Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 22 No. 9 (2026) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v22i09.61101

PAPER

Implementation and Comparative Analysis of Magnetic Resonance Image (MRI) Reconstruction Using Generative Adversarial Networks (GANs)

Amauri da Costa Júnior(✉), Gerardo Antonio Idrobo Pizo, Cristiano Jacques Miosso

Universidade de Brasília, Brasília, Brasil

amauri_cj@hotmail.com

ABSTRACT

Magnetic resonance imaging (MRI) scans tend to be long procedures, making it difficult for patients to remain motionless. Generative adversarial networks (GANs) are artificial intelligence models widely used in image processing. This paper addresses the use of GANs to aid in the reconstruction of magnetic resonance images made with undersampled single-coil data, aiming to reduce MRI scan time. The GAN improves quality metrics of images reconstructed with different trajectories (vertical Cartesian, random Cartesian, radial, spiral 1, and spiral 4), which present distortions due to not meeting the Nyquist–Shannon sampling theorem. Tests are performed by feeding the trained model with images made from undersampled data to generate enhanced images. The generator receives these images and attempts to produce enhanced versions based on typical MRI scans, while the discriminator compares generated and typical images to determine which are real or fake. At each epoch, both models refine themselves based on the discriminator's verdict, improving image quality. Metrics such as the structural similarity index and signal-to-noise ratio are extracted from the produced images. In the best tests, increases of 4× in average SSIM and up to 14× in average SNR were achieved.

KEYWORDS

magnetic resonance imaging (MRI), generative adversarial networks (GAN), reconstruction, undersampled, U-net

1 INTRODUCTION

Imaging tests are among the most widely used tools in medicine for the diagnosis and treatment of various pathologies. Among the various types of tests, magnetic resonance imaging (MRI) has proven to be one of the most versatile, and its use grows every year [1].

Da Costa Júnior, A., Pizo, G. A. I., Miosso, C. J. (2026). Implementation and Comparative Analysis of Magnetic Resonance Image (MRI) Reconstruction Using Generative Adversarial Networks (GANs). *International Journal of Online and Biomedical Engineering (iJOE)*, 22(9), pp. 76–89. https://doi.org/10.3991/ijoe.v22i09.61101

Article submitted 2026-02-16. Revision uploaded 2026-04-09. Final acceptance 2026-04-09.

Magnetic resonance imaging can obtain images of soft or dense bodies and even blood flow, volume, and oxygenation [2]. It is also an exam that uses non-ionizing radiation, which brings significantly lower health risks compared to other imaging exams, such as CT scans [3], which, despite being a safe exam with acceptable amounts of radiation, cannot be done frequently so that the dose of radiation absorbed by the body does not exceed the limits, which, if it occurs, increases the chances of cancer developing in the patient [4].

One of the downsides of MRI is the discomfort it causes. The exam is long, and the patient must remain motionless in a cramped and noisy environment. This makes it difficult to remain in for long periods of time, causing involuntary movement in some patients, hindering signal acquisition, and compromising the quality of the final image by creating artifacts [5].

Magnetic resonance imaging consists of applying a sequence of magnetic pulses throughout the patient's body, causing the hydrogen nuclei of the body's atoms to align with the imposed field and subsequently emit signals that are captured by coils. Those signals are stored in a data matrix called k-space, which is later used to create the image [6].

There are several ways to fill the k-space; these shapes are called trajectories and are chosen according to the type of exam, each with its positive and negative aspects [7]. Figure 1 shows some of these trajectories and also shows in its second row of images how the sub-sampling of the k-space would be with each of the trajectories and in the third row, the final image produced by each of these k-spaces.

