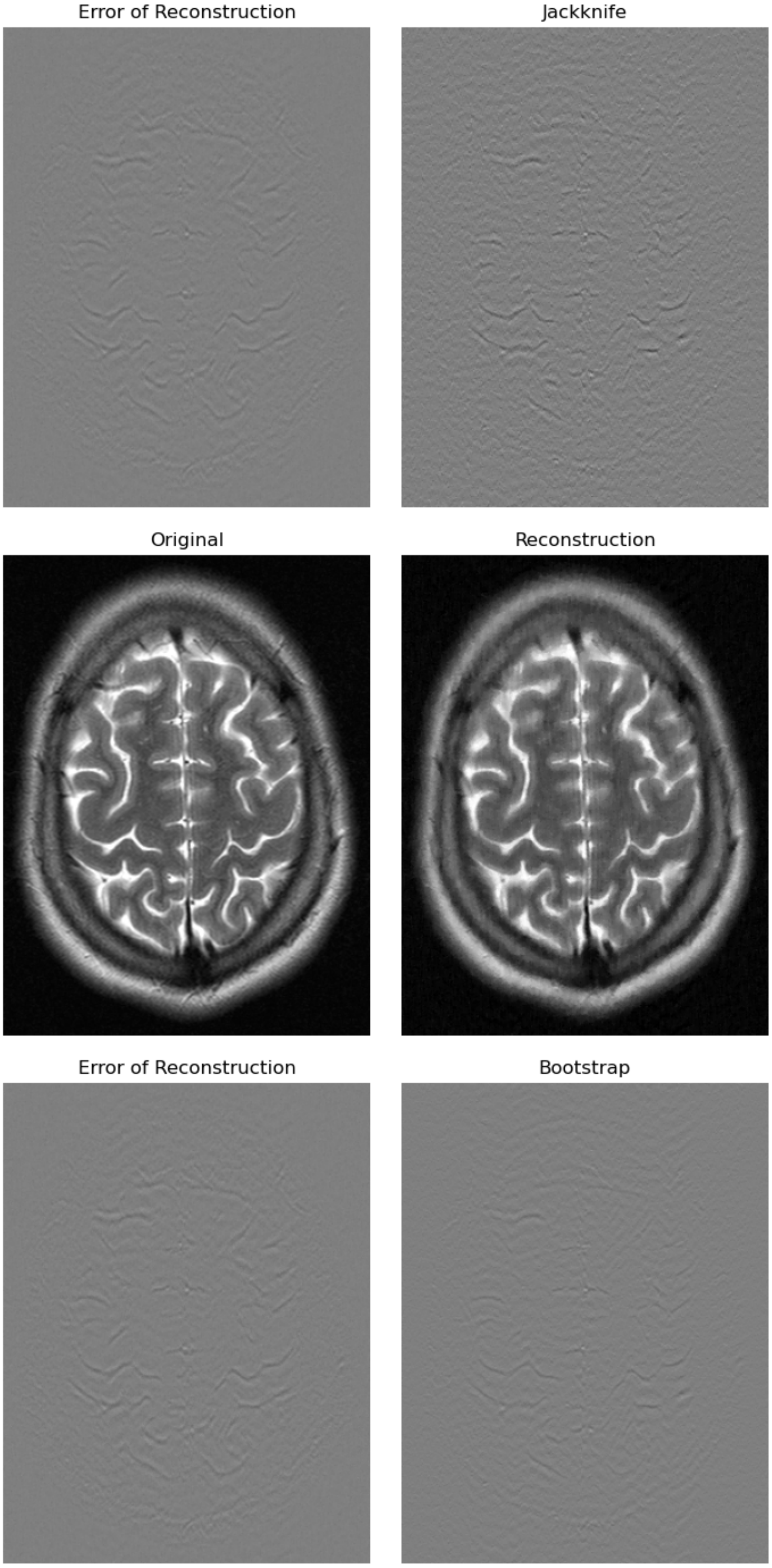

Figure 95: 2× horizontally retained sampling — slice 20

B. Blurred errors over a threshold overlaid

For reference, this appendix displays the same errors over a threshold overlaid over the reconstruction as in Subsection 2.2, together with the blurred errors over a threshold overlaid over the reconstruction (blurring with a Gaussian convolutional kernel whose standard deviation is one pixel, as in Subsection 2.3). The labeling conventions ("lower," "upper," etc.) conform to those introduced in Section 3. The blurred errors certainly introduce less distracting noise than without blurring, yet the colors still appear really distracting.

