Reinforcement learning enhanced flow matching for reference-based tumor generation on unpaired CT images

Xiaoxuan Gong[1,2,3]* **Zhilin Zheng**[2,3,*†] **Tiancheng Lin**[2,3]* **Qinji Yu**[2,3]*
Jianpeng Zhang[2,3] **Shaoteng Zhang**[2,3] **Kai Cao**[4] **Xiaoli Yin**[5]
Yu Shi[6] **Jie Ma**[1†] **Ling Zhang**[2] **Yingda Xia**[2]
[1]School of Artificial Intelligence and Automation, HUST
Huazhong University of Science and Technology, Wuhan, China
[2]DAMO Academy, Alibaba Group [3]Hupan Lab
[4]Shanghai Institution of Pancreatic Disease
[5]Shengjing Hospital of China Medical University
[6]The Second Affiliated Hospital of Zhejiang University School of Medicine

Abstract

The success of deep learning in medical image analysis is often hindered by the long-tail distribution of disease cases, where rare or early-stage tumors are critically under-represented. While controllable synthesis offers a potential solution, existing methods suffer from a *specification bottleneck*, as low-dimensional signals like text or binary masks fail to capture complex textures and morphological variations. In this paper, we propose an exemplar-driven synthesis framework that utilizes real tumor samples as high-dimensional templates to enable precise, case-specific generation. To the best of our knowledge, this is the first work to introduce reinforcement learning (RL) into the field of 3D medical image generation, called **MedGRPO**. We formulate the unpaired tumor synthesis task as a constrained trajectory optimization problem and leverage Flow-GRPO to find an optimal generation path. By designing a multi-objective reward system encompassing texture, shape, and semantic consistency, our model effectively balances pathological fidelity with anatomical coherence without requiring paired supervision. Rigorous evaluations across four major tumor types and the external AbdomenAtlas 2.0 dataset demonstrate that our method produces high-fidelity synthetic data that significantly enhances the performance of downstream diagnostic models.

1 Introduction

The success of deep learning in medical image analysis depends heavily on the availability of large-scale, annotated datasets [1, 2, 3, 4]. However, a fundamental challenge remains: the long-tail distribution of disease cases [5, 6]. While common lesions are easy to access, early-stage or rare tumors are critically under-represented, yet they are the most vital for clinical diagnosis [7, 8]. To alleviate this data scarcity, controllable tumor synthesis in 3D Computed Tomography (CT) has emerged as a key tool for generating diverse samples to complement training data [9, 10, 11]. Current methods generally follow two paradigms: (1) global conditional generation, which uses spatial priors to guide the overall structure [12, 13], and (2) specialized localized inpainting, which either trains separate models for specific tumor types [14, 15] or relies on coarse signals like textual prompts [16]

*Equal contribution.
†Corresponding authors (Email: zhilinzheng@foxmail.com and majie@hust.edu.cn)

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

and histogram-based statistical controls [15]. Despite their utility, these approaches suffer from a fundamental *specification bottleneck*: such sparse control signals are inherently underspecified, providing insufficient detail to capture the fine textures and complex shape variations of real-world pathologies presented in CT scans. Consequently, the model tends to fill in this informational gap with smooth or average patterns, failing to achieve clinical realism.

To overcome this, we propose an exemplar-driven synthesis framework. Our core insight is that instead of asking a model to imagine complex textures from a simple word or mask, we can use a real tumor sample (an exemplar) as a high-dimensional template. By using a specific exemplar as a detailed prompt, we enable precise control over the generation process. The paradigm allows for the direct transfer of unique clinical features from a rare real case into various anatomical contexts. This ensures that even for the rarest pathologies, we can preserve their authentic textures and shapes while making sure they fit naturally into the target organ.

The major technical challenge in this approach is the lack of paired data (i.e., we do not have images of the same tumor in different patient contexts). Naive "copy-paste" or simple blending inevitably introduces boundary artifacts and anatomical inconsistencies [17, 18]. To address this, we introduce **MedGRPO**, which, to the best of our knowledge, is the first framework to employ reinforcement learning (RL) for 3D medical tumor synthesis. We formulate this challenge as a constrained trajectory optimization problem. Specifically, we leverage Flow-GRPO [19], adapting Group Relative Policy Optimization for flow-matching models [20], to find an optimal generation path that balances two competing objectives: pathological fidelity to the exemplar and anatomical coherence within the host organ. To provide precise guidance, we design a multi-objective reward system comprising texture, semantic, and shape consistency rewards. This allows the model to contextually integrate the exemplar tumor with the target anatomy, ensuring a seamless result that respects the underlying anatomical boundaries without requiring paired supervision.

We rigorously evaluate our framework on a curated large-scale dataset covering four major tumor types. Beyond perceptual realism, we demonstrate the clinical utility of our method through extensive downstream validation. By augmenting training sets with our synthetic tumors, we yield significant performance gains in tumor segmentation on the largest open-source dataset AbdomenAtlas 2.0 [10]. Our results suggest that exemplar-driven synthesis, optimized via reinforcement learning on flow trajectories, offers a robust solution to the data scarcity problem in medical AI.

Our main contributions are summarized as follows:

- We introduce an **exemplar-driven paradigm** for CT tumor synthesis, shifting the control mechanism from low-dimensional labels to high-dimensional pathological templates for precise, case-specific generation.
- We propose **MedGRPO**, the first reinforcement learning framework for 3D medical tumor synthesis. It optimizes generation trajectories via a **multi-objective reward system** to achieve a principled balance between preserving pathological fidelity and ensuring morphological adaptation to the target anatomy.
- Through rigorous evaluation across four major cancer types and the external AbdomenAtlas 2.0 dataset, we demonstrate that our synthetic data significantly enhances the performance of downstream diagnostic models.

2 Preliminary

2.1 Notation

For clarity, we summarize the main mathematical notations used throughout this paper in Table 1.

2.2 Conditional flow matching

Flow Matching (FM)[20] has emerged as a powerful simulation-free framework for training continuous-time generative models. Unlike traditional Diffusion Models that rely on stochastic differential equations, FM directly learns a time-dependent vector field to define a deterministic probability flow between a source distribution p_0 (e.g., Gaussian noise) and a target data distribution p_1 (e.g., CT images).

Table 1: Summary of the main notations used in MedGRPO.