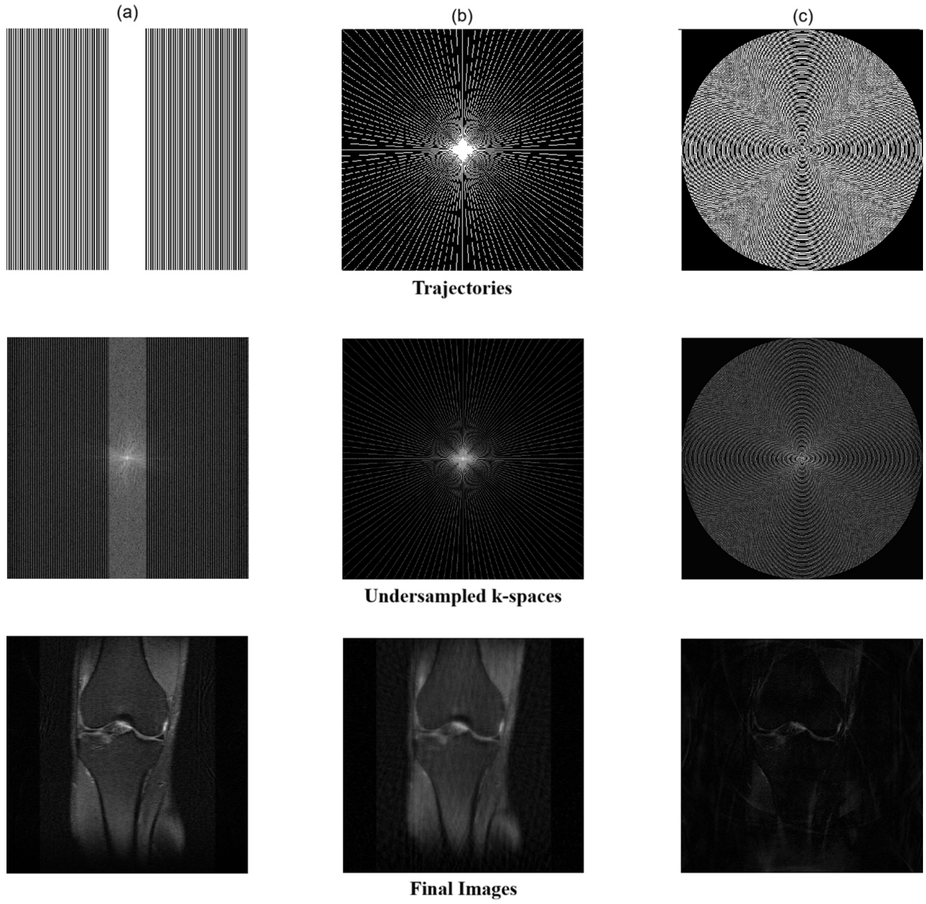

Fig. 1. Trajectories for obtaining k-space – (a) Vertical Cartesian trajectory; (b) Radial trajectory; (c) Spiral trajectory

One way to reduce scan time is to acquire fewer body signal samples, reducing the size of the k-space. However, decreasing the number of samples reduces the quality of the final image. The reconstructed image with fewer samples is blurry, with little detail. Figure 2 shows a comparison between the quality of the reconstructed images for different numbers of rows in k-space. Therefore, a balance must be struck between scan time (number of samples acquired) and the desired image sharpness.

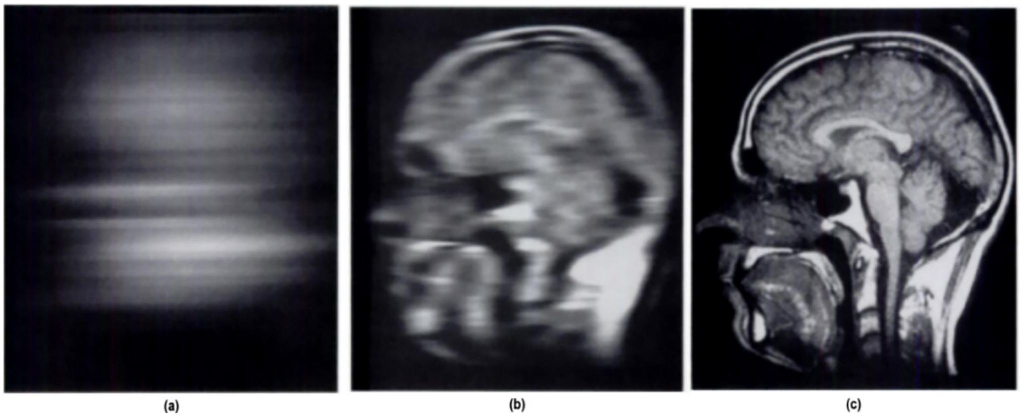

Fig. 2. Comparison between images produced from k-spaces with different numbers of rows: (a) 2 rows; (b) 16 rows; (c) 256 rows [8]

Studies are being conducted to reduce exam time by acquiring fewer body signals without compromising image quality. Techniques such as parallel imaging, compressed sensing, improved magnetic field gradients, and improved magnetic pulse sequences are just some of the techniques employed to reduce this time. Reducing exam time also has the benefit of optimizing machine time, as faster exams would allow more patients to be examined, reducing costs and patient discomfort.

One of the most widely explored areas of research today to reduce examination time is the use of machine learning to reconstruct and enhance these images. Image processing models that utilize machine learning and deep learning are widely used because they can detect patterns that other mathematical or software models would not [9].

This work proposes the use of generative adversarial networks (GANs) for processing images produced with a number of samples below the Nyquist–Shannon rate in order to improve their quality based on metrics.

The GAN model is separated into two distinct neural networks: a generative network that creates images to simulate images from a reference dataset and a discriminator neural network that evaluates whether the created images belong to the reference dataset or were created. If the created image is identified as fake, the first neural network makes adjustments and repeats the process, while the second network improves in identifying fake images. In this way, the two neural networks are perfected over time, the first creating images increasingly similar to those in the dataset and the second increasingly refined in identifying created images. At the end of the process, a generator network is obtained that is capable of highlighting patterns in distorted images based on high-quality images [10].

Models of this type can be used to process MRI images, using images made with undersampled data to feed the generating network, and images made with fully sampled data as a reference dataset. In this way, it is possible to improve the quality of MRI images made with undersampled data by enhancing images until they are confused with a control group, which in this case would be images reconstructed from fully sampled k-spaces [11].

Several other approaches to using GANs for magnetic resonance image reconstruction have been studied in recent years. In [11], the authors propose using different undersampling techniques in the same training to achieve better enhancement results, obtaining SSIM scores of up to 0.92.

In [12], the authors use a U-net-based model similar to that used in this work. The authors use the neural network to enhance images made with subsampled data and combine the k-space of these new images with the original undersampled k-space to obtain the final image, achieving high-resolution images with only 29% of the original data.

This paper presents a unique approach to analyzing the reconstruction of images created from subsampled data with different trajectories, demonstrating the impact of these trajectories on the final image, evaluating their performance and particularities, and highlighting the positive and negative points of each. This type of analysis allows us to observe how the data acquisition trajectories can be better adjusted for potential GAN-assisted reconstructions.

2 MATERIALS AND METHODS

2.1 Dataset description

The k-space database from the FastMRI research group was used to train the GAN. This group encourages research and competitions in the area of MRI image reconstruction and enhancement from undersampled data using artificial intelligence. The competitions aim to continually improve the similarity index of reconstructed images and the signal-to-noise ratio and are held across several categories. The database was created specifically to allow researchers to work with raw MRI image data free of charge, and all images contained within it are previously anonymized [13].

