MRI Super Resolution

Aritro Roy, ar10067@nyu.edu

1 Introduction and background

Magnetic Resonance Imaging (MRI) is essential for brain imaging, with high-field scanners (e.g., 3T) providing high signal-to-noise ratio, sharp anatomical detail, and reliable contrast. However, these systems are expensive, infrastructure-intensive, and often inaccessible in low-resource settings. Low-field MRI (e.g., 64 mT) offers a more affordable and portable alternative, but produces images with lower resolution, reduced contrast, and higher noise. This creates a significant quality gap that limits clinical and research use.

This problem can be framed as a paired image enhancement task: learning a mapping from low-field MRI to high-field quality images. Earlier approaches relied on interpolation, filtering, or CNN-based super-resolution, but these methods often produced over-smoothed outputs or hallucinated structures. Recent advances in transformer-based encoder–decoder models for medical image restoration, such as AMIR, have demonstrated improved structural fidelity and robustness across a wide range of imaging tasks. Unlike earlier task-specific or diffusion-based approaches, AMIR Yang et al. 2024 uses a unified transformer architecture that learns shared anatomical representations while adapting to different restoration objectives. This design enables better generalization and more consistent preservation of anatomical structure, making it well suited for low-field to high-field MRI enhancement.

Historically, progress was limited by scarce paired datasets, insufficient computational power, and model limitations. Today, improved datasets, modern generative architectures, and accessible GPU resources make this challenge more tractable, enabling models that better preserve anatomical structure while enhancing image quality.

2 Datasets

2.1 Data Preprocessing

The dataset consists of paired low-field (64 mT) and high-field (3T) T1-weighted MRI volumes. Although the volumes are spatially aligned, they differ significantly in resolution, noise, and contrast. To ensure consistent model inputs, we applied the following preprocessing steps.

2.1.1 Volume Loading

MRI scans were loaded from NIfTI files using the `nibabel` library. Each subject contains a paired low-field and high-field volume.

2.1.2 Spatial Resampling

Low-field volumes have lower spatial resolution than their high-field counterparts. To align both volumes in the same coordinate space, the low-field scans were resampled to match the high-field volume shape using cubic interpolation. If V_{lr} is the low-field volume and V_{hr} is the high-field volume, the zoom factor along each axis is:

$$z_i = \frac{\text{shape}_{hr,i}}{\text{shape}_{lr,i}}$$

This produces a resampled low-field volume with the same dimensions as the high-field target.

2.1.3 Intensity Normalization

MRI intensities vary across scanners and acquisitions. Each volume was normalized independently using percentile-based normalization:

1. Compute the 1st and 99th intensity percentiles.
2. Clip values outside this range.
3. Scale intensities to the range $[0, 1]$.

This approach reduces the influence of extreme outliers and ensures stable training.

2.1.4 Slice Extraction

After resampling and normalization, each 3D volume was converted into 2D sagittal slices. For a volume of shape (X, Y, Z), slices were extracted along the sagittal axis:

$$\text{slice}_x = V[x, :, :]$$

This increases the number of training samples and reduces GPU memory requirements compared to full 3D training. Each slice pair is stored as:

- Input: low-field slice
- Target: corresponding high-field slice

2.2 Data Characteristics and Challenges

2.2.1 Noise and Blur

Low-field MRI exhibits:

- Higher noise levels
- Reduced contrast between tissues
- Blurred anatomical boundaries

2.2.2 Resolution Mismatch

The original low-field scans have:

- Lower in-plane resolution
- Thick slices along the axial direction

This leads to loss of fine structural details.

