Architectural Enhancements for Neural Attenuation Field Reconstruction in CBCT

(COSC440 Deep-Learning Project Report)

Wensheng Wu
University of Canterbury, Christchurch, New Zealand
wensheng.wu@pg.canterbury.ac.nz

July 28, 2025

1 Introduction

This report presents an architectural upgrade for cone-beam CT (CBCT) reconstruction. The baseline architectural is a multi-layer fully connected MLP whose input is the multi-resolution positional encoding of each 3D sample point produced by a Hierarchical Grid Positional Encoder (HGPE) [1]; it outputs the scalar X-ray linear attenuation coefficient μ at that spatial location. To enhance the reconstruction quality, I propose two key architectural improvements:

1. **Multi-Resolution Hash Grid Encoding:** We replace the original Hierarchical Grid Positional Encoder (HGPE) with a compact multi-resolution hash grid encoder to enhance the spatial encoding of 3D sample pointswith the objective of enhancingve both reconstruction accuracy and encoding efficiency.
2. **RAD-UNet with SE Attention:** The original MLP structure is replaced with a RAD-UNet encoder-decoder network, with residual connections and Squeeze-and-Excitation (SE) channel attention mechanism introduced in each convolutional module, referred to as **SE-RAD-UNet** in the remainder of this report.

Empirical evidence suggests that the proposed method achieves improved efficacy in terms of *Structural Similarity Index Measure (SSIM)* [2] and *Peak Signal-to-Noise Ratio (PSNR)* across most datasets, with faster convergence compared to the baseline MLP model with HGPE.

2 Experimental Design and Methods

2.1 Multi-Resolution Hash Grid Encoding

To enhance the spatial encoding capacity of the architecture, I replace the baseline Hierarchical Grid Positional Encoder (HGPE) with a custom-designed multi-resolution hash embedding encoder. The encoder is based on the Instant-NGP framework [3]. For each input 3D coordinate, trilinear interpolation is performed across 8 neighboring grid points

at each level to generate a smooth and continuous encoding. The final embedding vector is concatenated from the features of all layers. This design can fully express the spatial position features, thereby improving our reconstruction accuracy in CBCT reconstruction tasks.

Lack of TensorFlow-Compatible Hash Encoding. The hash grid encoding introduced in Instant-NGP [3] is implemented using the `tiny-cuda-nn` library [4], a CUDA-accelerated framework developed by the authors. While efficient and tightly integrated with PyTorch, `tiny-cuda-nn` does not offer native support or bindings for TensorFlow. To address this limitation, we implemented a custom HashGridEncoder in TensorFlow using XOR-based spatial hashing and trilinear interpolation. Due to the lack of low-level CUDA integration in our implementation, it runs on the CPU and exhibits significantly lower performance compared to GPU-accelerated alternatives.

The custom HashGridEncoder adopts a compact yet expressive spatial encoding mechanism based on spatial hashing and trilinear interpolation. Specifically, it employs $L = 16$ levels of hash tables, each corresponding to a 3D grid of resolution $R_l = R_0 \cdot 2^l$, where $R_0 = 16$ is the base resolution. Each grid vertex is associated with a learnable embedding vector of dimension $d = 2$, resulting in a total output embedding dimension of $L \cdot d = 32$. The encoder structure is summarized in Table 1.

Table 1: Key hyperparameters of the custom HashGridEncoder

Parameter	**Value**	**Description**
`input_dim`	3	Input spatial dimensionality (x, y, z)
`num_levels (L)`	16	Number of resolution levels
`level_dim (d)`	2	Dimensionality of embedding per level
`base_resolution` (R_0)	16	Grid resolution at the coarsest level
`log2_hashmap_size`	19	$T = 2^{19}$ hash slots per level
`output_dim`	32	Final embedding dimension ($L \cdot d$)
`primes`	$[1,\ 2^{28}+1,\ 2^{30}+5]$	Large primes used in the hash function

2.1.1 Advantages of HashGridEncoder over HGPE

Compared with HGPE, the hash grid encoder offers several advantages:

- **Hierarchical representation:** Although our implementation does not exploit true sparse memory structures, the multi-resolution hash table design conceptually enables efficient encoding across different spatial scales.
- **Local smoothness:** Using trilinear interpolation can result in smoother gradients and better training stability.
- **Multi-scale representation:** Features at different spatial scales are captured by independent levels, improving both global and local reconstruction fidelity.

2.2 SE-RAD-UNet

The baseline model in the project uses a simple multilayer perception (MLP) to predict attenuation coefficients at 3D spatial locations. Although this method is effective, it does

not fully exploit the spatial relationships between adjacent sample points on the ray, nor does it account for the inherent multiscale nature of radiographic features. To enhance the quality of reconstruction, I replaced the original multilayer perceptron (MLP) with a convolutional encoder-decoder network based on the U-Net framework. The proposed architecture, called RAD-UNet, combines residual connections and Squeeze-and-Excitation (SE) modules to enhance feature propagation capabilities and achieve channel-wise adaptive recalibration. The overall structure and data flow is illustrated in Figure 1. The detailed configuration of each block is listed in Table 2.

