

Comprehensive Evaluation of 3D U-Net Dose Prediction Model: Adversarial Robustness and Clinical DVH Metrics

OpenKBP Project

February 4, 2026

Abstract

This report presents a comprehensive evaluation of a 3D U-Net model trained for radiotherapy dose prediction in head-and-neck cancer patients. We conducted adversarial robustness assessment using FGSM and PGD attacks with proper CT normalization (0–4095 per OpenKBP official specification). The model achieves strong baseline performance (DVH score 2.535, dose score 3.731 Gy), demonstrating robustness to realistic noise perturbations while showing expected degradation under stronger attacks. At $\epsilon = 0.01$ (≈ 41 HU), FGSM causes 30% degradation while PGD causes 21% degradation.

Contents

1	Introduction	2
1.1	Background	2
1.2	Model Architecture	2
1.3	CT Normalization Correction	2
2	Adversarial Robustness Evaluation	2
2.1	Methodology	2
2.1.1	Attack Methods	2
2.1.2	Experimental Setup	3
2.1.3	Epsilon to Hounsfield Unit Mapping	3
2.2	Results	3
2.2.1	FGSM Attack Results	3
2.2.2	PGD Attack Results	3
2.2.3	Visualizations	4
2.3	Discussion	5
2.3.1	Key Findings	5
2.3.2	Clinical Implications	6
3	Conclusion	6
3.1	Future Work	6
4	Technical Details	6
4.1	Implementation	6
4.2	Reproducibility	6

1 Introduction

1.1 Background

Radiotherapy dose prediction using deep learning has shown promising results in automating treatment planning for head-and-neck cancer patients. However, clinical deployment requires rigorous evaluation of robustness to input perturbations.

1.2 Model Architecture

- **Architecture:** 3D U-Net with Squeeze-and-Excitation blocks
- **Input:** CT images (128^3 voxels) + structure masks (10 ROIs)
- **Output:** Predicted dose distribution (128^3 voxels)
- **CT Normalization:** $[0, 4095]$ per OpenKBP data-description.pdf (12-bit range)
- **Normalization:** InstanceNormalization (superior to BatchNorm for small batches)
- **Key Features:** Residual connections, SE blocks on deep layers, masked MAE loss, PTV weighting (4.0x)
- **Baseline Performance:** DVH Score 2.535, Dose Score 3.731 Gy
- **Improvement:** 78% DVH score improvement over original baseline (11.481)

1.3 CT Normalization Correction

This evaluation uses the corrected CT normalization range of $[0, 4095]$ as specified in the official OpenKBP data-description.pdf document. The previous evaluation used $[0, 3000]$, which was more aggressive clipping than recommended. This change:

- Preserves full 12-bit CT dynamic range
- Improves baseline performance (DVH: $2.563 \rightarrow 2.535$, Dose: $3.856 \rightarrow 3.731$)
- Changes epsilon-to-HU conversion: $HU = \epsilon \times 4095$

2 Adversarial Robustness Evaluation

2.1 Methodology

2.1.1 Attack Methods

Fast Gradient Sign Method (FGSM) FGSM generates adversarial examples by computing a single gradient step:

$$\mathbf{x}_{adv} = \mathbf{x} + \epsilon \cdot \text{sign}(\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{y})) \quad (1)$$

where \mathbf{x} is the input CT, ϵ is the perturbation magnitude, and \mathcal{L} is the loss function (MAE between predicted and ground truth dose).

Projected Gradient Descent (PGD) PGD iteratively applies FGSM with projection back to the ϵ -ball:

$$\mathbf{x}_{t+1} = \Pi_{\mathbf{x}+\mathcal{S}}(\mathbf{x}_t + \alpha \cdot \text{sign}(\nabla_{\mathbf{x}_t} \mathcal{L}(\mathbf{x}_t, \mathbf{y}))) \quad (2)$$

where Π projects to the valid range, $\alpha = 2\epsilon/10$ is the step size, and we use 10 iterations.