The dataset used will be the same as that used in the knee single-coil image reconstruction challenge. The single-coil image sets were not directly obtained by a single coil; they are simulated and obtained by combining samples from multiple coils [14]. The advantage of performing tests with single-coil samples is that they can be performed on computers with lower processing power. This allows for relevant results for research in the field by performing tests that generate results more quickly, thus allowing for more efficient refinement by performing the training multiple times and altering the parameters.

The complete set of fully sampled 512×512 pixel images has 34,742 slices, so training was performed using a minimum of 973 images up to 6,000 images. Only fully sampled images were used because the study aimed to observe the impact that different sampling trajectories would have on GAN-assisted reconstruction. Therefore, fully sampled k-spaces were used and then subsampled according to the training specifications.

2.2 Preprocessing and undersampling simulation

Preprocessing is performed on the images, consisting of a sequence of steps to prepare the dataset images for processing by the GAN. The images used in the dataset are made from fully sampled k-spaces; therefore, it is necessary to simulate the undersampling of this k-space. To do this, some samples are deleted from the matrix according to the chosen trajectory, and then the image is reconstructed, as shown in Figure 1. This process of deleting samples in k-space is commonly used in studies of this type because it faithfully simulates the undersampling generated by resonance equipment [14].

The preprocessing algorithm has several specifications that can be modified as needed. These specifications alter undersampling parameters, such as sampling rate, trajectories, and the presence or absence of artifacts. During preprocessing, the images are modified so that they are all the same size and have the same height and width. This task is performed without resizing the images; instead, all images are zero-completed until they have the same dimensions as the largest image. The model also provides the option to fix the image dimensions, cropping images larger than the value and zero-completing those smaller. These dimensions are always rounded to powers of two, as GAN processing involves multiple divisions by two, simplifying this process.

The algorithm has three types of trajectories that simulate trajectories typically used in the medical industry: Cartesian, random Cartesian, spiral 1, spiral 4, and radial. Figure 3 shows the trajectories with different sampling levels.

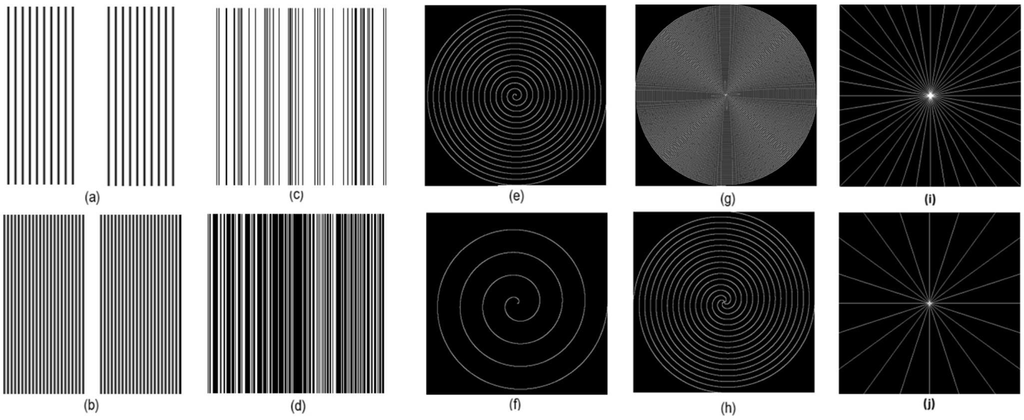

Fig. 3. Trajectories that can be applied to k-space with different sampling levels: (a) Cartesian with 24.22% undersampling; (b) Cartesian with 49.22% undersampling; (c) Random Cartesian with 19.92% undersampling; (d) Random Cartesian with 69.92% undersampling; (e) Spiral 1 with 93.79% undersampling; (f) Spiral 1 with 98.43% undersampling; (g) Spiral 4 with 75.16% undersampling; (h) Spiral 4 with 93.72% undersampling; (i) Radial with 96.13% undersampling; (j) Radial with 98.06% undersampling

After performing preprocessing in all k-spaces, the images are reconstructed using Inverse Fast Fourier Transform (IFFT) and then separated into three groups to be used by the GAN: the training group, the validation group, and the test group. The three groups can undergo different preprocessing depending on the application being studied, and the algorithm allows the selection of different parameters for the training, validation, and test sets for each of the trajectories.

The images are stored in pairs containing two images from the same exam, concatenated side by side as shown in Figure 4. On the left side is the image that will be used to train the GAN's discriminative model. These are undistorted images made from fully sampled data. On the right side is an image made with data undersampled according to the specifications used in preprocessing; these will be the images used by the GAN's generative and discriminative model as input to create new images.