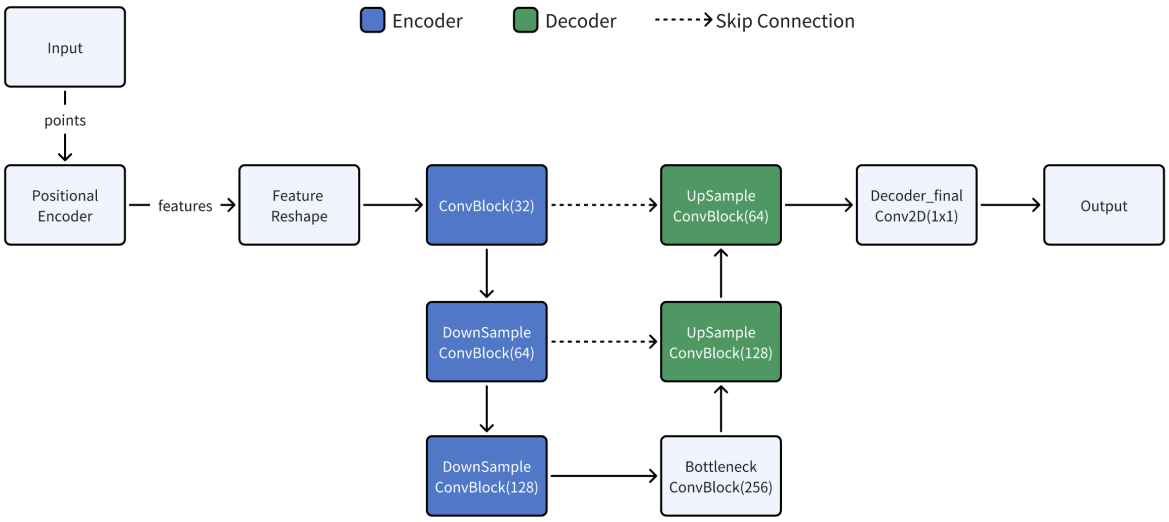

Figure 1: Network flow diagram of SE-RAD-UNet.

Table 2: Detailed layer configuration of SE-RAD-UNet components.

Component	**Channels**	**Description**
Input	-	Encoded 2D feature map from Hash / Positional Encoder
Encoder Block 1	32	ConvBlock with 2 convolutions, SE attention, and residual connection
Downsample Block 1	64	MaxPooling followed by ConvBlock(64)
Downsample Block 2	128	MaxPooling followed by ConvBlock(128)
Bottleneck	256	ConvBlock(256) with SE attention at the deepest level
Upsample Block 2	128	Transposed convolution, skip connection with Downsample Block 1, followed by ConvBlock(128)
Upsample Block 1	64	Transposed convolution, skip connection with Encoder Block 1, followed by ConvBlock(64)
Output Conv	1	Final 1x1 convolution to predict attenuation coefficient

Key changes include:

- **Convolutional Residual Blocks (`ConvBlock`):**
  Each block contains two 3×3 convolutional layers, followed by a SE attention mechanism and a skip connection to enhance gradient flow and preserve fine-grained spatial information. The internal structure is shown in Figure 2.

- **SE Attention Blocks(`SEBlock`):** Implements global average pooling followed by a bottleneck structure of two fully connected layers, allowing the model to adaptively recalibrate channel-wise feature responses. As illustrated in Figure 3, the SEBlock enables adaptive channel-wise recalibration.

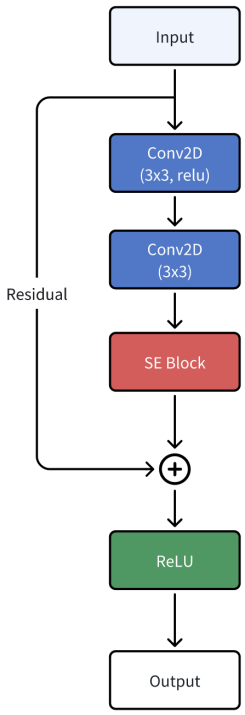

Figure 2: Structure of the ConvBlock.

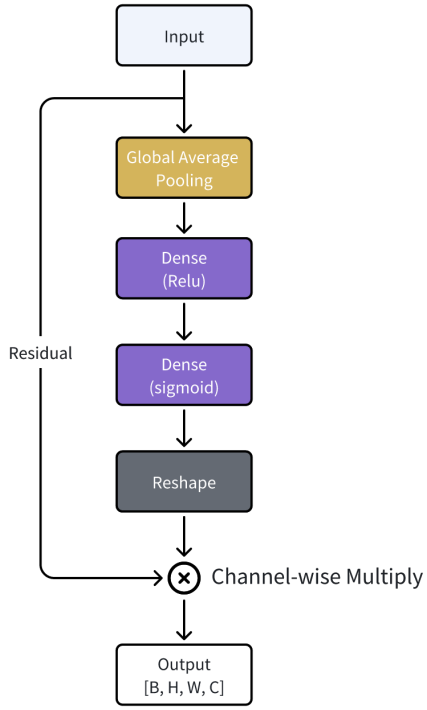

Figure 3: Structure of the SEBlock.

- **Downsampling & Upsampling:**
  Downsampling is performed using max-pooling followed by convolution, while upsampling uses transposed convolution followed by concatenation with encoder skip connections.

- **Model Replacement:**
  The `Baseline MLP` was replaced with `SE-RAD-UNet`, operating on reshaped feature maps.
  During training, features are reshaped to `[1, n_rays, n_points, C]`; during inference, the model automatically detects input size and reshapes to `[1, H, W, C]` accordingly.