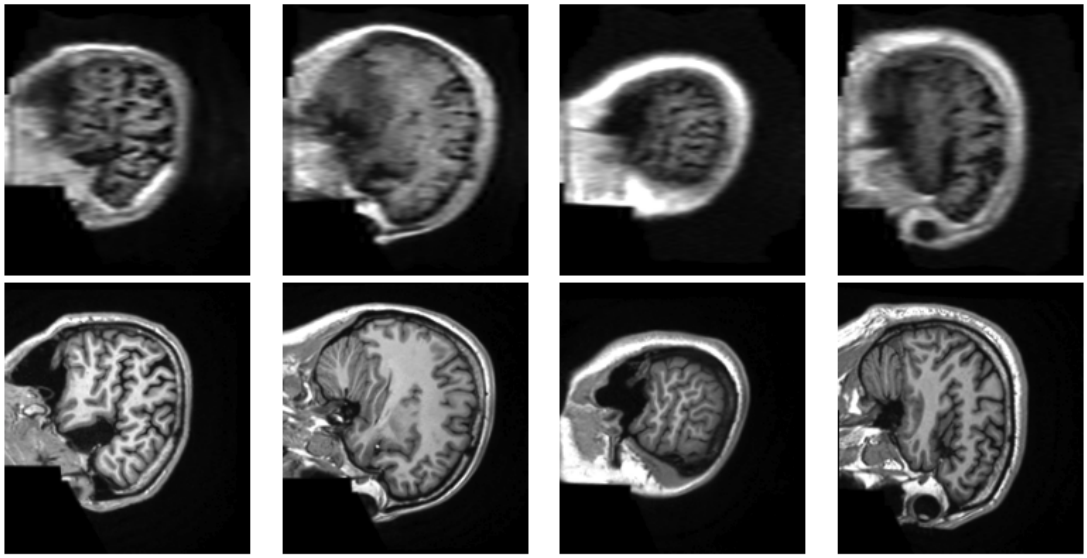
2.3 External Data and Synthetic Low-Field Generation

To increase the amount and diversity of training data, we incorporated external brain MRI scans from the IXI dataset.* In addition, we generated synthetic low-field data from the original high-field volumes in the training set.

2.3.1 Low-Field and High-Field Volume Reconstruction

For both low-field (LQ) and high-field (HQ) datasets, individual slices were provided as separate NIfTI files. These slices were grouped and reconstructed into full 3D volumes for each subject to ensure consistent spatial processing.

*https://brain-development.org/ixi-dataset/



Example training pairs from the dataset. Each column shows corresponding sagittal slices from the same subject. Top row: low-field (64 mT) input slices, exhibiting increased noise, reduced contrast, and blurred anatomical boundaries. Bottom row: corresponding high-field (3T) target slices, with higher signal-to-noise ratio, sharper tissue contrast, and clearer structural details. These pairs illustrate the quality gap the model aims to bridge during training.

Slice grouping and stacking The reconstruction process was identical for both domains:

1. Group slices belonging to the same patient using filename identifiers.
2. Sort slices according to their slice index to preserve anatomical order.
3. Load each slice and remove singleton depth dimensions if present.
4. Stack slices along the depth axis to form a 3D volume.

Low-field volume processing After reconstruction, each low-field volume was resampled to a consistent target shape: $(112, 138, 40)$ using cubic interpolation. A new affine matrix was assigned to match the low-field voxel spacing: $(1.6, 1.6, 5.0)$ mm. The processed volumes were saved in NIfTI format and used as low-field inputs during training.

High-field volume processing Similarly, each reconstructed high-field volume was resampled to a consistent target shape: $(179, 221, 200)$ using cubic interpolation. A new affine matrix was assigned to produce isotropic voxel spacing: $(1.0, 1.0, 1.0)$ mm. These volumes were saved in NIfTI format and used as high-field targets.

Outcome This reconstruction and resampling pipeline ensured:

- Consistent spatial dimensions across subjects.
- Proper anatomical slice ordering.
- Alignment between low-field inputs and high-field targets.

2.3.2 Synthetic Low-Field Generation

To further augment the training data, synthetic low-field volumes were generated from the original high-field scans. This process simulates the physical degradation observed in low-field MRI.