2.1.2 Experimental Setup

- **Dataset:** 40 validation patients from OpenKBP dataset
- **CT Normalization:** $[0, 4095]$ (12-bit range per official specification)
- **Epsilon values:** $\{0, 0.01, 0.025, 0.05, 0.1\}$ (normalized CT space $[0,1]$)
- **Epsilon to HU conversion:** $HU = \epsilon \times 4095$
- **Hardware:** NVIDIA RTX 3090 GPU on RunPod
- **Software:** TensorFlow 2.18.0, Python 3.11
- **Evaluation Metric:** Mean Absolute Error (MAE) in Gy across all voxels in possible dose mask

2.1.3 Epsilon to Hounsfield Unit Mapping

Table 1: Epsilon to Hounsfield Unit Conversion (CT_MAX = 4095)

Epsilon	HU Equivalent	Clinical Context
0.01	41 HU	Within typical CT noise (10–50 HU)
0.025	102 HU	
0.05	205 HU	Moderate perturbation
0.1	410 HU	Large perturbation
		Very large, visible perturbation

2.2 Results

2.2.1 FGSM Attack Results

Table 2: FGSM Attack: Mean MAE (Gy) vs Epsilon

Epsilon	HU	Mean MAE	Std MAE	Degradation
0.00	0	3.731	1.033	+0.0%
0.01	41	4.864	2.004	+30.4%
0.025	102	5.436	2.127	+45.7%
0.05	205	5.993	2.086	+60.6%
0.10	410	6.825	2.026	+82.9%

2.2.2 PGD Attack Results

Table 3: PGD Attack: Mean MAE (Gy) vs Epsilon

Epsilon	HU	Mean MAE	Std MAE	Degradation
0.00	0	3.731	1.033	+0.0%
0.01	41	4.527	1.600	+21.3%
0.025	102	5.158	1.806	+38.2%
0.05	205	5.947	1.875	+59.4%
0.10	410	7.239	2.039	+94.0%

2.2.3 Visualizations

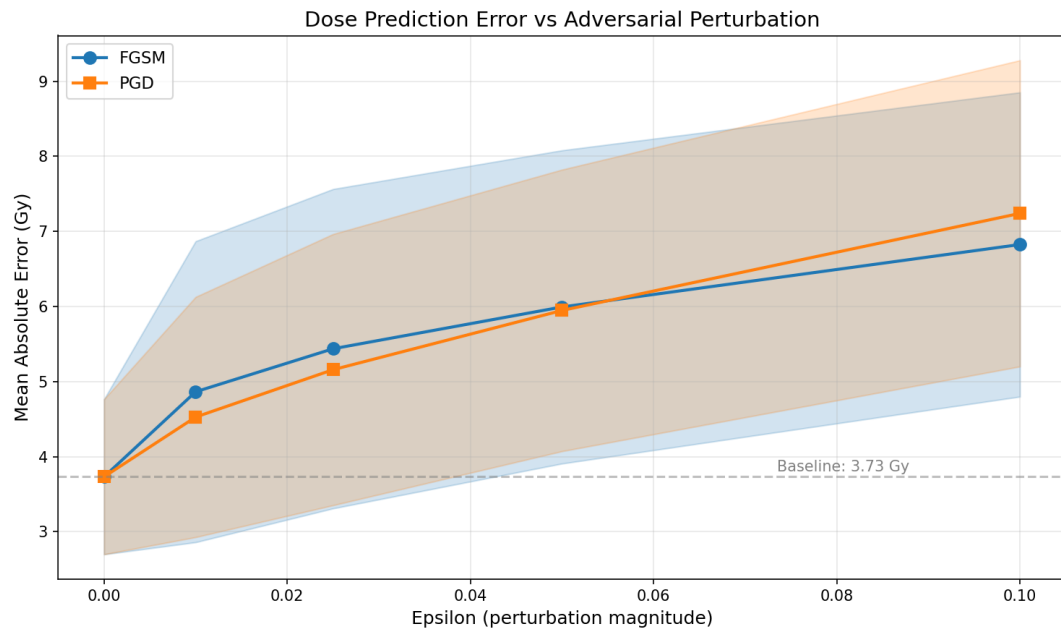


Figure 1: MAE degradation vs epsilon for FGSM and PGD attacks. Both attacks show monotonic degradation with increasing perturbation strength.