- **Final Output:**
  The network predicts attenuation slices clipped to the `[0, 1]` range.

2.2.1 Architectural Advantages of SE-RAD-UNet over MLP

The proposed SE-RAD-UNet offers several theoretical advantages over the baseline MLP model:

1. **Exploitation of Spatial Context**: By processing the encoded points in a 2D grid structure, the network can leverage contextual information from neighboring points,

both along rays (depth) and across adjacent rays (lateral relationships), enabling the model to learn both depth-wise and lateral dependencies for better volumetric understanding.

2. **Multi-scale Feature Representation**: The encoder-decoder structure with skip connections enables the network to capture features at multiple scales simultaneously, which is particularly important for medical imaging where anatomical structures exhibit hierarchical organization.

3. **Channel Attention Mechanism**: SE blocks allow dynamic feature recalibration based on global context, improving focus on diagnostically relevant features, improving focus on diagnostically relevant regions by weighting informative channels more heavily.

4. **Residual Learning**: The residual connections facilitate gradient flow during training and enable the network to learn incremental refinements to the reconstructed volume, facilitating training and allowing the network to model residual errors for refined predictions.

2.3 Hypothesis and Expected Outcomes

Based on two key improvements, we propose the following hypothesis:

Hypothesis: These architectural changes are expected to enhance the spatial expressiveness of the model and improve its geometric generalization ability. Specifically, the HashGridEncoder capture features at different spatial through a compact encoding mechanism, while SE-RAD-UNet can exploit both depth-wise and lateral spatial dependencies using residual and attention-enhanced convolutional blocks.

Therefore, we hypothesize that these changes will lead to measurable improvements in the quality of CBCT reconstruction, as reflected by higher SSIM and PSNR metrics. It should be noted that since our HashGridEncoder implementation is CPU-based and lacks GPU acceleration, we do not anticipate faster convergence in wall-clock time. Instead, we expect improved learning efficiency within a fixed number of training epochs.

Expected Outcomes:

1. Improved reconstruction accuracy, reflected by higher SSIM and PSNR;

2. Faster convergence at the same number of epochs during training due to more stable gradients;

3. Better anatomical detail preservation from multi-scale features and channel attention;

4. Reduced noise/artifacts and more coherent volumes from spatial context modeling.

3 Experimental Setup

3.1 Hardware Configuration

All experiments were conducted on a laboratory computer with the following hardware specifications:

GPU: NVIDIA RTX 4090, 24 GB VRAM
CPU: Intel Core Ultra 9 Processor 285K
RAM: 62 GB

3.2 Software Environment

The software environment was managed using `conda`, with all experiments executed under the same isolated virtual environment. The following list summarizes the key system components and package versions used during model training (see Table 3).

Table 3: Software environment and key package versions

Component	Version
Operating System	Linux Mint 22 (based on Ubuntu 22.04 LTS)
Python	3.12.2
TensorFlow	2.18.0 (GPU-enabled)
CUDA Toolkit	12.5
cuDNN	9.8.0
NumPy	2.2.4

3.3 Training and Evaluation Setup

We experimented with four datasets: Chest, Jaw, Foot, and Abdomen. The model was trained for 1000 epochs with the Adam optimizer with a fixed learning rate of 1×10^{-3}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1 \times 10^{-7}$. Each training batch contained 2048 rays, with 192 equidistant sampling points per ray. Fixed validation set was used for evaluation after every 10 epochs. PSNR and SSIM were the primary metrics used for evaluating reconstruction quality.

The *HashEmbeddingEncoder* used 16 levels, each with an embedding dimension of 4, a base resolution of 32, and a hash map size of 2^{20} per level. The input dimension was 3.

All experiments were set to save the final tiff file at epoch 1000.

4 Results

4.1 Quantitative Results

Table 4: PSNR (dB) / SSIM results after 1000 epochs on four datasets. Higher is better.

	Chest	Jaw	Foot	Abdomen
MLP (HGPE)	29.4 / 0.87	32.2 / 0.81	29.4 / 0.85	22.9 / 0.77
SE-RAD-UNet (HashGridEncoder)	**33.9 / 0.93**	**34.8 / 0.88**	**31.2 / 0.87**	**21.3 / 0.84**

Table 4.1 presents the final PSNR (dB) and SSIM values after 1000 training epochs across four dataset. The proposed SE-RAD-UNet (HashGridEncoder) demonstrates consistent improvements over the MLP (HGPE) baseline in most cases.

Chest: PSNR increased significantly from 29.43 to 33.9, and SSIM from 0.87 to 0.93.

Jaw: The model achieved a PSNR of 34.8 and SSIM of 0.88, compared to 32.2 / 0.81 from the baseline.

Foot: Despite the presence of noise in the ground truth, the proposed model still outperformed the baseline with PSNR of 31.2 vs. 29.4 and identical SSIM of 0.87.

Abdomen: While the PSNR slightly dropped from 22.9 to 21.3, the SSIM improved markedly from 0.77 to 0.84, indicating better structural reconstruction despite a noisier signal.