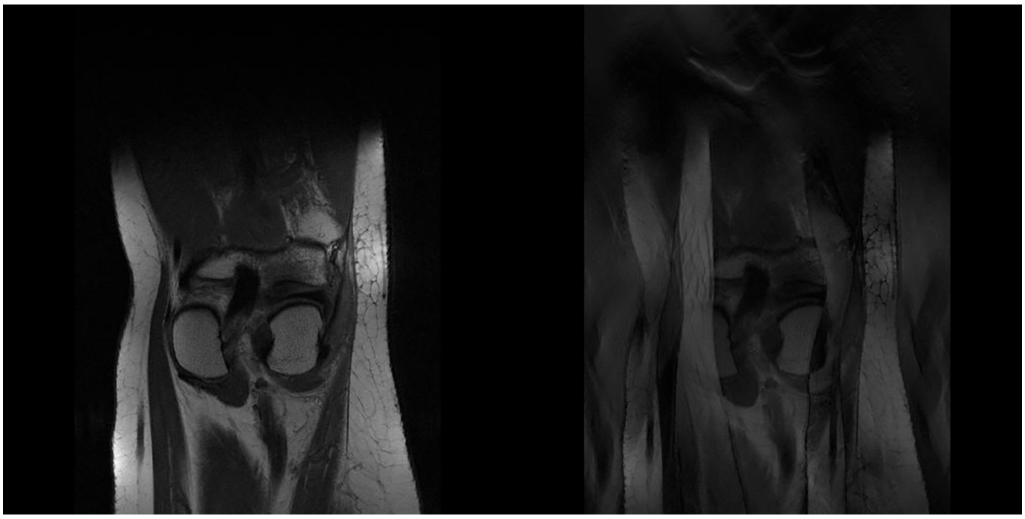

Fig. 4. Image from the dataset used by the GAN

Note: The image is separated into two parts: on the left, the image made with fully sampled data, and on the right, the image made with undersampled k-space.

2.3 GAN model architecture

Once preprocessing is complete, the image enhancement phase begins with the GAN. As mentioned previously, the GAN model is separated into two distinct neural networks: a generative network and a discriminative network. The generative network was based on the U-Net model, used in pix2pix algorithms. The U-Net model is a convolutional neural network model widely used in medical image segmentation because it is efficient at extracting information from images even with a limited number of samples. The U-Net architecture consists of an encoder that reduces spatial resolution, emphasizing image features (downsampling), and a decoder that increases image resolution while preserving the highlighted features (upsampling) [15].

The discriminative network is a convolutional classification model that receives several images containing pairs of concatenated images, as shown in Figure 4, which must then be classified. There are two types of pairs: The first type of pair contains the image made from undersampled data, together with the image made from fully sampled data, called the target image. This first pair can also be called a true pair because it contains two images extracted from the original data. The second type of pair consists of the image made from fully sampled data and the image generated by the generative model, called a false pair because it contains an image created by the generative model. The discriminative network then assigns values to its classifications, ranging from 0 to 1, with zero representing a false image pair and one representing a true image pair.

2.4 Evaluation metrics

Both models are trained to minimize their errors through the loss metric. In other words, the discriminator network seeks to increase the number of correct classifications, and the generative network seeks to increase the discriminator network's errors. Therefore, the GAN is trained with a weighted loss.

Images generated by the GAN must be evaluated to see if the result is similar to an image made with fully sampled data. Two values are used to indicate image quality: the structural similarity index (SSIM) and the signal-to-noise ratio (SNR).

The structural similarity index indicates the similarity between two images where one is given as a standard and the second receives a value from 0% to 100% similarity [16]. The image used as a standard is the image made with fully sampled data, and the generated image is the image that will be compared.

$$SSIM(x,y)=\frac{(2\mu_x\mu_y+C_1)(2\sigma_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)} \tag{1}$$

In the equation (1), μ_x and μ_y are the average pixel intensities in each of the images, σ_x and σ_y are the standard deviations of each image, σ_{xy} is the covariance between the two images, and C_1 and C_2 are two constants to balance the division, avoiding division by zero [16].

The signal-to-noise ratio shows the signal amplitude in relation to the noise amplitude contained in the image. The higher the SNR value, the greater the signal presence in relation to the noise. The area of interest method is used to calculate the SNR, where the center of the image is considered the signal and the edges are considered the noise [17].

$$SNR=10log_{10}\left(\frac{\sigma_s^2}{\sigma_n^2}\right) \tag{2}$$

In the formula, σ_s is the standard deviation of the amplitude values of the pixels contained within an area of interest in the center of the image, considered as the signal, and σ_n is the squared standard deviation of the amplitude values of the pixels at the edges of the image, considered noise.

3 RESULTS

The generative model has eight convolutional encoder layers that perform image downsampling. Seven transposed convolution layers, the first three of which are implemented with dropout, discarding half of their results. This forces the network to not rely entirely on the discarded portions, preventing the model from not generalizing the training data that is, overfitting [18]. Along with the transposed convolution layers, seven concatenation layers are implemented for the decoder, which perform upsampling together. Finally, a transposed convolution is performed, which increases image resolution. It also outputs three RGB channels and uses the tanh activation function, which normalizes the matrix values between –1 and 1.

The discriminative model has seven convolutional layers that perform downsampling. An L2 regularization layer penalizes very large weights in the network, forcing the model to have smaller weights to avoid overfitting [19]. As in the generator network, some of the downsampling layers apply dropout, which randomly deactivates half of the convolutional network units with each weight update. This helps the network avoid becoming completely dependent on those connections, thus avoiding overfitting and, in other words, better generalizing the training data.

The model has a layer that adjusts the size of the data by adding two layers of zeros to the data (zero-padding), followed by a convolution layer with 512 filters of size 4×4, skips of size 1, and L2 regularization.

Next, a batch normalization layer is used to normalize the previous layer to improve training stability, followed by a layer that applies the LeakyReLU activation function, which keeps positive values and reduces negative ones without canceling them, avoiding dead neurons and convergence problems.

Finally, another zero-padding layer is applied and then a final convolution that causes the data output to have very reduced dimensions, of size (None, 2, 2, 1), with the first dimension being the batch size, the second and third being the size of the final image, and the fourth, which contains only one value, the discriminator's verdict, which goes from 0 to 1, with zero being a pair of false images and one, a pair of true images.