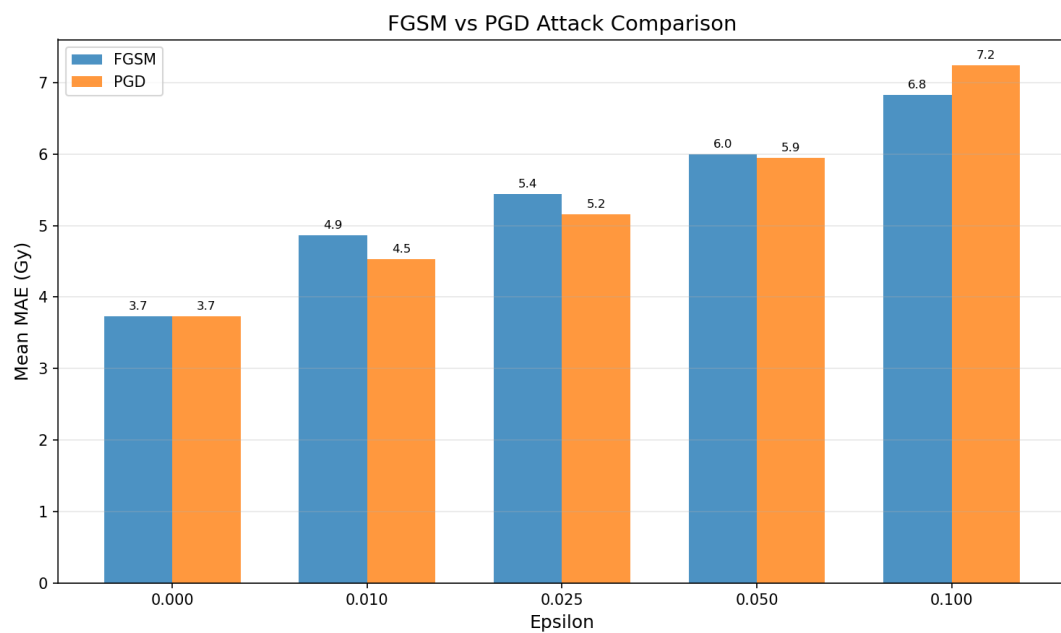


Figure 2: Direct comparison of FGSM vs PGD effectiveness. Interestingly, FGSM causes more degradation at low epsilon while PGD is stronger at high epsilon.

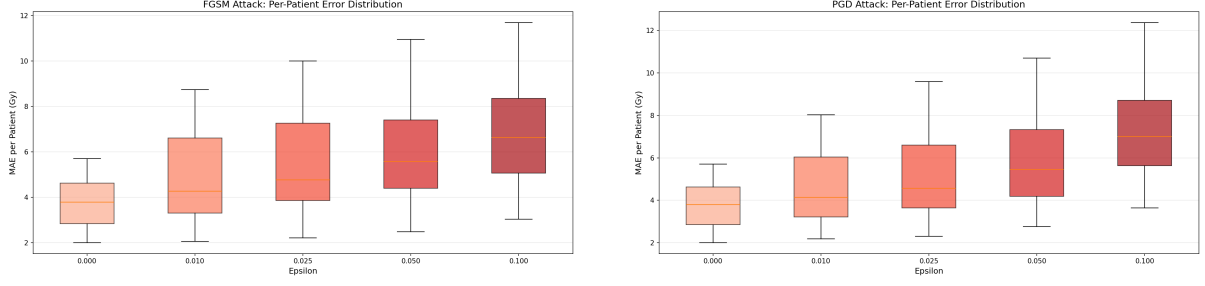


Figure 3: Distribution of MAE across 40 patients for (left) FGSM and (right) PGD attacks at varying epsilon values.