4.2 Training Curves

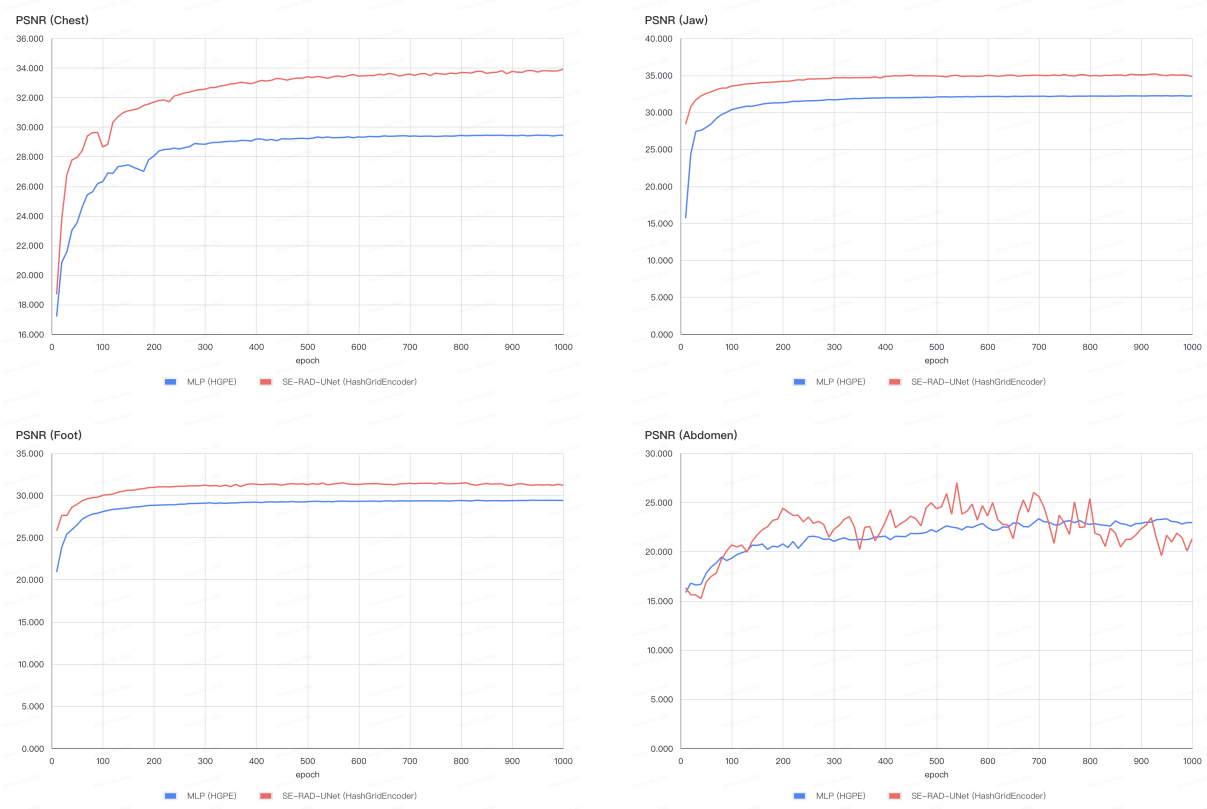

Figure 4: PSNR curves on the four dataset over 1000 epochs

Figure 4, Figure 5, and Figure 6 show the PSNR, SSIM, and MSE training curves of the four datasets under 1000 epochs of training.

Overall, the proposed SE-RAD-UNet (HashGridEncoder) not only achieves higher final scores on most datasets, but also shows faster convergence speed and more stable training performance, which has obvious advantages over the baseline model MLP (HGPE).

Notably, the Abdomen dataset exhibits larger PSNR fluctuations. However, the proposed architecture still achieves noticeably better SSIM, indicating improved structural reconstruction despite instability in intensity-based metrics.

These training curves demonstrate that SE-RAD-UNet (HashGridEncoder) not only enhances reconstruction quality, but also improves training dynamics across diverse anatomical regions.

