# PADM: A Physics-Aware Diffusion Model for Attenuation Correction (Supplementary Material)

Anonymous WACV Algorithms Track submission

Paper ID 3311

## 1. Experiments Details

### 1.1. System & Hyper-parameters

The experiments are conducted on a system featuring an AMD EPYC 7402P CPU and an NVIDIA A6000 GPU. In this section, we also provide further implementation details for PADM and other methods.

#### 1.1.1. Our proposed model - PADM

We implement PADM as described in Section 4. In contrast to the original BBDM setup, we do not embed the preprocessed NAC and AC images into the latent space of VQGAN, since their relatively low resolution diminishes the benefit of such compression. Instead, diffusion is directly performed in the image space. The Brownian Bridge timesteps are set to 500 for training and reduced to 10 for inference, striking a balance between reconstruction quality and computational efficiency.

#### 1.1.2. Pix2Pix

Pix2Pix [2] is a conditional GAN framework designed for paired image-to-image translation. In our setup, the generator adopts a ResNet backbone with 9 residual blocks, while the discriminator follows the 70×70 PatchGAN design [2]. The model is optimized with Adam ($\beta_1 = 0.5$) using the least-squares adversarial loss (LSGAN). Instance normalization is consistently applied to both the generator and discriminator. Training runs for 100 epochs at a fixed learning rate of $2 \times 10^{-4}$, followed by a linear decay across an additional 100 epochs. To improve stability, a buffer of 50 previously generated images is maintained. Our implementation is adapted from the public repository at https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix.

#### 1.1.3. Palette

Palette [6] is an image-to-image diffusion framework that adopts an exponential moving average (EMA) strategy from the very first training step, with a decay coefficient of 0.9999 to ensure stability. The model is optimized using Adam with a fixed learning rate of $5 \times 10^{-5}$ and no weight decay. A linear beta schedule is applied over 1000 training timesteps, starting from $1 \times 10^{-6}$ and ramping up to 0.01. During inference, sampling is carried out with 100 steps. Our implementation follows the official open-source release available at https://github.com/Janspiry/Palette-Image-to-Image-Diffusion-Models

Table 1. Training and optimization hyperparameters for PADM.

| Component | Configuration |
|---|---|
| **Denoising Network** | |
| Channels | 128 |
| Channel multipliers | 1, 4, 8 |
| Residual blocks per downsample | 2 |
| Attention resolutions | 32, 16, 8 |
| # of Trainable Parameters | 237.11M |
| **EMA** | |
| Start step | 30,000 |
| Decay | 0.995 |
| Update interval | 16 |
| Batch size | 8 |
| **Learning Rate Scheduler (ReduceLROnPlateau)** | |
| Max learning rate | $1.0 \times 10^{-4}$ |
| Min learning rate | $5.0 \times 10^{-7}$ |
| Factor | 0.5 |
| Patience | 3,000 |
| Cooldown | 3,000 |
| Threshold | $1.0 \times 10^{-4}$ |

#### 1.1.4. BBDM

BBDM [4] formulates image-to-image translation as a diffusion process defined by a stochastic Brownian bridge in the VQ-GAN latent domain. For experimental consistency, no pretrained VQ-GAN is used for initialization. The network is optimized using Adam with a fixed learning rate of

$1 \times 10^{-4}$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. A learning rate schedule is applied, consisting of a 3000-step cooldown, multiplicative decay of 0.5, and a minimum threshold of $5 \times 10^{-7}$. To stabilize training, an exponential moving average (EMA) with decay factor 0.995 is introduced after 30k iterations. The diffusion process employs 500 steps during training, while inference leverages DDIM [7] with only 10 steps for efficiency. All experiments are implemented in accordance with the official codebase at https://github.com/xuekt98/BBDM.git .

#### 1.1.5. ResViT

ResViT [1] is a GAN-based framework for medical image synthesis, featuring a generator with Aggregated Residual Transformer (ART) blocks that unify convolutional and transformer modules to jointly exploit local detail and global context. The training of ResViT was performed in two stages. In the first stage, we pre-trained the CNN backbone (ART blocks without transformers) to learn local structural representations. This was implemented with learning rate of 0.0002, and 100 epochs (50 epochs with fixed learning rate followed by 50 epochs of linear decay). The resulting CNN weights were saved as initialization for the second stage. In the fine-tuning stage, the full ResViT model, including both convolutional and transformer modules. The network was initialized with the pre-trained CNN backbone, while both transformer and residual modules were jointly optimized. Fine-tuning was performed for 50 epochs (25 epochs with fixed learning rate followed by 25 epochs of linear decay) with a higher learning rate of 0.001 to accelerate convergence. Our implementation is adapted from the official repository at https://github.com/icon-lab/ResViT.