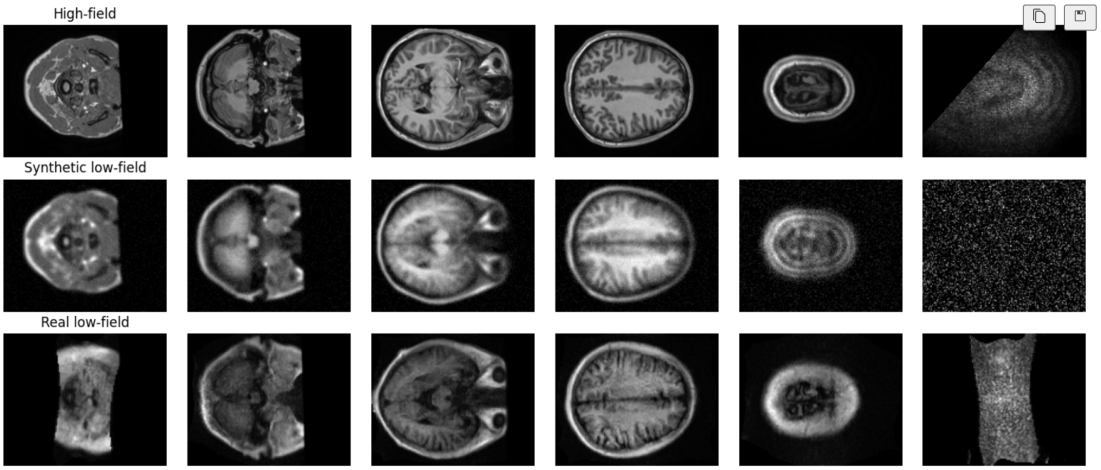
Each high-field volume was processed as follows:

1. Resample the volume to the target low-field voxel spacing $(1.6, 1.6, 5.0)$ mm.
2. Resize to the target shape $(112, 138, 40)$.
3. Apply Gaussian blurring to simulate reduced spatial resolution.
4. Darken intensities using a nonlinear intensity transform.
5. Add background-weighted Gaussian noise.

The synthetic low-field volume is defined as:

$$I_{low} = \text{clip}\left(\mathcal{N}\left(\left(G_\sigma(I_{high})\right)^\gamma\right), 0, 1\right)$$

where G_σ denotes Gaussian blurring, γ is the intensity darkening factor, and $\mathcal{N}$ represents additive noise.



Example slices from the training data showing real and synthetic low-field characteristics - Each column shows corresponding anatomical slices across three domains. Top row: high-field (3T) MRI slices used as ground-truth targets, exhibiting high signal-to-noise ratio, clear tissue contrast, and sharp anatomical boundaries. Middle row: synthetic low-field slices generated from the high-field volumes using the degradation pipeline, including spatial downsampling, Gaussian blurring, intensity darkening, and noise injection. Bottom row: real low-field (64 mT) MRI slices from the dataset, showing reduced contrast, increased noise, and structural blur. The comparison illustrates the domain gap between high-field and low-field acquisitions, as well as the visual similarity between real and synthetically generated low-field data used for training augmentation.

2.3.3 Benefits of External and Synthetic Data

The combination of real and synthetic data provides:

- Increased anatomical diversity.
- Improved robustness to noise and resolution variations.
- Reduced overfitting due to the small original dataset.