Category	Symbol	Description
Flow Matching	x_0, x_1	Source and target samples
	t, p_t	Continuous time and intermediate probability distribution
	v_θ, u_t	Learned and target velocity fields
	ϕ_t	Probability flow induced by the velocity field
	σ_t	Diffusion coefficient for stochastic exploration
	ϵ	Standard Gaussian noise
MedGRPO	I, R	CT image and reference tumor exemplar
	$\mathcal{E}$	Frozen VAE encoder
	z_0, z_{ref}	CT and exemplar latent representations
	B, b	Tumor bounding box and its latent-space representation
	z_1, z_t	Partially corrupted and interpolated CT latents
	X_{in}	Concatenated input to the generation network
	$f_\theta, E_{\text{tumor}}$	U-Net predictor and tumor-type text embedding
	$v_{\text{pred}}, m_{\text{pred}}, M$	Predicted velocity, predicted mask, and ground-truth mask
Supervised Learning	$\mathcal{L}_v$	Velocity regression loss
	$\mathcal{L}_{\text{Dice}}$	Dice loss for tumor mask prediction
	$\mathcal{L}_{\text{SL}}, \lambda_{\text{Dice}}$	Overall SL objective and Dice loss weight
Reinforcement Learning	K	Number of sampled trajectories per condition
	$\mathcal{R}^k_\sigma, A^k_\sigma$	Reward and normalized advantage for category σ
	A^k	Aggregated advantage across reward categories
	S_ϕ, Ψ	Shape extractor and segmentation network
	H_B, I_{pred}	Histogram operator and generated CT image
	$\pi_\theta, \pi_{\text{old}}, \pi_{\text{ref}}$	Current, behavior, and reference policies
	$\rho_t, \varepsilon, \beta$	Importance ratio, clipping threshold, and KL weight

Probability flow and vector fields. Let $x \in \mathbb{R}^d$ be a sample in the image space. Conditional Flow Matching (CFM) aims to learn a time-dependent vector field $v_\theta(x,t) : \mathbb{R}^d \times [0,1] \to \mathbb{R}^d$, parameterized by a neural network θ. This vector field generates a flow ϕ_t that transforms samples from p_0 to p_1 according to the Ordinary Differential Equation (ODE):

$$\frac{d}{dt}\phi_t(x) = v_\theta(\phi_t(x), t), \quad \phi_0(x) = x_0 \sim p_0 \tag{1}$$

The push-forward of the probability density under this flow is denoted as p_t. While the marginal vector field $u_t(x)$ that generates p_t is typically intractable, CFM utilizes conditional probability paths to simplify the learning objective.

The CFM objective. By conditioning the probability path on individual data samples $x_1 \sim p_1$, the training objective can be formulated as a regression task:

$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t\sim\mathcal{U}[0,1], x_1\sim p_1, x\sim p_t(x|x_1)} \|v_\theta(x,t) - u_t(x|x_1)\|^2 \tag{2}$$

where $u_t(x|x_1)$ is the conditional vector field and $p_t(x|x_1)$ is the conditional probability path. It has been proven that the gradient of $\mathcal{L}_{\text{CFM}}$ is equivalent to the gradient of the marginal flow matching objective, allowing for efficient training without expensive simulations.

Optimal transport path. A prevalent choice for the conditional path in medical imaging tasks is the Optimal Transport (OT) displacement map, which defines a linear interpolation between the source and target:

$$p_t(x|x_1) = \mathcal{N}(x|tx_1 + (1-t)x_0, \sigma^2 I) \tag{3}$$

The corresponding conditional vector field $u_t(x|x_1)$ is simplified to a constant straight-line trajectory: $u_t(x|x_1) = x_1 - x_0$. Compared to the curved trajectories in Diffusion Models, this linear formulation enables the model to learn the shortest path between distributions. This property is particularly beneficial for high-resolution 3D CT generation, as it significantly reduces discretization errors and allows for high-fidelity synthesis with fewer numerical function evaluations (NFE).

2.3 Flow-GRPO

Flow-GRPO[19] is a recently proposed framework that integrates online policy gradient reinforcement learning (RL) into Flow Matching (FM) models. It enables the alignment of generative trajectories with specific reward signals (e.g., clinical fidelity or anatomical consistency) by addressing the deterministic nature of flow models and the efficiency challenges of online RL.

Stochastic Exploration via ODE-to-SDE. The standard Flow Matching model follows a deterministic Ordinary Differential Equation (ODE), $dx_t = v_t(x_t, t)dt$, which lacks the stochasticity required for RL exploration. To facilitate exploration, Flow-GRPO converts the deterministic flow into an equivalent Stochastic Differential Equation (SDE) that preserves the original marginal probability density $p_t(x)$ at all timesteps. Using Euler-Maruyama discretization, the update rule becomes:

$$x_{t+\Delta t} = x_t + \left[v_\theta(x_t, t) + \frac{\sigma_t^2}{2t}\big(x_t + (1-t)v_\theta(x_t, t)\big) \right] \Delta t + \sigma_t \sqrt{\Delta t}\epsilon \tag{4}$$

where $\epsilon \sim \mathcal{N}(0, I)$ and σ_t is a diffusion coefficient controlling the noise level. This stochastic sampling allows the model to explore diverse trajectories within the image space.

Group Relative Policy Optimization. Flow-GRPO adopts the Group Relative Policy Optimization (GRPO) [21] algorithm to eliminate the need for a resource-intensive Critic network, which is particularly beneficial for memory-constrained tasks like 3D medical image generation. For a given condition c, the model samples a group of G independent trajectories $\{x^i\}_{i=1}^G$. The advantage $\hat{A}_i$ for the i-th sample is computed by normalizing the reward r_i relative to the group:

$$\hat{A}_i = \frac{r_i - \text{mean}(\{r_j\}_{j=1}^G)}{\text{std}(\{r_j\}_{j=1}^G)} \tag{5}$$

This formulation provides a stable baseline for policy updates without an explicit value function.

Training Objective. The model is optimized by maximizing the expected advantage while maintaining proximity to a reference model via a Kullback-Leibler (KL) divergence penalty. The loss function is defined as:

$$\mathcal{J}(\theta) = \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G} \left(\min\left(\rho_t^i \hat{A}_i, \text{clip}(\rho_t^i, 1-\epsilon, 1+\epsilon)\hat{A}_i \right) - \beta D_{KL}(\pi_\theta || \pi_{\text{ref}}) \right) \right] \tag{6}$$

where $\rho_t^i = \frac{\pi_\theta(x_{t-1}^i | x_t^i, c)}{\pi_{\theta_{\text{old}}}(x_{t-1}^i | x_t^i, c)}$ is the importance sampling ratio. Additionally, Flow-GRPO employs a *Denoising Reduction* strategy, where only a fraction of denoising steps (e.g., $T = 10$) are used during training to accelerate sample collection, while the full schedule is retained during inference for high-quality synthesis.