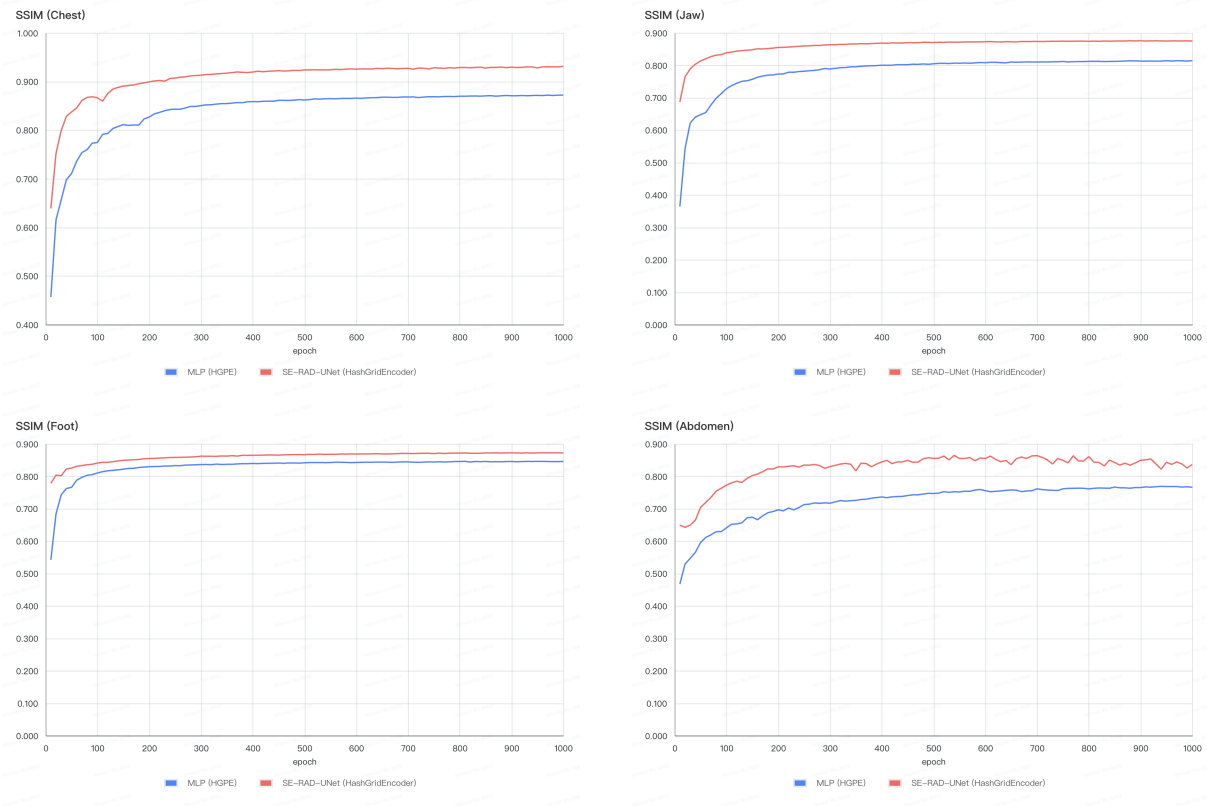

Figure 5: SSIM curves on the four dataset over 1000 epochs

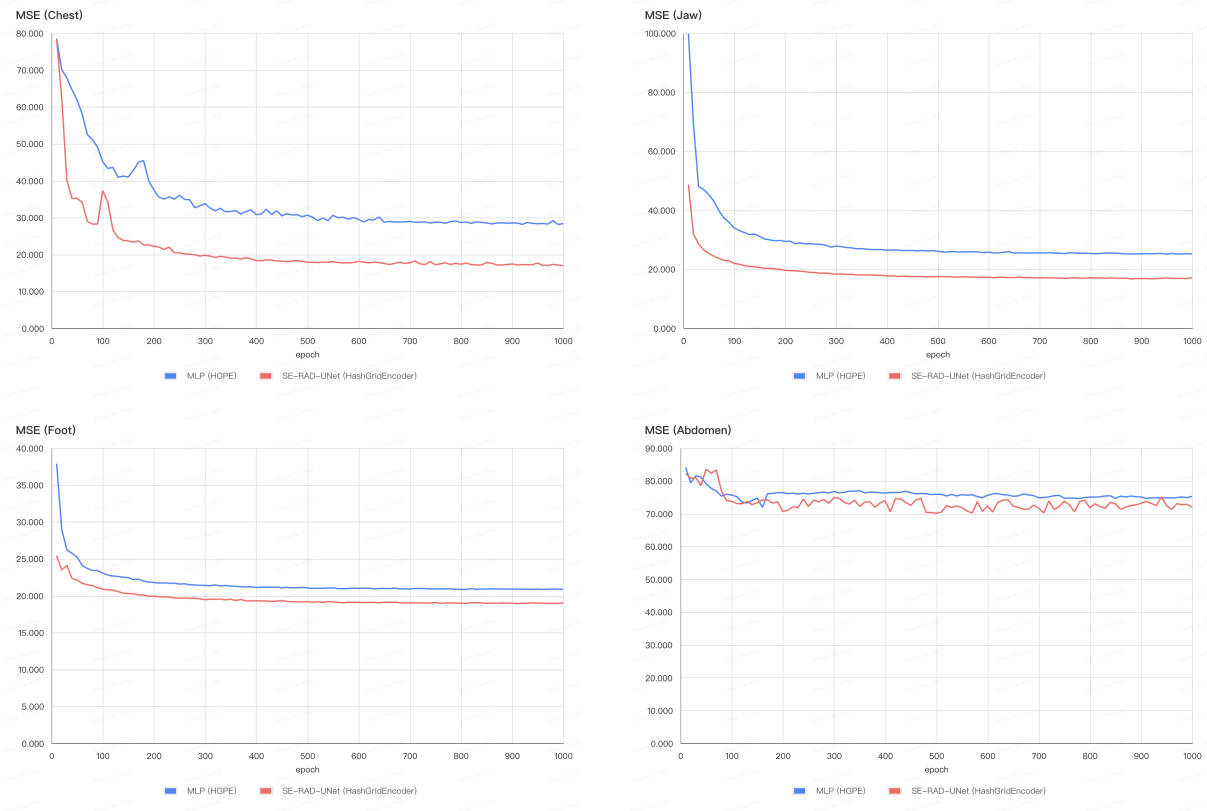

Figure 6: MSE curves on the four dataset over 1000 epochs

4.3 Analysis on the Abdomen Dataset

Make note of the Abdomen dataset. The Abdomen dataset, unlike others, displays certain instability with the PSNR and SSIM metrics simultaneously across 1000 training epochs, with PSNR showing even greater variability. Interestingly, the proposed architecture shows a drop in both metrics after about 550 epochs.

This phenomenon may be attributed to few causes:

- **Projection inconsistency caused by physiological motion**: The abdomen is susceptible to motion artifacts from respiration and intestinal peristalsis. These motions create gaps between adjacent CT projections due to the ground truth being either too noisy or mismatched. The network during training attempts to fit real anatomical signals against projection noise oscillation, which results in large PSNR fluctuations.