A total of 110 different training runs were performed with different subsampling parameters and GAN configurations. The model was able to improve the quality metrics of the processed images in different test scenarios and for different parameters.

The GAN's configurable parameters included the number of epochs, buffer size, lambda, and the number of images that could be fed into the training, validation, and testing stages. To assess the GAN's performance, step vs. loss curves were extracted from the generative and discriminative models.

3.1 Quantitative results

To verify the improvement in these metrics, the ratio of the SSIM between the output and input (RSSIM) was analyzed, allowing us to see how many times greater the output SSIM is compared to the input. The best SSIM increase results obtained with each of the different trajectories can be seen in Table 1.

Table 1. Best SSIM increase obtained for each trajectory

Trajectory	k-Space Sampling (%)	SSIM (In)	SSIM (Out)	RSSIM (x)
Cartesian	50.39	0.53 ± 0.10	0.63 ± 0.09	1.18
Random Cartesian	30.07	0.23 ± 0.10	0.50 ± 0.04	2.12
Spiral 1	24.49	0.08 ± 0.04	0.38 ± 0.06	4.77
Spiral 4	24.84	0.29 ± 0.06	0.63 ± 0.08	2.18
Radial	6.14	0.53 ± 0.11	0.65 ± 0.09	1.24

The results containing the comparison of SNR values for the same tests in Table 1 can be seen in Table 2 along with the ratio between the output and input SNR (RSNR).

Table 2. Best results obtained for each trajectory (SNR)

Trajectory	k-Space Sampling (%)	SNR (In)	SNR (Out)	RSNR (x)
Cartesian	50.39	4.74 ± 3.13	7.43 ± 5.31	1.56
Random Cartesian	30.07	0.47 ± 2.33	6.98 ± 3.19	14.75
Spiral 1	24.49	2.33 ± 1.30	6.68 ± 3.15	2.87
Spiral 4	24.84	4.02 ± 2.15	7.84 ± 3.85	1.94
Radial	6.14	5.38 ± 3.38	7.40 ± 4.98	1.37

3.2 Qualitative results

Figure 5 shows an image produced by the test using the random Cartesian trajectory contained in the tables, compared to the input image and the reference image, made with fully sampled data (ground truth). The hyperparameters used for the test were batch size = 1, buffer size = 400, Lambda = 150, 50 epochs, Adam optimizer, and the loss function being binary Cross-entropy.

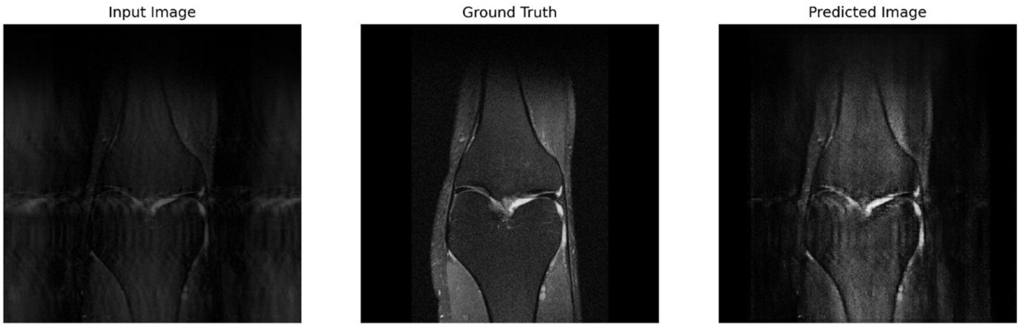

Fig. 5. Comparison between undersampled input, ground truth, and GAN-reconstructed images for the random Cartesian trajectory

Figure 6 shows a comparison of an image produced by the test performed with the radial trajectory of Table 2 compared to the input image and the ground truth image. The hyperparameters used for the test were batch size = 1, buffer size = 1500, Lambda = 150, 18000 epochs, Adam optimizer, and loss function being Binary Cross-entropy.

Fig. 6. Comparison between undersampled input, ground truth, and GAN-reconstructed images for the radial trajectory

4 DISCUSSION

After 110 tests, results demonstrating improvements in image quality metrics were obtained. Analyzing the results, it was possible to observe patterns in the quality metrics and image characteristics. Take the experiment performed with spiral trajectory 4 mentioned in Table 2 as an example. The average SSIM of the generated images was 0.63, but the individual metrics of each test image were analyzed, and the data are organized in the boxplot diagram in Figure 7.

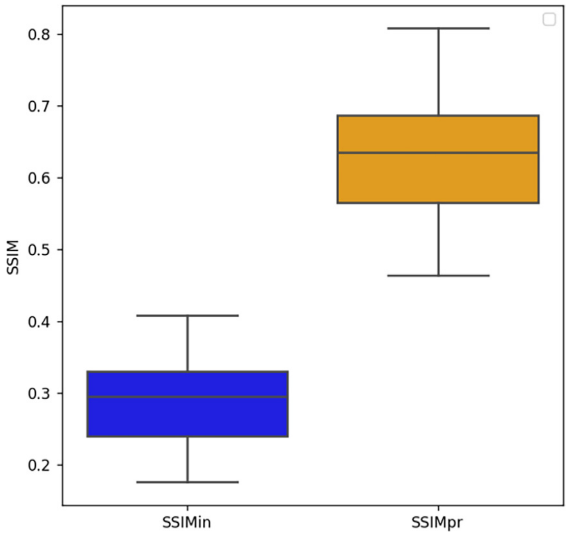

Fig. 7. Distribution of SSIM for input and GAN-generated images (Spiral-4 trajectory)

The boxplot shows that all images in the set were enhanced, and although the average SSIM of the enhanced images is 0.63, some images have an SSIM above 0.7. Figure 8 shows all the images that obtained an SSIM above 0.7.