3 Method

The overall architecture of our proposed MedGRPO is illustrated in Figure 1. Specifically, to achieve exemplar-based controllable tumor generation in a stepwise manner, we train the model in two stages: Supervised learning (SL) and reinforcement learning (RL). We will provide a detailed description of this process in this section.

3.1 Supervised learning for paired data

In this stage, we leverage a large collection of tumor CT scans with segmentation labels to perform inpainting-like Flow Matching training, enabling the model to learn the foundational tumor distribution. As illustrated in Figure 1(b), this stage involves frozen VAE encoder, text encoder (BiomedClip[22]), and a trainable Unet-based predictor.

Given a tumor CT image I and its corresponding tumor reference exemplar R, a frozen VAE encoder $\mathcal{E}$ is utilized to project the visual information into the latent space. The CT latent z_0 and reference tumor latent z_{ref} are obtained via:

$$z_0 = \mathcal{E}(I), \quad z_{ref} = \mathcal{E}(trans(R)), \tag{7}$$

where $trans(\cdot)$ denotes random data augmentation transformations, including centering, rotation, and scaling. This is designed to enable the model to flexibly leverage exemplar tumor information to generate plausible tumors, rather than simply performing a "copy-paste" operation.

Subsequently, we derive a bounding box B that encloses the original tumor location in I. We then obtain b by performing downsampling and channel expansion on B to align its dimensions with those of z_0. We partially corrupt z_0 with noise n based on b:

$$z_1 = (1-b)z_0 + b \cdot n. \tag{8}$$

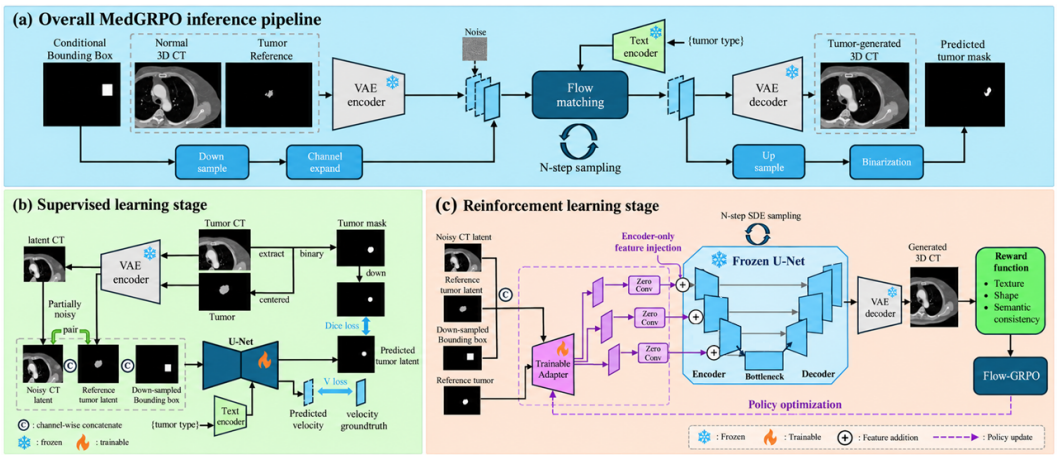


Figure 1: Overall architecture of MedGRPO. (a) Inference Stage: We generate tumors at arbitrary locations (specified by bounding boxes) within the target CT by conditioning on a exemplar tumor CT and the textual tumor type. The exemplar tumor can be sourced from different CT scans. (b) Supervised Learning Stage: *Paired* data, consisting of tumor-bearing CTs and their corresponding extracted tumors, are encoded via a VAE into the latent space for Flow Matching training. This process enables the model to learn the general distribution of tumors. (c) Reinforcement Learning Stage: The model is trained using *unpaired* tumor-free CTs and exemplar tumors. Initially, random samples are generated and evaluated by a reward function. Based on these evaluation results, the model undergoes adapter-based fine-tuning, thereby enhancing its generalization capability on *unpaired* data.

To provide exemplar-based guidance for the generation process, the model takes three components as input: the partially noisy CT latent z_t at time t, the reference tumor latent z_{ref}, and the down-sampled conditional bounding box b. These components are fused via channel-wise concatenation (denoted as ©):

$$X_{in} = [z_t © z_{ref} © b], \quad z_t = (1-t)z_0 + tz_1, \quad t \in [0,1]. \tag{9}$$

The trainable Unet, denoted as $f_\theta(\cdot)$, is optimized to predict the velocity field v_{pred} that drives the transformation and the tumor mask m_{pred}:

$$f_\theta(X_{in}, t, E_{tumor}) = v_{pred} © m_{pred}, \tag{10}$$

where E_{tumor} denotes the tumor type embedded by text encoder that is incorporated into the model via cross-attention mechanism.

The optimization objective consists of two primary loss functions:

Velocity Loss (V loss): Following the flow matching framework, the model minimizes the mean squared error between the predicted velocity $\hat{v}$ and the ground-truth velocity $v = z_0 - z_1$:

$$\mathcal{L}_v = \mathbb{E}_{t,z_t}\left[\frac{1}{|b|}\left\|b \cdot (v_{pred} - v)\right\|^2\right] \tag{11}$$

where v is the target vector field that defines the optimal probability flow from the source distribution to the target tumor distribution.

Dice Loss: To achieve initial control over the shape of the generated tumor using the exemplar tumor, a Dice loss is computed between the predicted tumor and the down-sampled (for aligning the spatial dimensions with the latent space) binary tumor mask M. The mask M is derived from the ground-truth tumor CT through down sampling and binarization. The Dice loss is formulated as:

$$\mathcal{L}_{Dice} = 1 - \frac{2(m_{pred} \cdot M) + \epsilon}{m_{pred} + M + \epsilon} \tag{12}$$

where ϵ is a small constant to ensure numerical stability.

The final objective function for the SFT stage is a weighted sum of the aforementioned losses:

$$\mathcal{L}_{SL} = \mathcal{L}_v + \lambda_{Dice}\mathcal{L}_{Dice} \tag{13}$$

where λ_{Dice} is a hyper-parameter balancing the velocity regression and the structural consistency. Through this supervised process, the Unet learns to generate high-fidelity tumor features conditioned on the specified location and reference morphology.