- **Low-gradient soft tissue dominance**: The abdomen is predominantly composed of continuous soft tissues with subtle intensity gradients. As PSNR is the logarithmic transformation of mean squared error (MSE), even minor prediction deviations in these low-contrast regions can cause a notable drop in PSNR—particularly when the model produces visually plausible but slightly blurred or over-smoothed reconstructions.

- **Limitations of hash grid encoding in low-contrast domains**: While multi-resolution hash grid encoding excels at modeling high-frequency and spatially sparse structures (such as teeth or spinal features), it provides limited benefit in large, low-contrast soft tissue regions. In fact, the high-dimensional encoding may introduce feature noise, amplifying pixel-wise errors and thus reducing PSNR, even though SSIM may still capture structural improvements.

Table 5: Effect of HashEncoder Configuration on Abdomen Dataset (500 epochs)

Configuration	**Num Levels**	**Base Resolution**	**Hashmap Size**	**SSIM**	**PSNR (dB)**
Original	16	32	20	0.855	24.35
Modified	24	16	21	0.852	23.76

Additional Experimentation on Encoder Configuration. To further investigate the performance instability on the Abdomen dataset, an additional experiment was conducted by modifying the resolution parameters of the *HashEmbeddingEncoder*. The original encoder used 16 levels with a base resolution of 32 and a hash map size of 2^{20}. The revised configuration increased the number of levels to 24, reduced the base resolution to 16, and expanded the hash map size to 2^{21}.

As shown in Table 5, the modified configuration did not lead to improved results. After 500 training epochs, the original configuration achieved a PSNR of 24.35 and SSIM of 0.855, while the revised setup resulted in slightly lower metrics: PSNR of 23.76 and SSIM of 0.852. These findings suggest that simply increasing encoder capacity does not mitigate the observed instability and may even degrade performance by introducing excessive representational complexity or feature noise.

4.4 Ablation Study: Effect of HashGridEncoder

To evaluate the effectiveness of the proposed *HashGridEncoder*, we conducted an ablation study comparing it with the original *HGPE* under two model configurations: MLP and SE-RAD-UNet. All other components of the architecture remained unchanged to isolate the effect of the encoder.

Table 6: Ablation study: PSNR (dB) / SSIM impact of model and encoder choice

Encoder	HGPE		HashGridEncoder	
Model	MLP	SE-RAD-UNet	MLP	SE-RAD-UNet
Chest	29.4 / 0.87	30.26 / 0.89	31.38 / 0.90	33.9 / 0.93

Figure 7: Ablation study on Chest dataset.

Table 6 and Figure 7 present the results of the ablation experiment. Whether in the baseline MLP or the SE-RAD-UNet architecture, introducing the HashGridEncoder can improve PSNR and SSIM.

However, under baseline MLP, obvious fluctuations will occur in the early stage of the training process. This instability may be attributed to the mismatch between the rich spatial coding capabilities of HashGridEncoder and the limited representational capabilities of shallow MLP, which leads to the architecture being more prone to oscillations during the initial learning phase.

In contrast, the SE-RAD-UNet architecture combined with HashGridEncoder demonstrates outstanding reconstruction performance. This indicates that stronger network architectures are better equipped to utilize the high-resolution spatial features provided by HashGridEncoder.

4.5 Training Efficiency Comparison

To evaluate the training efficiency, we recorded the total training time required for 1000 epochs of the two architectures. The results are summarized in Table 8.

As expected, due to the deeper and more complex structure of the SE-RAD-UNet architecture, its computational cost is higher than that of the simpler MLP baseline. Among all the configurations, the combination of SE-RAD-UNet and HashGridEncoder takes the longest time, which reflects the additional computational overhead brought