#### 1.1.6. Reg-GAN

RegGAN [3] is a GAN-based framework designed for medical image-to-image translation. In our setup, all input images are normalized to the range $[-1, 1]$ and resampled to $256 \times 256$. The generator follows the CycleGAN backbone with two downsampling convolution blocks, nine residual blocks, and two upsampling deconvolution blocks, while the discriminator consists of four convolutional layers. A U-Net-based registration network is incorporated to estimate deformation fields for noise correction. The model is trained using the Adam optimizer with learning rate $1 \times 10^{-4}$, $\beta_1 = 0.5$, $\beta_2 = 0.999$, and weight decay $1 \times 10^{-4}$, with batch size set to 1. Training is conducted for 80 epochs (approximately 640K iterations). The overall objective combines multiple loss terms with fixed weights: adversarial loss ($\lambda_{adv} = 1$), L1 loss ($\lambda_{L1} = 100$), cycle-consistency loss ($\lambda_{cyc} = 10$), correction loss ($\lambda_{corr} = 20$), and smoothness loss ($\lambda_{smooth} = 10$). Our implementation is adapted from the official repository at https://github.com/Kid-Liet/Reg-GAN.

#### 1.1.7. UNIT

UNIT [5] is an unsupervised image-to-image translation framework that combines variational autoencoders (VAEs) and generative adversarial networks (GANs) under the assumption of a shared latent space between domains. The model employs the Adam optimizer with a learning rate of $1 \times 10^{-4}$ and momentum parameters $\beta_1 = 0.5$ and $\beta_2 = 0.999$, with no weight decay. Training alternates between updating the generator and discriminator networks to maintain stability. The architecture consists of domain-specific encoders and decoders linked by a shared latent representation, enabling bidirectional translation. Training is performed with a batch size of 16 over 150 epochs. Inference is achieved by encoding an input image into the shared latent space and decoding it into the target domain. The implementation is based on the official UNIT code release at https://github.com/mingyuliutw/UNIT

### 1.2. Detail of the Evaluation Metrics

We utilize Root Mean Squared Error (RMSE), Structural Similarity Index Measure (SSIM), and Peak Signal-to-Noise Ratio (PSNR) metrics to assess the quality of generated AC images. RMSE measures the average error, indicating accuracy. SSIM reflects visual quality by assessing changes in the image structure, whereas PSNR gauges image fidelity.

#### 1.2.1. Root Mean Squared Error (RMSE)

The Root Mean Squared Error (RMSE) measures the square root of the average of the squared differences between the predicted values and the actual values. RMSE penalizes larger errors more strongly than MAE and is widely used to evaluate regression and reconstruction performance in machine learning and computer vision. A lower RMSE indicates better model accuracy. The formula for RMSE is given by:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

where $y_i$ is the ground-truth value, $\hat{y}_i$ is the predicted value, and $n$ is the number of samples.

#### 1.2.2. Structural Similarity Index Measurement (SSIM)

SSIM is a perceptual metric that quantifies how similar two images are in terms of visual quality. It evaluates image fidelity using three components: (i) luminance, (ii) contrast, and (iii) structural information. The score ranges from –1 to 1, where a value of 1 indicates perfect similarity. Models achieving higher SSIM values produce outputs that more closely resemble the reference images. The formulation of SSIM is expressed as:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

where $\mu_x$ and $\mu_y$ are the mean values of $x$ and $y$, $\sigma_x^2$ and $\sigma_y^2$ are the variances of $x$ and $y$, $\sigma_{xy}$ is the covariance of $x$ and $y$, and $c_1$ and $c_2$ are constants to stabilize the division.

#### 1.2.3. Peak Signal-to-Noise Ratio (PSNR)

This metric quantifies the proportion between the peak possible signal power and the noise power that distorts the signal, and it is usually reported in decibels (dB). In image quality assessment, a larger PSNR value reflects superior fidelity. When two images are exactly the same, the PSNR tends toward infinity. As a rule of thumb, a PSNR of 40 dB or above indicates that visual differences between the original and reconstructed images are barely perceptible to the human eye. The PSNR is formally defined as:

$$\text{PSNR} = 10 \cdot \log_{10} \left( \frac{\text{MAX}^2}{\text{MSE}} \right)$$

where MAX is the maximum possible pixel value of the image and MSE is the mean squared error between the original and generated images.

## 2. Additional Qualitative Results

Supplementary results on cardiac SPECT slices are illustrated in Figures 1. Alongside each visualization, error maps are displayed to reveal deviations from the ground truth and facilitate clearer method comparisons.