3.2 Reinforcement learning for unpaired data

After the SL phase, the model demonstrates basic exemplar-guided generation capabilities. Nevertheless, experiments show that in unpaired settings (where target CTs and exemplar tumors come from different subjects), the exemplar's influence on generation is markedly reduced compared to paired cases. Specifically, there are significant discrepancies between the generated and exemplar tumors regarding shape, texture, and high-level semantics (e.g., segmentation results) for unpaired data. This implies that the model has largely learned merely to "generate a plausible tumor", while neglecting the condition to "reference the exemplar tumor". More critically, the lack of tumor ground truth prevents the effective use of abundant unpaired data in a supervised learning framework.

To address these challenges, we introduce a Reinforcement Learning (RL) stage (as shown in Figure 1(c)) following Supervised Learning (SL) to further enhance the model's generalization capability. Since RL optimizes the model based on the evaluation of generated results rather than relying on precise quantitative ground truth, it is particularly well-suited for handling unpaired data. Specifically, we designed three reward functions to evaluate the generated results in terms of shape, texture, and segmentation semantics, respectively.

Shape reward. To quantitatively evaluate the shape similarity between the generated tumors and the exemplar tumors, we additionally trained a Lightweight shape extractor network to capture abstract shape features. Specifically, we extracted approximately 3000 tumors from our collected CT dataset to construct a dedicated tumor template dataset. We then employed this dataset to train the shape extractor $S_\phi(\cdot)$ using the following triplet-loss contrastive learning strategy[23]:

$$\begin{aligned}
&\mathcal{L}_{shape\ extractor} = \tfrac{1}{N}\textstyle\sum_{i=1,i\neq j}^{N}\left([d_{i,pos} - d_{i,j,neg} + \alpha] + d_{i,pos}\right), \\
&d_{i,pos} = \|\underbrace{S_\phi(r_i)}_{anchor} - \underbrace{S_\phi(trans(r_i))}_{positive\ sample}\|^2, \quad d_{i,j,neg} = \|\underbrace{S_\phi(r_i)}_{anchor} - \underbrace{S_\phi(r_j)}_{negtive\ sample}\|^2,
\end{aligned} \tag{14}$$

where r_i denotes i-th sample of the tumor template dataset, and $trans(\cdot)$ denotes random data augmentation transformation including centering, rotation and scaling, α is a predefined hyperparameter that defines the margin to distinguish between positive and negative pair distances. Once the shape extractor is trained, we can simply define the reward function as follows:

$$R_{shape}(m_{pred}, r) = -\|S_\phi(m_{pred}) - S_\phi(r)\|^2 \tag{15}$$

where r is input exemplar tumor.

Texture reward. To quantitatively evaluate the texture consistency between the generated tumor and the exemplar tumor, we can simply employ the Bhattacharyya distance [24](BD) for assessment:

$$BD(I_{pred}, m_{pred}, r) = -\ln \sum_{i=1}^{B} \sqrt{H_B(m_{pred}\cdot I_{pred})_i \cdot H_B(r)_i}, \quad R_{texture} = -BD, \tag{16}$$

where $H_B(X)$ denotes the B-bin normalized histograms of X, and I_{pred}, is the final generated CT.

Segmentation reward. In addition to enhancing the exemplar tumor's control over low-level semantic features such as shape and texture, we also aim to strengthen the influence of the input tumor type text on the generated tumor. To this end, we employ a pre-trained semantic segmentation network $\Psi(\cdot)$ that is capable of segmenting diverse fine-grained tumor types, to evaluate the final generated CT I_{pred}:

$$R_{seg}(I_{pred}, m_{pred}, \{tumor\ type\}) = -Dice(\Psi_{\{tumor\ type\}}(I_{pred}),\ Br(r)) \tag{17}$$

where $Dice(\cdot)$ denotes the Dice loss same as (12), and $\Psi_{\{tumor\ type\}}(X)$ denotes the segmentation mask for the specific tumor type within X, $Br(\cdot)$ is the binarization opration.

The first sub-stage of RL requires the model to perform random sampling. Specifically, under identical input conditions (r, b, and {tumor type}), the model generates K samples via stochastic exploration using an SDE solver (4) and records the SDE trajectories. These samples are then evaluated by the reward functions:

$$\bar{R}_\sigma = \{R^k_\sigma(I_{pred}, m^k_{pred})\}|^K_{k=1}, \quad \sigma \in [shape, texture, seg]. \tag{18}$$

For each group $\bar{R}_\sigma$, we compute the advantage for each sample using the **Group Relative Policy** (5):

$$A_\sigma^k = \frac{R_\sigma^k - \text{mean}(\bar{R}_\sigma)}{\text{std}(\bar{R}_\sigma)}. \tag{19}$$

We then compute the final advantage for each sample by averaging the advantages across the three reward categories:

$$A^k = \frac{1}{3}\sum_\sigma A_\sigma^k \tag{20}$$

A positive A^k indicates that the generation process for that sample will be encouraged, whereas a negative one implies it will be penalized. Specifically, we modify the log-probability of each sample point along the SDE trajectory using (6). For samples with positive advantages ($A^k > 0$), we shift the distribution of each sampling step towards its mean; conversely, for samples with negative advantages ($A^k < 0$), we push it away from the mean.

During our experiments, we observed that performing full fine-tuning on the U-Net leads to training instability and difficulties in reward convergence. Therefore, inspired by [25], we introduce an external lightweight CNN module (adapter) that takes the exemplar tumor as input. Its output is added to the intermediate features of the U-Net via zero-initialized convolutions to prevent excessive perturbation of the U-Net during the initial training phase. During RL training, we freeze the U-Net and train only the adapter. This strategy preserves the pre-trained U-Net parameters and prevents the model from deviating significantly from the distribution learned during the SL phase, thereby ensuring a stable increase in the reward.

4 Experiments

The experimental implementation details are provided in the Appendix A.

4.1 Datasets

Our experiments are conducted on the following three datasets:

Internal 4-Cancer Dataset. A privately curated dataset collected from multiple clinical institutions, comprising 15,419 labeled CT images. It serves as the primary data source for training our generative model and downstream diagnostic models. The training set includes 1,287 breast, 389 kidney, 6,176 liver, and 4,767 pancreatic tumor cases. During inference, cases without corresponding lesions are utilized as tumor-free cases for tumor synthesis.

External AbdomenAtlas 2.0 [10] Dataset. The AbdomenAtlas 2.0 is used as an external dataset to assess model generalizability. We evaluate our downstream models on liver (1,493 scans), pancreas (1,135 scans), and kidney (2,200 scans) subsets. This dataset is kept entirely unseen during the training of both the generative models and the downstream segmentation models, serving only for evaluation purposes.