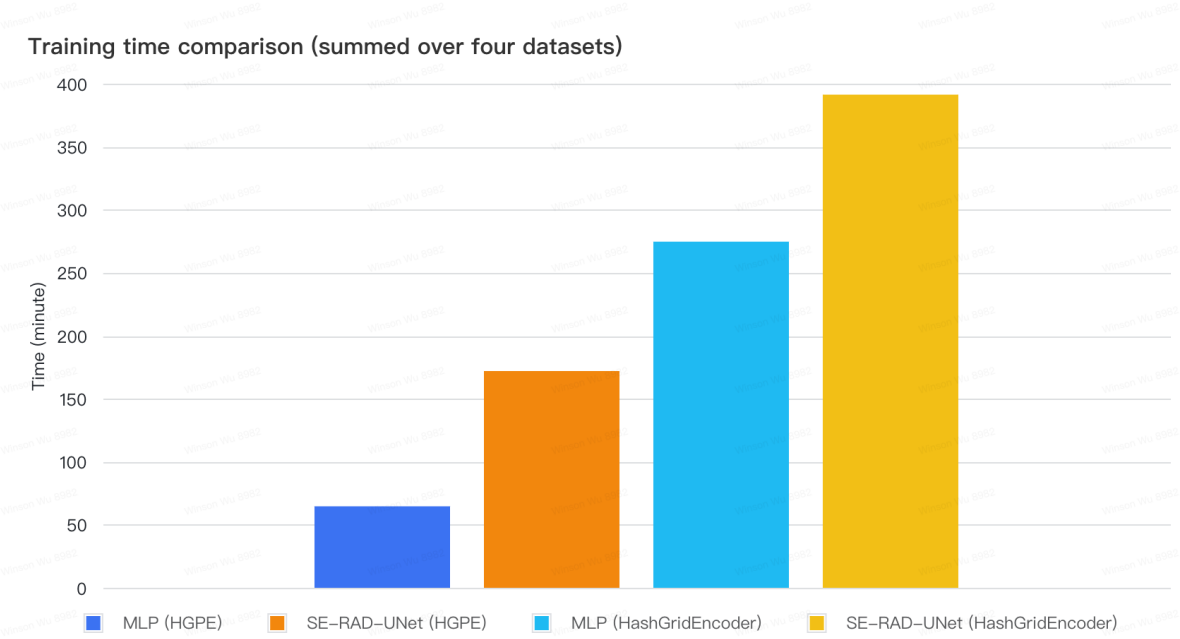

Figure 8: Training Time Comparison

about by the complexity of advanced encoders and network structures. On the other hand, MLP with HGPE is the lightest and has the fastest training speed. It is worth noting that switching from HGPE to HashGridEncoder significantly increases the training time of MLP and SE-RAD-UNet models. This indicates that although HashGridEncoder is beneficial for improving accuracy, it also brings additional training costs.

4.6 Visual Comparison

We also provide qualitative visualization results of different reconstruction methods in Fig. 9. The displayed slices are taken after 1000 training epochs. From left to right in each row: MLP (HGPE), SE-RAD-UNet (HashGridEncoder), Ground Truth.

5 Conclusions

5.1 Validation of the Proposed Hypothesis

The proposed SE-RAD-UNet (HashGridEncoder) demonstrates significant improvements over the baseline MLP (HGPE) in both reconstruction quality and training efficiency. Our hypothesis - that the introduction of convolution structure, multi-resolution spatial encoding, and channel-wise attention can improve the effective CBCT reconstruction effect - has been strongly supported by the experimental results.

- **Faster convergence speed:** As shown in Table 7, the SE-RAD-UNet (HashGridEncoder) can achieve comparable or better reconstruction quality to the baseline model in significantly fewer training epochs. For example, on the Jaw dataset, this model achieved the PSNR level of baseline MLP (HGPE) in 1000 epochs in just 50 epochs.

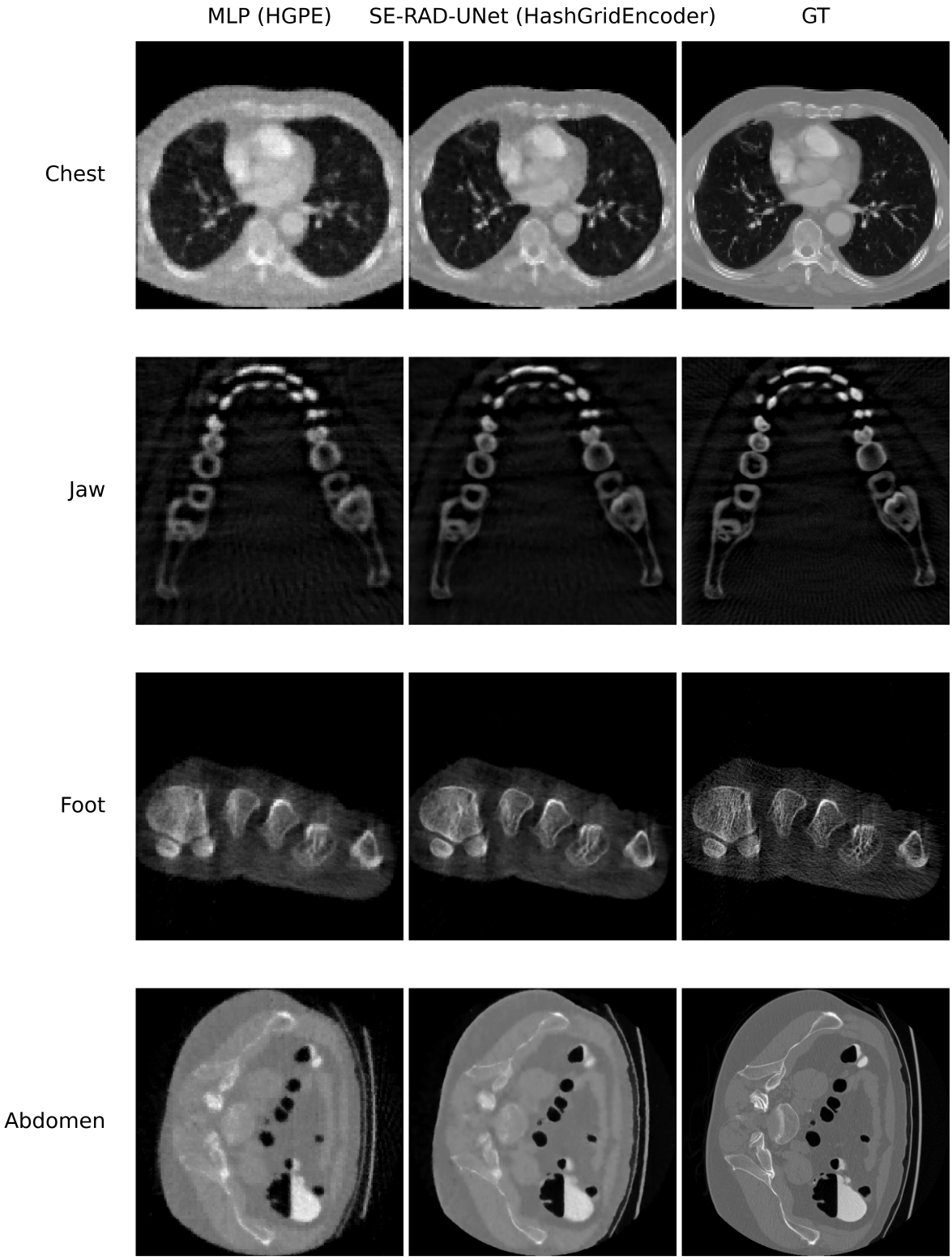

Figure 9: Visualization results of two methods.