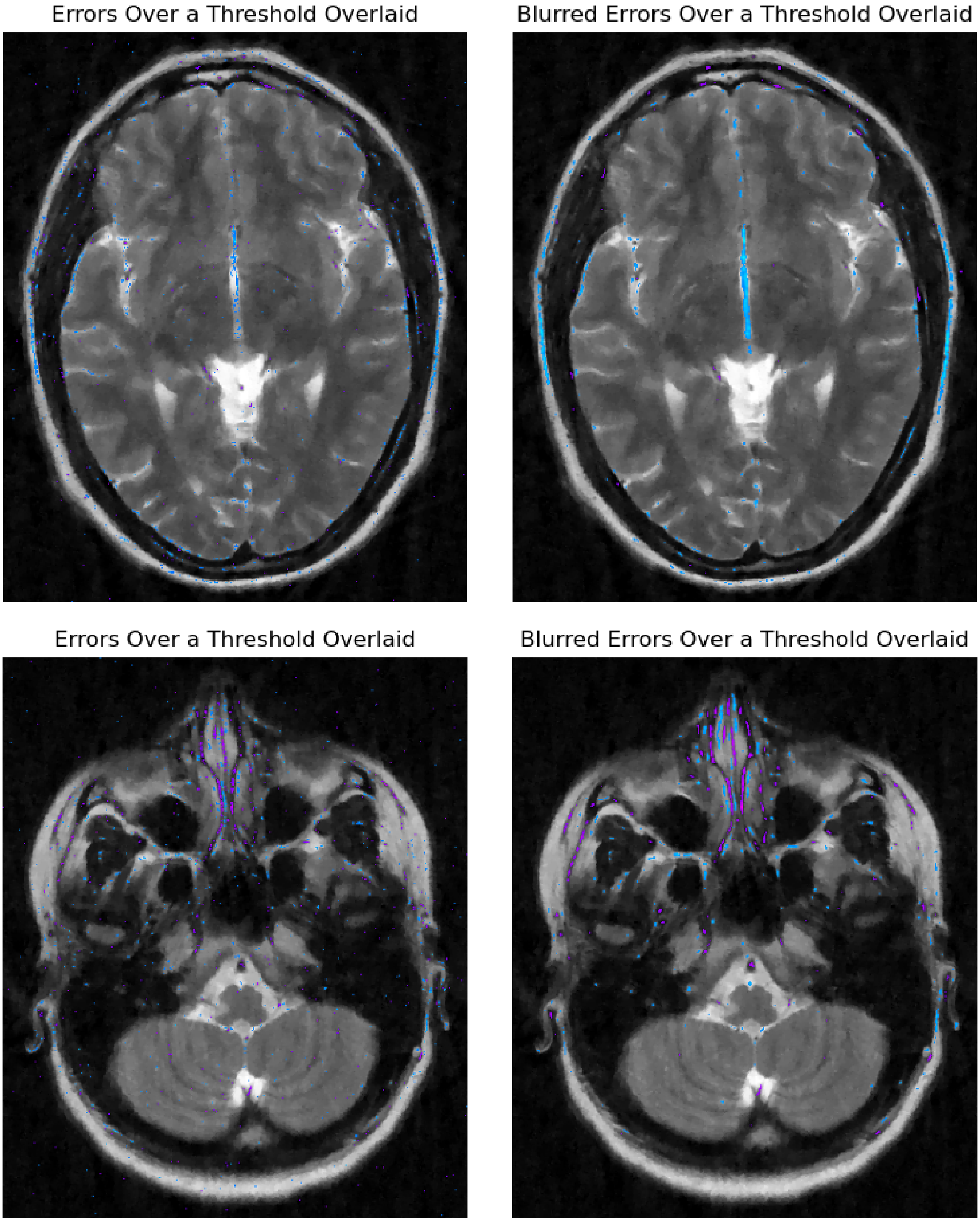
Errors Over a Threshold Overlaid
Blurred Errors Over a Threshold Overlaid
Errors Over a Threshold Overlaid
Blurred Errors Over a Threshold Overlaid

Figure 96: Radially sampling (b) — upper plots display the upper slice; lower plots show the lower