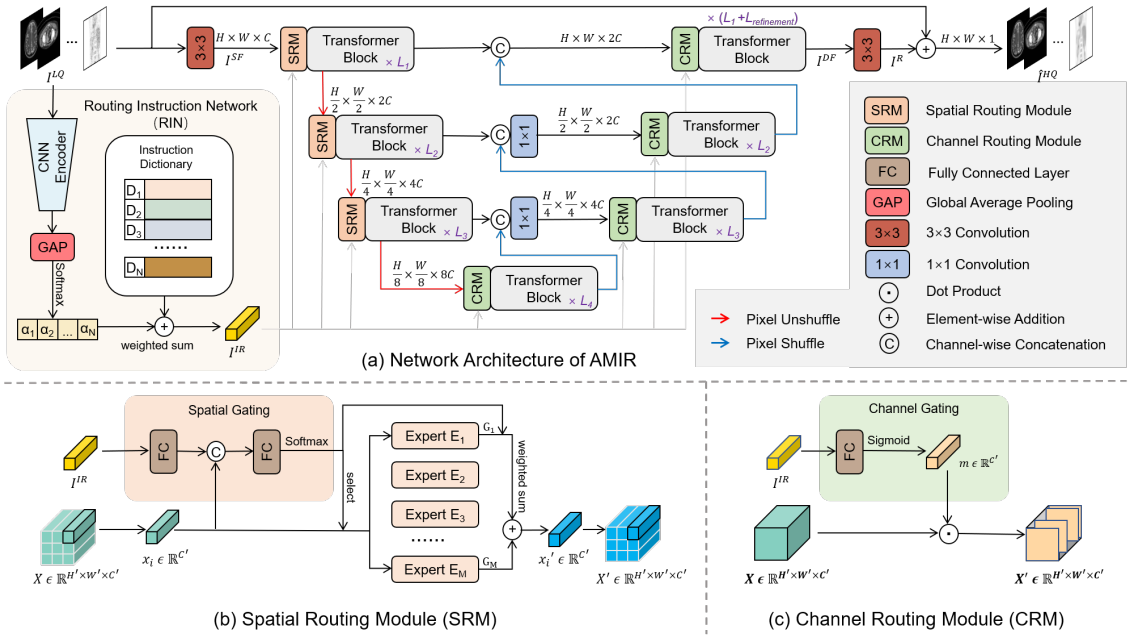
3 Methods

The goal of this work is to learn a mapping from a low-field MRI volume to its corresponding high-field counterpart. Given a low-field input volume V_{lr} and the corresponding high-field target V_{hr}, the model aims to learn a function:

$$\hat{V}_{hr} = f_\theta(V_{lr})$$

where f_θ is a neural network parameterized by θ. The objective is to minimize the discrepancy between the predicted high-field volume $\hat{V}_{hr}$ and the ground-truth volume V_{hr} using image similarity losses.

3.1 Proposed Method and Model Architecture



AMIR Model Architecture

We adopt the All-in-One Medical Image Restoration (AMIR) architecture Yang et al. 2024, a transformer-based encoder–decoder model with task-adaptive routing. AMIR is designed to handle multiple restoration tasks within a unified framework while preserving structural fidelity across different imaging conditions.

Architecture overview The overall architecture follows a U-shaped encoder–decoder structure composed of hierarchical transformer blocks. The encoder extracts multi-scale features from the low-field input, while the decoder reconstructs the enhanced high-field output using skip connections.

At each stage, AMIR employs routing mechanisms that dynamically select spatial and channel-wise feature transformations.

Routing Instruction Network (RIN) A Routing Instruction Network generates task-specific instructions from the input image. The network consists of a convolutional encoder followed by global average pooling and a softmax layer. The output is a weighted combination of learned instruction tokens:

$$I_{IR} = \sum_{i=1}^{N} \alpha_i D_i$$

where D_i represents instruction embeddings and α_i are softmax-normalized weights.

Spatial and channel routing AMIR uses two specialized routing modules:

- **Spatial Routing Module (SRM):** Selects spatial experts to adapt feature processing across different anatomical regions.
- **Channel Routing Module (CRM):** Applies channel-wise gating to emphasize relevant feature channels.

These modules dynamically adjust the transformer blocks, allowing the network to specialize its behavior for different restoration conditions.

Transformer backbone The core of the model consists of hierarchical transformer blocks that capture both local textures and long-range anatomical dependencies. Pixel shuffle and unshuffle operations are used to change spatial resolution between stages, enabling multi-scale feature learning.

Rationale Low-field MRI images suffer from noise, reduced contrast, and loss of high-frequency anatomical detail. Traditional CNN-based approaches often struggle to capture global context or may produce over-smoothed results.

The AMIR architecture was chosen for three main reasons:

- Transformer blocks capture long-range spatial dependencies, which is critical for preserving anatomical structure.
- Task-adaptive routing allows the model to adjust feature processing dynamically.
- The unified architecture improves generalization on small medical datasets.

These properties make AMIR well suited for the low-field to high-field MRI enhancement task.

4 Training Details

Model size The AMIR model used in our experiments contains approximately 22.3 million trainable parameters.

Data split After preprocessing and augmentation with external and synthetic data, the training set contained approximately 35,800 sagittal slice pairs. With a batch size of 4, this resulted in about 8,950 training batches per epoch.