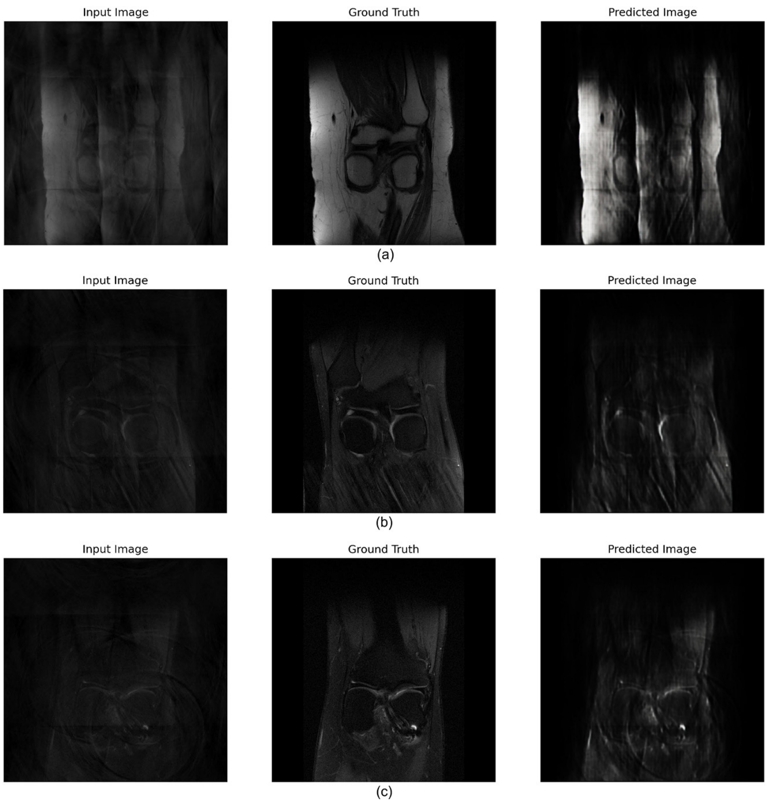

Fig. 8. Best images produced in the test with Spiral 4 trajectory from Table 2 showing input image, ground truth, and GAN reconstruction: (a) SSIM = 0.72; (b) SSIM = 0.80; (c) SSIM = 0.81

As with images (b) and (c), which were the best images produced in this test, the best images produced in other tests were also darker, with few clear details. This shows a pattern where darker images obtain higher SSIM values because, due to the fact that the images being compared have few details and low pixel amplitude, the generated images more closely resemble the reference images. However, these images have low SNR due to the low pixel amplitude. Considering that the average SNR for the test in Figure 8 was 7.84 dB, the three images with the highest SSIM achieved below-average results, having: (a) SNR = 4.50 dB, (b) SNR = 2.66 dB, and (c) SNR = 4.63 dB. This aspect highlights the importance of always analyzing SSIM and SNR together.

An example of the step x loss curve of the generative and discriminative models can be seen in Figure 9. This is the test curve with the spiral 4 trajectory in Table 2. It is possible to observe that at each epoch the loss of both models decreases, indicating that the models were refining without becoming unbalanced.

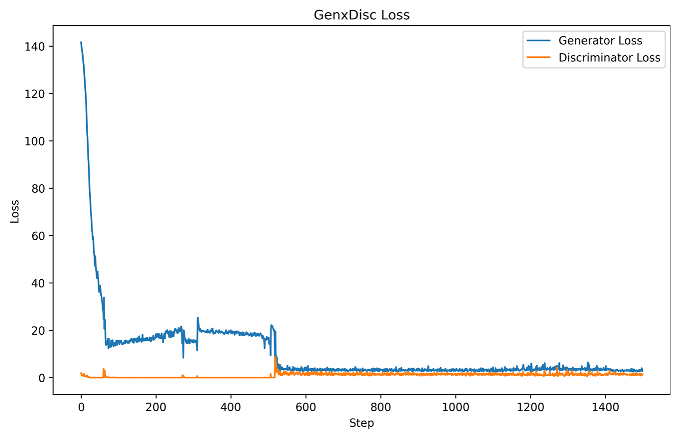

Fig. 9. Generator and discriminator loss as a function of training epochs for the spiral 4 test in Table 2

Another interesting point to note is the boxplot of the mean SSIM of the images generated in the tests, where there was an improvement in quality metrics, corresponding to 71 of the 110 tests performed. Figure 10 shows the boxplot of the mean SSIM of the input images and of the images generated by tests where the RSSIM was greater than one, that is, those in which the SSIM of the images improved.

In Figure 10, we can see that the input image metrics tend to be well dispersed, and after passing through the GAN, they tend to have a lower standard deviation. Furthermore, the algorithm tends to converge around the mean SSIM of 0.57, with input images with different SSIMs having an SSIM close to this mean after the GAN.