Tumor template dataset. This dataset consists of tumor-only CT patches extracted from the tumor-bearing samples in the **4-Cancer dataset**. It contains approximately 1,000 tumor templates for each organ.

4.2 Quantitative Comparison

We conduct a quantitative evaluation focusing on distributional fidelity and downstream clinical utility, benchmarking MedGRPO against state-of-the-art baselines (details in Appendix B): (1) Heuristic/GAN-based: SynTumor [26] and FreeTumor [27]; (2) Diffusion inpainting: DiffTumor [14], LeFusion [15], and LesionDiffusion [16]; and (3) Whole-volume generation: MAISI V2 [13].

Distributional Fidelity. We evaluate the perceptual realism of the generated tumors by calculating the Fréchet Inception Distance (FID) within the tumor Regions of Interest (ROI). As shown in Table 2, MedGRPO achieves the lowest FID scores across all four organs (pancreas, liver, kidney, and breast) in 4-Cancer dataset, significantly outperforming baseline methods, and achieves the lowest FID scores across two organs (pancreas and liver) in AbdomenAtlas 2.0 dataset. This superior

Table 2: Comparison of FID scores (mean ± standard deviation) with baseline methods on both internal and external datasets.

Datasets	4-Cancer (Internal)				AbdomenAtlas 2.0 (External)		
methods / organs	breast	liver	kidney	pancreas	liver	kidney	pancreas
Syntumor[26]	-	36.81±0.27	70.04±0.34	34.87±0.21	7.95±0.10	94.23±0.64	13.92±0.10
Lefusion[15]	196.08±2.80	68.75±0.67	79.78±0.50	39.98±0.40	27.55±0.38	104.64±0.15	10.55±0.13
LesionDiffusion[16]	97.38±0.98	36.17±0.34	79.93±0.13	42.06±0.26	10.84±0.15	98.83±0.17	13.65±0.30
DiffTumor[14]	-	36.94±0.53	67.50±0.40	39.72±0.21	9.68±0.35	**70.16±0.59**	10.17±0.14
Freetumor[27]	-	115.68±0.17	141.71±0.31	109.29±0.06	82.82±0.08	195.77±0.27	75.73±0.06
MAISI[13]	-	44.35±0.32	-	90.42±0.26	20.97±0.14	-	52.50±0.22
MedGRPO (no-RL)	64.67±1.71	30.71±0.37	59.24±0.67	27.71±0.23	4.95±0.05	90.51±0.45	**1.60±0.03**
MedGRPO	**57.67±1.04**	**25.21±0.29**	**50.66±0.31**	**23.55±0.51**	**3.27±0.03**	83.35±0.47	2.30±0.06

The best scores are indicated in **bold**.

Table 3: Lesion segmentation performance (mean ± standard deviation) on AbdomenAtlas 2.0 datasets.

	Kidney		Liver		Pancreas	
Method	DSC↑	NSD↑	DSC↑	NSD↑	DSC↑	NSD↑
nnUNet	64.51±0.49	63.95±0.74	60.08±0.45	57.53±0.61	31.54±0.28	27.61±0.30
SynTumor	64.39±0.55	64.32±0.57	63.08±0.42	**63.17±0.50**	33.20±1.27	30.31±1.19
LeFusion	65.59±0.46	65.71±0.53	59.83±0.31	58.45±0.32	31.75±1.15	27.43±0.91
LesionDiffusion	67.99±0.23	68.41±0.25	62.80±0.52	62.13±0.47	34.04±1.06	31.02±1.11
DiffTumor	69.51±0.20	70.31±0.26	63.40±0.20	62.92±0.37	34.85±0.81	32.26±0.89
FreeTumor	69.99±0.23	70.58±0.26	61.55±0.34	60.79±0.51	34.10±0.60	31.48±0.78
MAISI	-	-	60.36±0.34	59.12±0.19	31.65±0.96	28.22±0.81
MedGRPO (no-RL)	70.35±0.37	71.62±0.29	64.21±0.29	62.76±0.23	34.43±1.21	31.62±1.28
MedGRPO	**71.20±0.27**	**72.43±0.41**	**64.23±0.25**	62.73±0.31	**36.66±0.80**	**33.85±0.80**

The best scores are indicated in **bold**.

alignment suggests that our reinforcement learning strategy successfully guides the model to capture the complex manifold of real-world pathologies.

Downstream Performance Gain. The ultimate measure of synthetic data quality lies in its ability to enhance the performance and generalization of diagnostic models. Following the out-of-distribution (OOD) evaluation protocol [10], we train nnUNet [28] on the internal dataset using either real data alone or real data augmented with synthetic samples generated by different methods. The trained models are then evaluated on the external AbdomenAtlas 2.0 dataset to assess cross-site generalization. As shown in Table 3, MedGRPO achieves the best performance in five out of six evaluation metrics across three organs, consistently outperforming the real-data-only baseline. Specifically, compared with the baseline, MedGRPO improves DSC by 6.69, 4.15, and 5.12 percentage points for kidney, liver, and pancreas, respectively. Moreover, MedGRPO generally outperforms competing synthesis methods, demonstrating the effectiveness of exemplar-guided tumor generation for downstream segmentation. Compared with MedGRPO (no-RL), the full model further improves five out of six metrics, highlighting the contribution of reinforcement learning to enhancing synthetic data quality. These results suggest that leveraging unpaired data through reinforcement learning can improve the clinical utility of synthetic tumors and facilitate cross-site generalization.

4.3 Qualitative Comparison

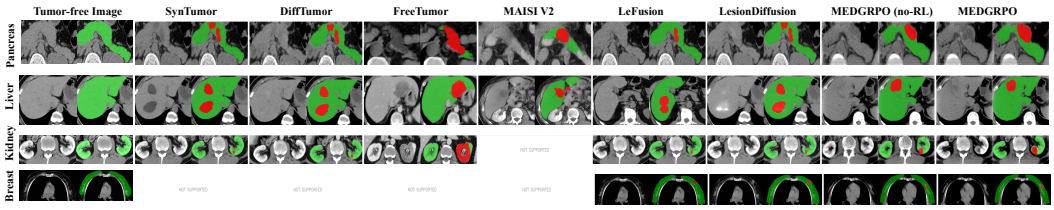


Figure 2: Visual comparison of tumors generated by different methods. Synthetic tumors and their corresponding segmentation masks across four distinct organs: pancreas, liver, kidney, and breast are visualized.