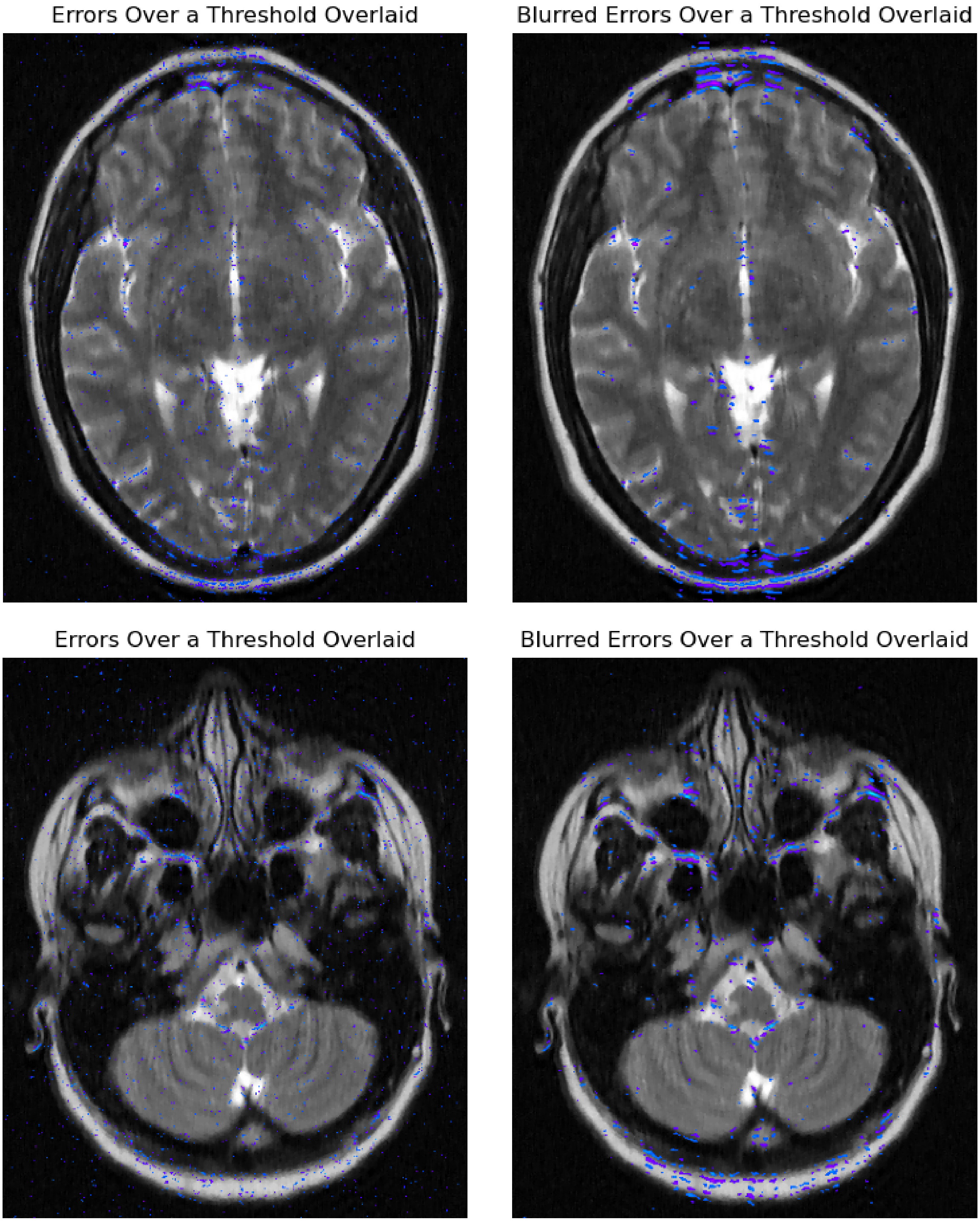
Errors Over a Threshold Overlaid
Blurred Errors Over a Threshold Overlaid
Errors Over a Threshold Overlaid
Blurred Errors Over a Threshold Overlaid

Figure 97: Horizontally sampling (b) — upper plots show the upper slice; lower plots show the lower

Affiliation:

Mark Tygert
Facebook Artificial Intelligence Research
1 Facebook Way
Menlo Park, CA 94025
E-mail: mark@tygert.com
URL: http://tygert.com

Journal of Data Science, Statistics, and Visualisation https://jdssv.org/
published by the International Association for Statistical Computing
http://iasc-isi.org/

MMMMMMM YYYY, Volume VV, Issue II *Submitted:* yyyy-mm-dd
doi:XX.XXXXX/jdssv.v000.i00 *Accepted:* yyyy-mm-dd