## References

[1] Onat Dalmaz, Mahmut Yurt, and Tolga Cukur. Resvit: Residual vision transformers for multimodal medical image synthesis. *IEEE Transactions on Medical Imaging*, 41(10): 2598–2614, 2022. 2

[2] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1125–1134, 2017. 1

[3] Lingke Kong, Chenyu Lian, Detian Huang, Yanle Hu, Qichao Zhou, et al. Breaking the dilemma of medical image-to-image translation. *Advances in Neural Information Processing Systems*, 34:1964–1978, 2021. 2

[4] Bo Li, Kaitao Xue, Bin Liu, and Yu-Kun Lai. BBDM: Image-to-image translation with brownian bridge diffusion models. In *Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1952–1961, 2023. 1

[5] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. *Advances in neural information processing systems*, 30, 2017. 2

[6] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *Proceedings of the 2022 ACM Special Interest Group on Computer Graphics and Interactive Techniques Conference*, pages 1–10, 2022. 1

[7] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *Computing Research Repository arXiv Preprints arXiv:2010.02502*, 2020. 2

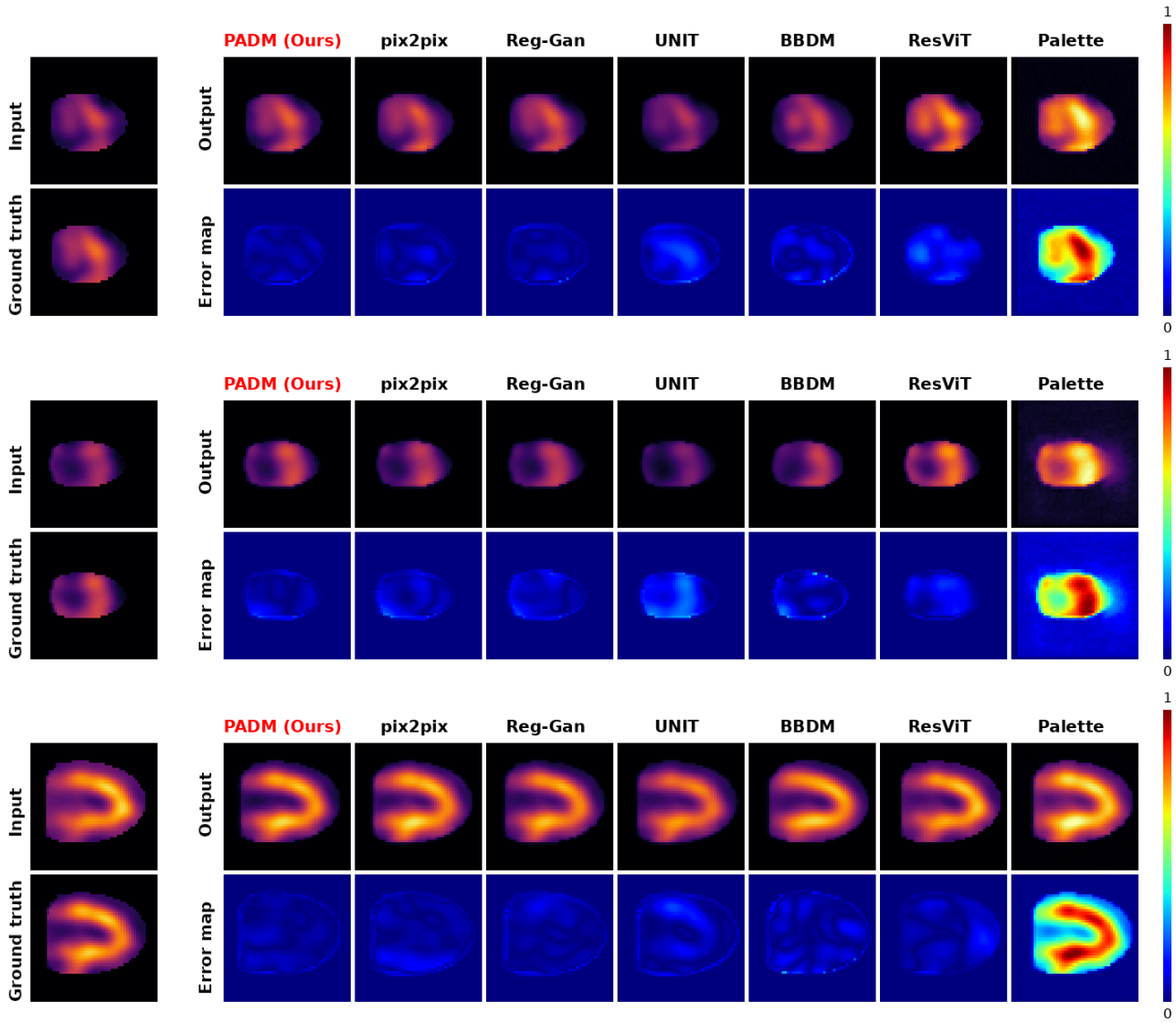


Figure 1. Qualitative comparison of reconstructed images across three standard views: horizontal long axis (top), short axis (middle), and vertical long axis (bottom).