Visual Realism. Figure 2 provides a visual comparison of synthetic tumors across four organs. Compared to existing inpainting methods like LeFusion and LesionDiffusion, our framework generates high-fidelity tumors with seamless integration into the host organs. Notably, the comparison between

MedGRPO and its no-RL variant demonstrates that RL fine-tuning effectively enhances textural realism and eliminates boundary artifacts. More visualizations are provided in Appendix C.

Exemplar Control Analysis. To investigate the controllability of the proposed exemplar-driven paradigm, we visualize generation results where a single CT volume is conditioned on varying exemplar tumors. With a fixed random seed, the tumor region in the generated samples varies according to the exemplar tumor, exhibiting morphological similarity to the exemplar. However, instead of simply "copying and pasting", the generated tumors retain certain differences. This aligns with our goal of exemplar-based generation, which aims for resemblance rather than exact replication. Visual results are provided in Figure 3.

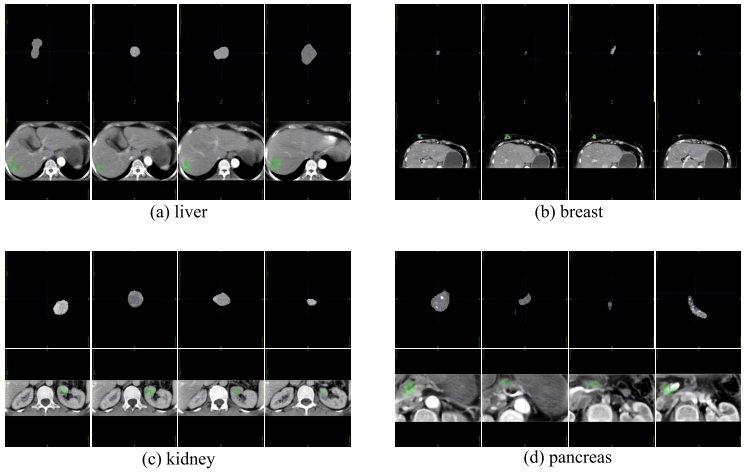


Figure 3: Generation results for a single target CT with varying exemplar tumors. Please zoom in for detailed visualization.

Expert Reader Study. To assess the clinical realism of the generated tumors, we conducted a blinded reader study involving two radiologists (R1: junior; R2: senior). A total of 200 CT cases were evaluated across four tumor categories, each containing 25 real and 25 synthetic cases. The radiologists independently examined consecutive CT slices and classified each case as real or synthetic without knowing the class distribution. As shown in Table 4, R1 and R2 achieved accuracies of 55.5% and 75.0%, respectively. Notably, R1 misclassified 76.0% of synthetic tumors as real, indicating that the generated tumors can exhibit convincing clinical appearances. The inter-reader Cohen's Kappa of 0.1748 indicates slight agreement, suggesting considerable reader-dependent variability in distinguishing real from synthetic tumors. These findings further support the clinical plausibility of MedGRPO-generated tumors.

Table 4: Expert reader study on distinguishing real and synthetic tumors. Real tumors are considered positive, while synthetic tumors are considered negative.

Reader	Sensitivity	Specificity	Accuracy	Cohen's κ
R1 (Junior)	87.0%	24.0%	55.5%	0.1748
R2 (Senior)	88.0%	62.0%	75.0%	0.1748

4.4 Ablation study for RL

Figure 4 illustrates the trends of the three reward components during RL training. It is evident that under our training strategy, all three rewards improved simultaneously. Specifically, the texture reward converged at approximately 150k iterations, the segmentation reward continued to increase, and the shape reward converged as early as 50k iterations. More intuitively, Figure 5 presents the 3D visualizations of generated tumors alongside exemplar tumors, both before and after RL training. Prior to RL, the generated tumors often lacked obvious shape correlation with the exemplars; the model merely produced compact and anatomically plausible tumors. However, after applying our RL

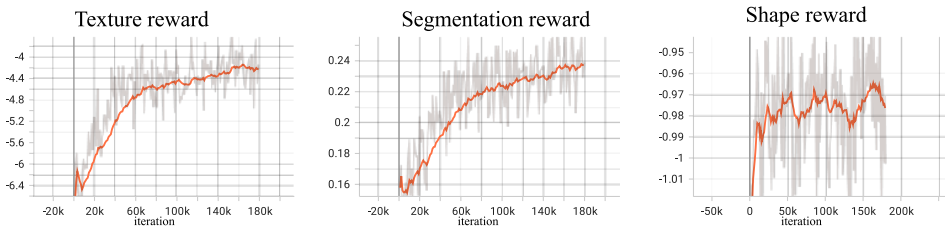


Figure 4: Trends in reward values.

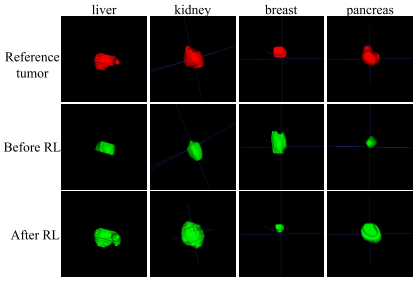


Figure 5: Comparison of generated tumor shapes before and after RL training (visualized using ITK-SNAP).

Table 5: Comparison of average reward values of generated samples under different reward settings.

setting/mean reward	Texture	Segmentation	Shape
MedGRPO (no-RL)	-6.2	0.16	1.14
+ Texture-only	**-3.9**	0.19	-1.03
+ Segmentation-only	-6.0	**0.22**	*-1.22*
+ Shape-only	*-6.3*	0.14	**-0.98**
MedGRPO (full)	-4.1	0.18	-1.01

The highest scores are indicated in **bold**, and the results that lower than "no-RL" are indicated in *italics*.

strategy, the model not only maintained compactness and plausibility but also successfully captured the shape characteristics of the exemplar tumors, generating structures that closely resemble them.

Furthermore, we explored the generation performance of the model when trained with individual reward functions. Specifically, we trained the model using different reward configurations, generated 3,000 samples for each setting, and recorded the average texture, segmentation, and shape reward values. As shown in Table 5, consistent with intuition, when each reward function is optimized individually, the corresponding average reward value for the generated samples tends to reach its maximum, surpassing the scores achieved under the joint reward setting. However, contrary to expectations, there appears to be a competitive relationship between the segmentation and shape rewards. In the Segmentation-only setting, the shape reward drops sharply, falling even below the baseline (no-RL) level. Similarly, in the Shape-only setting, the segmentation reward also decreases below that of the no-RL setting. Under the joint reward setting, the generated samples achieve acceptable performance across all metrics. This balanced approach prevents the model from learning 'shortcuts' by relying exclusively on a single aspect of the condition.