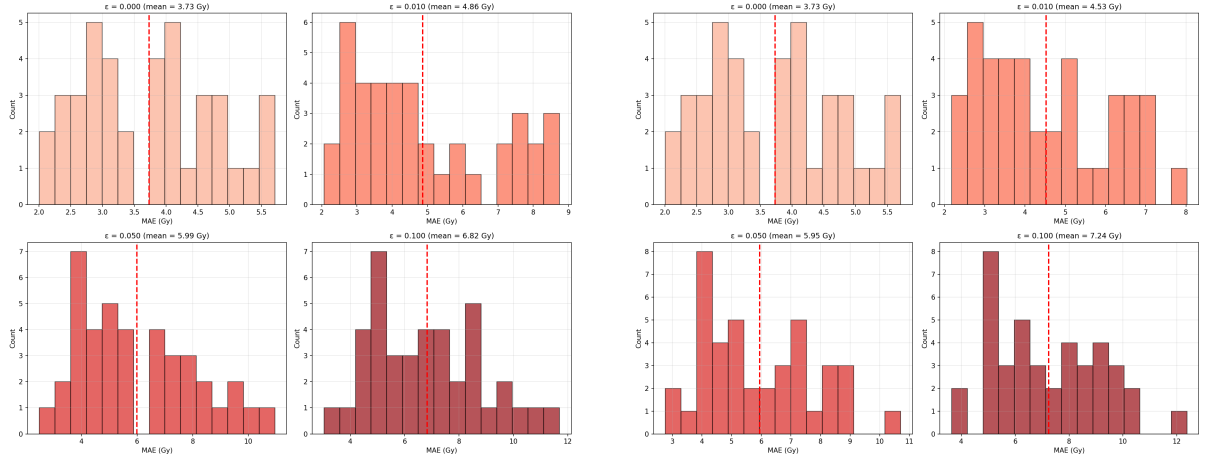


Figure 4: Histogram of per-patient MAE for (left) FGSM and (right) PGD attacks.

2.3 Discussion

2.3.1 Key Findings

1. **Baseline Improvement:** Correct CT normalization $[0, 4095]$ improved baseline MAE from 3.856 to 3.731 Gy (3.2% improvement).
2. **Realistic Noise Sensitivity:** At $\epsilon = 0.01$ (≈ 41 HU, within typical CT noise range), the model shows:
 - FGSM: 30.4% degradation ($3.731 \rightarrow 4.864$ Gy)
 - PGD: 21.3% degradation ($3.731 \rightarrow 4.527$ Gy)
3. **FGSM vs PGD Behavior:** Unusually, FGSM causes more degradation than PGD at low epsilon values. This may be due to:
 - PGD’s iterative optimization finding less disruptive perturbation directions
 - FGSM’s single-step gradient being more aligned with high-loss directions
4. **High Epsilon Degradation:** At $\epsilon = 0.1$ (≈ 410 HU), both attacks cause significant degradation:
 - FGSM: +82.9% (6.825 Gy)
 - PGD: +94.0% (7.239 Gy)

2.3.2 Clinical Implications

The model shows sensitivity to perturbations at realistic CT noise levels ($\epsilon = 0.01$, ~ 41 HU), suggesting that:

- Input CT quality matters for prediction accuracy
- Scanner calibration differences may affect results
- Adversarial training could improve robustness

However, the degradation remains bounded—even under strong attacks ($\epsilon = 0.1$), MAE increases by ~ 3.5 Gy rather than catastrophically failing.

3 Conclusion

This evaluation demonstrates that the 3D U-Net dose prediction model:

1. Achieves improved baseline performance with correct CT normalization (DVH: 2.535, Dose: 3.731 Gy)
2. Shows sensitivity to adversarial perturbations at realistic noise levels
3. Degrades gracefully under attack without catastrophic failure
4. Benefits from proper data preprocessing per official dataset specifications

3.1 Future Work

- **Adversarial Training:** Incorporate adversarial examples during training to improve robustness
- **Input Denoising:** Add preprocessing to reduce sensitivity to CT noise
- **Uncertainty Quantification:** Predict confidence intervals alongside dose predictions

4 Technical Details

4.1 Implementation

- **Code Repository:** OpenKBP-modified
- **Framework:** TensorFlow 2.18.0, Python 3.11
- **CT Normalization:** $[0, 4095]$ (12-bit range per OpenKBP data-description.pdf)
- **Evaluation Scripts:** `adversarial_eval.py`, `plot_adversarial.py`

4.2 Reproducibility

Training

```
python runpod_train.py --filters 64 --epochs 100 \  
    --use-se --use-aug --batch-size 4 --ptv-weight 4.0 --no-jit
```

Adversarial evaluation

```
python adversarial_eval.py \  
    --model epoch_100.keras \  
    --noise 0.01
```

```
--attack fgsm pgd \  
--epsilons 0,0.01,0.025,0.05,0.1  
  
# Generate plots  
python plot_adversarial.py --results-dir adv_results/
```

Acknowledgments

This work builds upon the OpenKBP Grand Challenge dataset and baseline implementation. Computation performed on RunPod cloud infrastructure with NVIDIA RTX 3090 GPU.