The original dataset consisted of 18 paired subjects. Two subjects were held out for validation and one subject was reserved for testing, while the remaining subjects were used for training. To improve generalization, additional paired volumes were created using external IXI data and synthetic low-field generation, resulting in approximately 200 paired subjects in total.

The data were split at the subject level to prevent data leakage.

Training was performed on sagittal 2D slices extracted from the 3D volumes.

Training setup The model was trained using the following configuration:

- Architecture: AMIR
- Trainable parameters: 22.25M
- Batch size: 4
- Optimizer: Adam
- Initial learning rate: 2×10^{-4}
- Loss function: L_1 reconstruction loss
- Training duration: up to 60 epochs

Training was performed using PyTorch with shuffled mini-batches for the training set and sequential loading for validation.

Model selection Validation PSNR was monitored after each epoch, and checkpoints were saved periodically. The model with the best validation PSNR was selected for final evaluation.

5 Results

The trained model was evaluated on a held-out test subject from the original dataset. Performance was measured using peak signal-to-noise ratio (PSNR) computed on sagittal slices. The proposed model achieved the following PSNR statistics on the test set:

- Mean PSNR: 18.64 dB
- Median PSNR: 15.75 dB
- Standard deviation: 7.76 dB
- Minimum PSNR: 11.89 dB
- Maximum PSNR: 43.13 dB

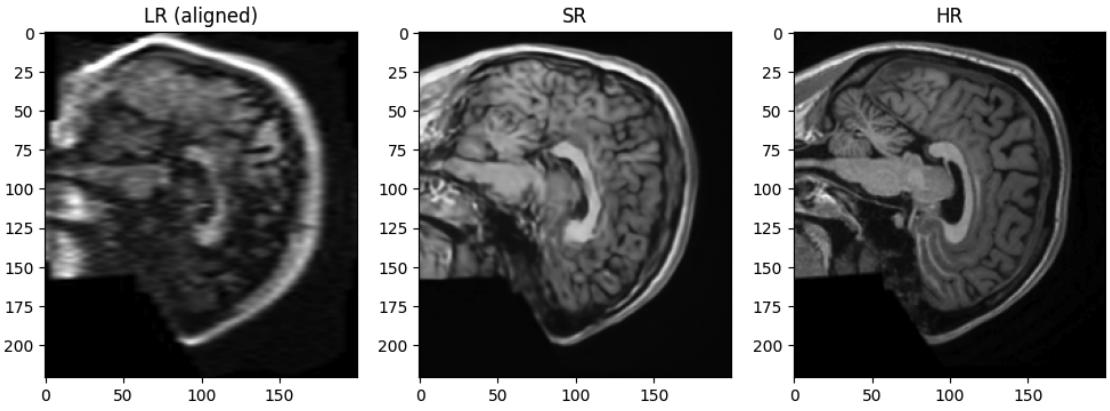
The difference between the mean and median PSNR indicates the presence of high-value outliers, which typically correspond to slices containing large background regions or low structural complexity. The relatively high standard deviation is expected due to slice-wise evaluation, where anatomical complexity varies significantly across slices.

Overall, the results demonstrate that the proposed AMIR-based model is capable of enhancing low-field MRI slices and recovering structural details, although performance varies across anatomical regions and noise levels.

Hyperparameter tuning Hyperparameters were selected based on validation PSNR. Preliminary experiments were conducted with different learning rates, batch sizes, and training durations. The final configuration was chosen as it provided the most stable convergence and the highest validation PSNR.

- Learning rates tested: 1×10^{-4}, 2×10^{-4}, 5×10^{-4}
- Batch sizes tested: 2, 4, and 8
- Training epochs tested: 40–80

The selected configuration (learning rate 2×10^{-4}, batch size 4, and 60 epochs) produced the best validation performance and stable training dynamics.



AMIR Model Reconstruction Qualitative example of low-field to high-field MRI restoration using the proposed AMIR model. From left to right: aligned low-field input (LR), super-resolved output produced by the model (SR), and the corresponding high-field ground-truth image (HR). The restored image shows improved contrast, sharper anatomical boundaries, and reduced noise compared to the low-field input, closely matching the high-field reference.