5 Conclusion

We present MedGRPO, an exemplar-driven 3D tumor synthesis framework that overcomes the "specification bottleneck" by utilizing high-dimensional pathological templates. By optimizing flow trajectories via Flow-GRPO, our model balances pathological fidelity with anatomical coherence. Evaluations on the internal 4-Cancer and external AbdomenAtlas 2.0 dataset demonstrate significant performance gains for downstream diagnostic models. MedGRPO offers a robust, scalable solution to data scarcity in medical AI.

Acknowledgments and Disclosure of Funding

This work received no third-party funding or support, and the authors declare no competing interests.

References

[1] Kai Cao, Yingda Xia, Jiawen Yao, Xu Han, Lukas Lambert, Tingting Zhang, Wei Tang, Gang Jin, Hui Jiang, Xu Fang, et al. Large-scale pancreatic cancer detection via non-contrast ct and

deep learning. *Nature medicine*, 29(12):3033–3043, 2023.

[2] Can Hu, Yingda Xia, Zhilin Zheng, Mengxuan Cao, Guoliang Zheng, Shangqi Chen, Jiancheng Sun, Wujie Chen, Qi Zheng, Siwei Pan, et al. Ai-based large-scale screening of gastric cancer from noncontrast ct imaging. *Nature Medicine*, 31(9):3011–3019, 2025.

[3] Jinzhuo Wang, Kai Wang, Yunfang Yu, Yuxing Lu, Wenchao Xiao, Zhuo Sun, Fei Liu, Zixing Zou, Yuanxu Gao, Lei Yang, et al. Self-improving generative foundation model for synthetic medical image generation and clinical applications. *Nature Medicine*, 31(2):609–617, 2025.

[4] Theodore Zhao, Yu Gu, Jianwei Yang, Naoto Usuyama, Ho Hin Lee, Sid Kiblawi, Tristan Naumann, Jianfeng Gao, Angela Crabtree, Jacob Abel, et al. A foundation model for joint segmentation, detection and recognition of biomedical objects across nine modalities. *Nature methods*, 22(1):166–176, 2025.

[5] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(9):10795–10816, 2023.

[6] Chiao-An Yang, Kuan-Chuan Peng, and Raymond A Yeh. Toward long-tailed online anomaly detection through class-agnostic concepts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23419–23430, 2025.

[7] Ms Aayushi Bansal, Dr Rewa Sharma, and Dr Mamta Kathuria. A systematic review on data scarcity problem in deep learning: solution and applications. *ACM Computing Surveys (Csur)*, 54(10s):1–29, 2022.

[8] Nora Pashayan and Paul DP Pharoah. The challenge of early detection in cancer. *Science*, 368(6491):589–590, 2020.

[9] Richard J Chen, Ming Y Lu, Tiffany Y Chen, Drew FK Williamson, and Faisal Mahmood. Synthetic data in machine learning for medicine and healthcare. *Nature Biomedical Engineering*, 5(6):493–497, 2021.

[10] Qi Chen, Xinze Zhou, Chen Liu, Hao Chen, Wenxuan Li, Zekun Jiang, Ziyan Huang, Yuxuan Zhao, Dexin Yu, Junjun He, et al. Scaling tumor segmentation: Best lessons from real and synthetic data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 24001–24013, 2025.

[11] Fei Lyu, Mang Ye, Andy J Ma, Terry Cheuk-Fung Yip, Grace Lai-Hung Wong, and Pong C Yuen. Learning from synthetic ct images via test-time training for liver tumor segmentation. *IEEE transactions on medical imaging*, 41(9):2510–2520, 2022.

[12] Pengfei Guo, Can Zhao, Dong Yang, Ziyue Xu, Vishwesh Nath, Yucheng Tang, Benjamin Simon, Mason Belue, Stephanie Harmon, Baris Turkbey, et al. Maisi: Medical ai for synthetic imaging. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4430–4441. IEEE, 2025.

[13] Can Zhao, Pengfei Guo, Dong Yang, Yucheng Tang, Yufan He, Benjamin Simon, Mason Belue, Stephanie Harmon, Baris Turkbey, and Daguang Xu. Maisi-v2: Accelerated 3d high-resolution medical image synthesis with rectified flow and region-specific contrastive loss. *arXiv preprint arXiv:2508.05772*, 2025.

[14] Qi Chen, Xiaoxi Chen, Haorui Song, Zhiwei Xiong, Alan Yuille, Chen Wei, and Zongwei Zhou. Towards generalizable tumor synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11147–11158, 2024.

[15] Hantao Zhang, Yuhe Liu, Jiancheng Yang, Shouhong Wan, Xinyuan Wang, Wei Peng, and Pascal Fua. Lefusion: Controllable pathology synthesis via lesion-focused diffusion models. *arXiv preprint arXiv:2403.14066*, 2024.

[16] Henrui Tian, Wenhui Lei, Linrui Dai, Hanyu Chen, and Xiaofan Zhang. Lesiondiffusion: Towards text-controlled general lesion synthesis. *arXiv preprint arXiv:2503.00741*, 2025.

[17] Pingping Dai, Licong Dong, Ruihan Zhang, Haiming Zhu, Jie Wu, and Kehong Yuan. Soft-cp: A credible and effective data augmentation for semantic segmentation of medical lesions. *arXiv preprint arXiv:2203.10507*, 2022.

[18] Jiawei Yang, Yao Zhang, Yuan Liang, Yang Zhang, Lei He, and Zhiqiang He. Tumorcp: A simple but effective object-level data augmentation for tumor segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 579–588. Springer, 2021.

[19] Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470*, 2025.

[20] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[21] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[22] Sheng Zhang, Yanbo Xu, Naoto Usuyama, Hanwen Xu, Jaspreet Bagga, Robert Tinn, Sam Preston, Rajesh Rao, Mu Wei, Naveen Valluri, et al. Biomedclip: a multimodal biomedical foundation model pretrained from fifteen million scientific image-text pairs. *arXiv preprint arXiv:2303.00915*, 2023.

[23] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[24] Anil Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distribution. *Bulletin of the Calcutta Mathematical Society*, 35:99–110, 1943.

[25] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 4296–4304, 2024.

[26] Qixin Hu, Yixiong Chen, Junfei Xiao, Shuwen Sun, Jieneng Chen, Alan L Yuille, and Zongwei Zhou. Label-free liver tumor segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7422–7432, 2023.