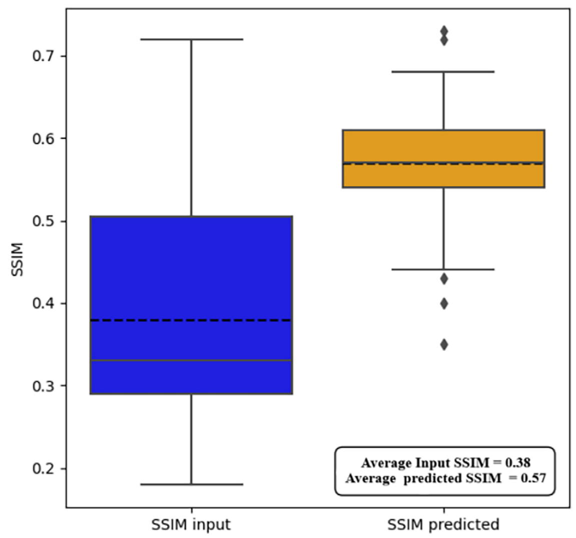

Fig. 10. Distribution of the mean SSIM of the input images (blue) and the mean SSIM of the generated images (orange)

The tests presented in Table 2 show that the trajectories that spared the lowest frequencies, which correspond to the center of k-space, obtained the highest SSIM values for the generated image, corresponding to the Cartesian, spiral 4, and radial trajectories. This occurs because most of the information in a k-space is contained in the low frequencies. This information is confirmed by observing the test with the radial trajectory, which spares the lowest frequencies, in which only 6.14% of k-space was spared, but the input image has an SSIM of 0.53. This behavior shows that the GAN achieves better results when trained with images that contain more information in the low frequencies of their k-space.

It's also interesting to note that the largest SSIM improvement was achieved with the Spiral 1 trajectory, increasing the value from 0.08 in the input image to 0.38. With a larger dataset, one could test the possibility of generating a new training dataset with this processing and training a new model with these images, thus investigating whether an even greater improvement would occur in the final images.

Another interesting point to note in the table with the best results is the increase in SNR for the random Cartesian trajectory test. Although it failed to achieve an SSIM above 0.6 like the best tests, the test with this trajectory managed to significantly reduce the input image noise, as shown in Figure 6, thus increasing the SNR from 0.47 dB to 6.98 dB.

The random Cartesian trajectory test can be considered one of the most successful, as it doubles the SSIM of the input image and drastically reduces the noise in the produced images, evident in the increased SNR. Figure 11 shows some more images from the test in question, performed using images created with data subsampled with the random Cartesian trajectory.

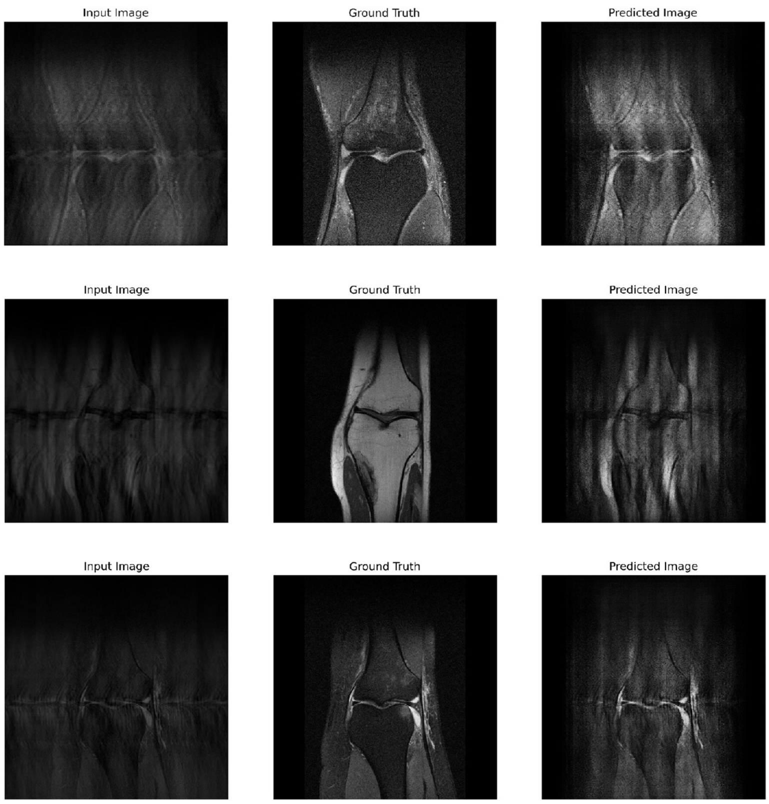

Fig. 11. Images produced by the test with the random Cartesian trajectory contained in Table 2

The results obtained show that the study could contribute to reducing the time of MRI scans due to the fact that it reconstructs images with fewer samples. The technology could be used to assist existing algorithms in achieving better results by aiding in image reconstruction.

5 CONCLUSION

This work implemented and analyzed a U-Net-based GAN to aid in the reconstruction of magnetic resonance images from sub-sampled data. The work utilized the FastMRI single-coil knee dataset and employed five different trajectories to sub-sample the k-space.

The results demonstrated consistent improvements in final image quality across all subsampling trajectories, especially in images created from trajectories that preserved low-frequency components (Cartesian, spiral 4, and radial), achieving the highest absolute SSIM values, while random Cartesian and spiral 1 trajectories showed the greatest relative increases in quality metrics. These findings confirm that GAN-assisted reconstruction can effectively reduce artifacts and increase structural similarity and signal-to-noise ratio even under high subsampling conditions.

The results demonstrate that GAN-assisted reconstruction can enhance magnetic resonance images obtained from undersampled data, preserving clinically relevant structures, validating its potential for future implementation in accelerated magnetic resonance acquisition systems. It also shows how different trajectories can influence the results obtained by the neural network and the importance of taking this information into account when designing this type of system.