Table 7: Epochs required to reach MLP (HGPE) PSNR/SSIM 1000 epoch levels for each dataset. Lower is better.

	Chest	**Jaw**	**Foot**	**Abdomen**
SE-RAD-UNet (HashGridEncoder)	**80 / 120**	**50 / 60**	**70 / 140**	**180 / 100**

- **Reconstruction quality Improvement:** Table 8 reports consistent gains in both PSNR and SSIM across all datasets after 1000 epochs.For example, on the chest dataset, our model outperforms the baseline by 4.6 dB in PSNR and 0.06 in SSIM.

Table 8: Comparison of final PSNR / SSIM performance at 1000 epochs. Higher is better.

	Chest	**Jaw**	**Foot**	**Abdomen**
MLP (HGPE)	29.4 / 0.87	32.2 / 0.81	29.4 / 0.85	22.9 / 0.77
SE-RAD-UNet (HashGridEncoder)	33.9 / 0.93	34.8 / 0.88	31.2 / 0.87	21.3 / 0.84
Improvement	**4.5 / 0.06**	**2.6 / 0.07**	**1.8 / 0.02**	**-1.6 / 0.07**

These results validate that the integration of multi-resolution hash grid encoding and a residual attention-enhanced convolutional architecture (SE-RAD-UNet) leads to superior CBCT reconstruction, combining both spatial encoding precision and structural learning efficiency.

5.2 Implications

The experimental results of this study provide several valuable insights into the design of neural network architectures in three-dimensional cone-beam CT (CBCT) reconstruction, particularly highlighting the limitations of traditional models and the effectiveness of the proposed method.

1. Limitations of MLP (HGPE)

The baseline multi-layer perceptron (MLP) combined with a hierarchical grid positional encoder (HGPE) - shows limited ability in modeling spatial dependencies inherent in volumetric data. Its point-wise prediction strategy lacks the ability to leverage contextual information across adjacent sample points, leading to fragmented reconstructions and suboptimal performance, particularly in anatomically complex regions. The experimental results show that shallow, fully connected networks is insufficient to meet the requirements of 3D reconstruction tasks with spatial continuity and structural consistency.

2. Effectiveness of HashGridEncoder

The HashGridEncoder captures local and global features at multiple scales, significantly enhancing the spatial representation capability, achieving significant improvements in PSNR and SSIM in most datasets. Although the current implementation is pure CPU due to TensorFlow limitations, its improvement in reconstruction quality shows strong application potential - especially when combined with GPU-accelerated frameworks such as `tiny-cuda-nn`, it has more promotion prospects. The experimental results verify that

HashGridEncoder is a powerful spatial coding strategy with good generalization ability, and is suitable for neural field-based reconstruction.

3. Effectiveness of SE-RAD-UNet

After replacing the MLP with the SE-RAD-UNet architecture, both the training dynamics and the final reconstruction quality have been significantly improved. The encoder-decoder design with skip connections supports multi-scale feature learning, while the integration of Squeeze-and-Excitation (SE) modules allows the model to adaptively focus on diagnostically relevant structures. This attention-guided refinement enhances anatomical fidelity and suppresses background noise.

6 Limitations and Future Work

- Due to the lack of a CUDA-accelerated HashGridEncoding implementation in TensorFlow, our custom HashGridEncoding implementation is purely CPU-based, resulting in prolonged training times. Therefore, we limited training to 1000 epochs per dataset. Future work may migrate the project to a PyTorch-based framework with CUDA-accelerated HashEncoder support for more efficient validation of model improvements.
- Inspired by the FACT method's use of meta-initialization and hash-encoding regularization [5], future work may consider incorporating optimization-based meta-learning strategies into our enhanced architecture, enabling rapid convergence even with extremely sparse-view CBCT data.
- The SE attention module has not been independently evaluated. Future work may conduct an ablation experiments to separate the contribution of SE attention and evaluate its impact on reconstruction quality, training stability, and convergence rate.

References

[1] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," 2020. [Online]. Available: https://arxiv.org/abs/2003.08934

[2] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[3] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics*, vol. 41, no. 4, p. 1–15, Jul. 2022. [Online]. Available: http://dx.doi.org/10.1145/3528223.3530127

[4] T. Müller, "tiny-cuda-nn: Lightning-fast neural networks on cuda," https://github.com/NVlabs/tiny-cuda-nn, 2021, accessed: 2025-05-10.

[5] H. Shin, T. Kim, J. Lee, S. Y. Chun, S. Cho, and D. Shin, "Fast and accurate sparse-view cbct reconstruction using meta-learned neural attenuation field and hash-encoding regularization," 2024. [Online]. Available: https://arxiv.org/abs/2312.01689