[27] Linshan Wu, Jiaxin Zhuang, Xuefeng Ni, and Hao Chen. Freetumor: Advance tumor segmentation via large-scale tumor synthesis. *arXiv preprint arXiv:2406.01264*, 2024.

[28] Fabian Isensee, Paul F Jaeger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 18(2):203–211, 2021.

A Implementation Details

We trained our model on 32 NVIDIA H20 GPUs using AdamW optimizer. The SL stage conducted on 4-Cancer dataset (150 epochs, batch size 64, learning rate 10^{-5}) and took 3,040 GPU hours. The RL stage used an unpaired strategy involving 200 tumor-free CTs from **4-Cancer dataset** and tumor templates from **Tumor template dataset**. This stage ran for 160 epochs with a batch size of 32, requiring 1,184 GPU hours. In total, the full training pipeline required over 4,200 GPU hours. The hyperparameters are set as follows: $\lambda_{Dice} = 0.5,\ K = 16$.

B Details of baseline methods

We benchmark MedGRPO against six state-of-the-art tumor synthesis frameworks:

Type I: Heuristic and GAN-based Simulation. SynTumor [26]is a model-free benchmark that synthesizes lesions via heuristic-driven mask simulation and texture matching. For three supported organs (liver, kidney, pancreas), we utilized its official implementation to generate 3,000 synthetic pairs on our tumor-free cases. **FreeTumor** [27] employs a GAN-based architecture for regional texture simulation. We directly deployed their official pre-trained checkpoints for the three supported organs to ensure a fair comparison.

Type II: Diffusion Inpainting. This category focuses on localized lesion synthesis within established organ contexts. **DiffTumor** [14] serves as a hybrid baseline, combining SynTumor's heuristic masks with a diffusion-based inpainting stage; we trained this model on our internal dataset for three-organ support. **LeFusion** [15] utilizes a two-stage pixel-level diffusion model with histogram-based intensity control. **LesionDiffusion** [16] operates in the latent space via a VQ-GAN, employing attribute-guided synthesis. For both, we trained the complete pipelines from scratch across all four cancer types in our dataset.

Type III: Whole Volumetric Generation. Departing from the inpainting paradigm, **MAISI v2** [13] utilizes a large-scale Latent Diffusion Transformer to generate entire 3D CT volumes. We leveraged their pre-trained models to synthesize 3,000 whole-volume cases for liver and pancreas tumors.

C Additional Visual Comparison

We present a broader array of synthetic tumor samples generated by MedGRPO to further illustrate its clinical plausibility. While baseline methods like SynTumor [26] and FreeTumor [27] often produce sharp, unrealistic edges or repetitive textures, the samples in Figure 6 demonstrate MedGRPO's ability to generate pathological heterogeneity.

D Potential societal impact

Our work on tumor image generation carries significant potential for both the medical research community and broader society. Positively, it addresses the critical challenge of medical data scarcity and high annotation costs, particularly for rare tumor types. By providing high-quality synthetic datasets, our method can accelerate the development of robust computer-aided diagnosis (CAD) systems without compromising patient privacy, as synthetic images do not belong to real individuals. This contributes to the democratization of medical AI research, allowing institutions with limited access to clinical data to develop effective diagnostic tools.

On the other hand, we recognize potential ethical risks. The misuse of synthetic images could lead to the dissemination of misinformation or the creation of "medical deepfakes." Furthermore, if the generative model inherits biases from the training distribution (e.g., under-representing certain demographics), it may lead to algorithmic unfairness in clinical applications. We emphasize that synthetic images should currently serve as a supplement to, rather than a replacement for, real clinical data, and should be used under professional supervision to avoid potential misdiagnosis.

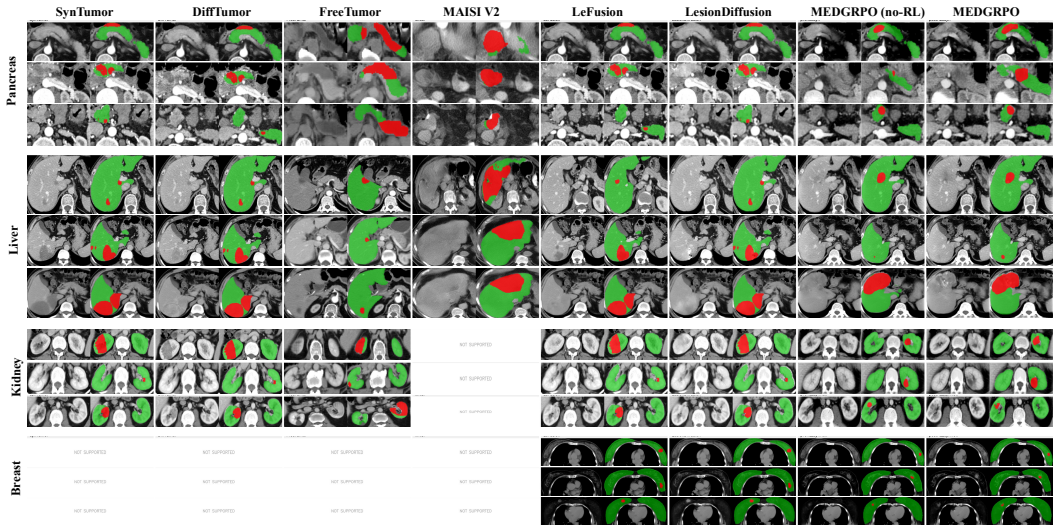


Figure 6: Additional visual comparison of tumors generated by different methods. Synthetic tumors and their corresponding segmentation masks across four distinct organs: pancreas, liver, kidney, and breast are visualized..

E Limitations

Currently, our framework is validated primarily on 3D CT images across four major solid organs (liver, kidney, pancreas, and breast). While these are critical for clinical diagnosis, the performance of MedGRPO on other modalities (such as MRI or PET) or other anatomical structures (e.g., brain or lung) remains to be explored.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In Section 1 and Abstract.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Described in Section 3.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: In section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Our code will be made publicly available after the camera-ready stage.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: In section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: In section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In section 4.1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We comply with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: In Appendix.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: In Appendix.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The methods and datasets mentioned in this paper are all publicly available and open-source or our private one. They are properly cited in the article, and there is no plagiarism or misuse.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We have provided a detailed description of how to use our code in and commit to making our code publicly available on GitHub after the camera-ready stage.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This work is the retrospective analysis of de-identified medical images and does not involve crowdsourcing or active participation from human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: Data collection was approved by the ethics committee of each institution.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: This research does not involve LLMs as any important, original, or non-standard components

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.