6 ACKNOWLEDGMENTS

We thank the Brazilian National Council for Scientific and Technological Development (CNPq) for its financial support, which was essential for this research. This support significantly contributed to the development of the academic and scientific activities associated with this work.

7 REFERENCES

[1] OECD, "Health at a Glance 2023," 2023. [Online]. Available: https://www.oecd.org/en/publications/health-at-a-glance-2023_7a7afb35-en.html [Accessed: Sep. 10, 2025].

[2] S. N. Krieger, L. Huber, B. A. Poser, R. Turner, and G. F. Egan, "Simultaneous acquisition of cerebral blood volume-, blood flow-, and blood oxygenation-weighted MRI signals at ultra-high magnetic field," *Magnetic Resonance in Medicine*, vol. 74, no. 2, pp. 513–517, 2015. https://doi.org/10.1002/mrm.25431

[3] K. H. Mild, R. Lundström, and J. Wilén, "Non-ionizing radiation in Swedish health care—exposure and safety aspects," *International Journal of Environmental Research and Public Health*, vol. 16, no. 7, p. 1186, 2019. https://doi.org/10.3390/ijerph16071186

[4] D. Bos, N. Guberina, S. Zensen, M. Opitz, M. Forsting, and A. Wetter, "Radiation exposure in computed tomography," *Deutsches Ärzteblatt International*, vol. 120, no. 9, p. 135, 2023. https://doi.org/10.3238/arztebl.m2022.0395

[5] M. Zaitsev, J. Maclaren, and M. Herbst, "Motion artifacts in MRI: A complex problem with many partial solutions," *Journal of Magnetic Resonance Imaging*, vol. 42, no. 4, pp. 887–901, 2015. https://doi.org/10.1002/jmri.24850

[6] A. A. Mazzola, "Magnetic resonance: Principles of image formation and applications in functional imaging," *Revista Brasileira de Física Médica*, vol. 3, no. 1, pp. 117–129, 2009.

[7] A. D. Elster, "Methods for filling k-space," MRI Questions. [Online]. Available: https://mriquestions.com/k-space-trajectories.html [Accessed: Sep. 15, 2025].

[8] J. P. Felmlee, R. L. Morin, J. R. Salutz, and G. B. Lund, "Magnetic resonance imaging phase encoding: A pictorial essay," *Radiographics*, vol. 9, no. 4, pp. 717–722, 1989. https://doi.org/10.1148/radiographics.9.4.2756194

[9] Y. Cui, S. Wang, and R. Zhao, "Machine learning-based student emotion recognition for business english class," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 16, no. 12, pp. 94–107, 2021. https://doi.org/10.3991/ijet.v16i12.23313

[10] J. Montalt-Tordera, V. Muthurangu, A. Hauptmann, and J. A. Steeden, "Machine learning in magnetic resonance imaging: Image reconstruction," *Physica Medica*, vol. 83, pp. 79–87, 2021. https://doi.org/10.1016/j.ejmp.2021.02.020

[11] S. U. Dar, M. Yurt, M. Shahdloo, M. E. Ildız, B. Tınaz, and T. Çukur, "Prior-guided image reconstruction for accelerated multicontrast MRI Via generative adversarial networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 6, pp. 1072–1087, 2020. https://doi.org/10.1109/JSTSP.2020.3001737

[12] C. M. Hyun, H. P. Kim, S. M. Lee, S. Lee, and J. K. Seo, "Deep learning for undersampled MRI reconstruction," *Physics in Medicine & Biology*, vol. 63, no. 13, p. 135007, 2018. https://doi.org/10.1088/1361-6560/aac71a

[13] NYU Langone Health, "Welcome to the FastMRI dataset," n.d. Available: https://fastmri.med.nyu.edu/ [Accessed: Sep. 15, 2025].

[14] J. Zbontar *et al.*, "FastMRI: An open dataset and benchmarks for accelerated MRI," *arXiv preprint arXiv:1811.08839*, 2018.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference*, Munich, Germany, 2015, pp. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28

[16] A. N. Avanaki, "Exact global histogram specification optimized for structural similarity," *Optical Review*, vol. 16, pp. 613–621, 2009. https://doi.org/10.1007/s10043-009-0119-z

[17] A. D. Elster, "MR quality control: SNR. How do you measure signal-to-noise in an image?" n.d. [Online]. Available: https://mriquestions.com/signal-to-noise.html [Accessed: Sep. 17, 2025].

[18] TensorFlow, "tf.keras.layers.Dropout," n.d. [Online]. Available: https://www.tensorflow.org/api_docs/python/tf/keras/layers/Dropout [Accessed: Sep. 25, 2025].

[19] TensorFlow, "tf.keras.Regularizer," n.d. [Online]. Available: https://www.tensorflow.org/api_docs/python/tf/keras/Regularizer [Accessed: Sep. 25, 2025].

8 AUTHORS

Amauri da Costa Júnior. Master's degree in biomedical engineering from the University of Brasília, Brazil. Bachelor's degree in electronic engineering from the University of Brasília (E-mail: amauri_cj@hotmail.com).

Gerardo Antonio Idrobo Pizo. Associate professor at the University of Brasília, Brazil. Master's and PhD in Mechatronic Systems from the University of Brasília. Bachelor's degree in physics engineering from the University of Cauca, Colombia (E-mail: gerardo_idrobo@unb.br).

Cristiano Jacques Miosso. Associate professor at university of Brasília, Brazil. PhD in Electrical and Computer Engineering from the University of Texas at El Paso. Master's and bachelor's degrees in electrical engineering from the University of Brasília (E-mail: miosso@unb.br).