6 LLM Prompts

Data preprocessing pipeline

Prompt: "Give code to convert MRI slices into volumes and resample them to a fixed resolution."
Rationale: To build a consistent preprocessing pipeline for both high-field and low-field data, ensuring that all volumes share the same spatial dimensions and voxel spacing before training.
Result: The LLM generated a Python preprocessing script using the `nibabel` library to:

- Load individual MRI slices.
- Group slices into full 3D volumes.
- Resample each volume to a fixed target shape.
- Save the processed volumes as NIfTI files.

This script formed the basis of the data preparation stage.

Synthetic low-field data generation

Prompt: "How do I simulate low-field MRI from high-field MRI?"
Rationale: The original dataset contained only 18 paired subjects. Synthetic low-field data was needed to increase the diversity and size of the training set using external high-field MRI volumes.
Result: The LLM suggested a degradation pipeline consisting of:

- Spatial downsampling
- Gaussian blurring
- Intensity scaling
- Noise injection

This pipeline was used to generate synthetic low-field images from external high-field MRI volumes, increasing the dataset to approximately 200 paired subjects.

Visualization of model inputs and outputs

Prompt: "Give code to visualize low-field and high-field pairs from the dataloader."
Rationale: To verify that preprocessing, pairing, and augmentations were working correctly, and to ensure that input–target alignment was preserved.
Result: The LLM generated a visualization function that:

- Extracts a batch from the dataloader.
- Displays low-field inputs and corresponding high-field targets.
- Arranges paired slices in a grid for comparison.

This visualization was used to confirm correct data alignment before training.

7 Conclusions

In this project, we explored the task of enhancing low-field (64 mT) MRI scans into high-field–like (3T) images using a transformer-based medical image restoration model. The experiments showed that a unified transformer encoder–decoder architecture can learn structural mappings between low-field and high-field domains, even with limited paired medical data.

One of the most important findings was the impact of data preprocessing and augmentation. The original dataset contained only 18 paired subjects, which was insufficient for training a large transformer

model. By incorporating external high-field data and generating synthetic low-field counterparts, the dataset was expanded to approximately 200 paired subjects. This significantly improved training stability and reduced overfitting compared to training on the original dataset alone.

However, the project was constrained by limited computational resources. Although approximately 450 paired subjects were available after augmentation, only 200 could be used for training due to GPU and time limitations. Similarly, the model was trained for 60 epochs, which may not have been sufficient for full convergence of a transformer-based architecture. These constraints likely limited the final reconstruction quality.

Another key observation was that the model occasionally hallucinated structures that were not present in the original low-field input. This behavior is a known issue in image restoration models, especially when trained on small or heterogeneous datasets. The model may learn statistical priors from the high-field domain and incorrectly introduce anatomical details that appear plausible but are not anatomically accurate. This highlights the importance of careful loss design and evaluation beyond simple pixel-wise metrics.

There are several limitations in the current approach. First, the effective dataset size was still relatively small for a model of this capacity. Second, training was performed on 2D sagittal slices rather than full 3D volumes, which may reduce spatial consistency across slices. Third, the model was trained primarily with pixel-wise losses, which can encourage over-smoothing or hallucinated details. Finally, compute limitations prevented training on the full dataset and for longer schedules.

In future work, several improvements could be explored. Training on the full 450-subject dataset with longer schedules would likely improve performance. Using 3D or 2.5D inputs could enhance anatomical consistency. Incorporating structural or perceptual losses, such as SSIM-based or adversarial losses, may reduce hallucinations and produce more realistic reconstructions. Additionally, uncertainty-aware or diffusion-based restoration models could provide more reliable outputs for clinical settings.

8 Author contributions

This project was completed individually. The author was responsible for the entire pipeline, including data preprocessing, model design, implementation, training, evaluation, and preparation of the final report.

References

Yang, Z. et al. (2024). “All-In-One Medical Image Restoration via Task-Adaptive Routing”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 67–77.