Generative Models for Visual Signals – Assignment

Chang-Yu Chao[1,2]

[1] **Student ID: F74096069**
[2] **Code and supplementary material are available at https://github.com/felixchao/Generative-Models-for-Visual-Signals.git**

Abstract

This report presents a novel integration of Denoising Diffusion Probabilistic Models (DDPM) and Deep Image Prior (DIP) within the Truncated Diffusion Probabilistic Models (TDPM) framework [1]. By combining the iterative denoising process of DDPM with the robust prior capabilities of DIP, this approach aims to enhance the efficiency and quality of image generation and restoration. Experimental results demonstrate that the TDPM-DIP framework not only improves image synthesis fidelity but also gets advancement in image denoising tasks, thereby pushing the boundaries of current generative modeling techniques.

1 Introduction

In recent years, advancements in generative models have significantly enhanced the capability to produce high-quality images. Among these models, Denoising Diffusion Probabilistic Models (DDPM) have garnered attention due to their ability to generate diverse and high-fidelity images through a diffusion process. DDPM operates by iteratively adding noise to an image and then learning to reverse this process, effectively generating samples from a complex distribution through a series of denoising steps. This probabilistic framework provides a robust mechanism for image synthesis, capable of capturing intricate details and structures within the data.

Concurrently, the concept of Deep Image Prior (DIP) has emerged as a powerful tool for image restoration tasks. Unlike traditional methods that rely on large datasets for training, DIP leverages the architecture of convolutional neural networks to act as a prior. This enables the network to capture the statistical properties of a single image, allowing for effective image reconstruction and denoising without the need for extensive training data. By optimizing the network parameters solely on the corrupted image, DIP has demonstrated remarkable results in various image restoration tasks, highlighting the implicit regularization capabilities of neural networks.

The integration of DDPM and DIP within the Truncated Diffusion Probabilistic Models (TDPM) framework presents a novel approach to generative modeling. By combining the strengths of DDPM's iterative denoising process with DIP's powerful prior, TDPM-DIP aims to enhance the quality and efficiency of image generation and restoration. This hybrid approach leverages the probabilistic nature of diffusion models and the structural regularization of DIP, offering a promising direction for future research in generative models and image processing.

2 Background

2.1 Denoising Diffusion Probabilistic Models

Denoising Diffusion Probabilistic Models (DDPM) (Ho et al., 2020) [2] represent a significant advancement in generative modeling, leveraging principles from diffusion processes and prob-

abilistic modeling to achieve high-quality image generation. Unlike traditional generative adversarial networks (GANs) and variational autoencoders (VAEs), DDPMs introduce a novel framework for learning data distributions through a sequence of denoising steps.
Mathematically, the forward diffusion process is defined as:

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}) \tag{1}$$

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}) \tag{2}$$

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)\mathbf{I}) \tag{3}$$

where x_t is the noisy image at time step t, $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_t \alpha_t$.

The reverse process can be parameterized by a neural network that predicts the mean and variance of the reverse transition at each step:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{4}$$

Here, μ_θ and Σ_θ are learned parameters, with μ_θ representing the predicted mean and Σ_θ the variance.

2.2 Truncated Diffusion Probabilistic Models

Diffusion-based generative models operate by gradually corrupting the data distribution into a simple noise distribution through a forward diffusion chain, and subsequently generating data by learning the reverse diffusion process. Despite their impressive performance, these models are computationally intensive due to the necessity of performing a large number of forward and reverse diffusion steps. This high computational cost stems from the need to ensure that each noise injection step is sufficiently small, maintaining the assumption that both the diffusion and denoising processes are Gaussian.

To address these inefficiencies, Truncated Diffusion Probabilistic Models (Zheng et al., 2023) [1] shorten the diffusion trajectory by learning an implicit distribution to start the reverse diffusion process. Instead of relying on a tractable noise distribution, the model truncates the forward diffusion chain and learns to generate data from a hidden noisy-data distribution, thus requiring fewer reverse steps. By leveraging an implicit generative distribution, TDPM can start the reverse diffusion process more efficiently.

In the proposed Truncated Diffusion Probabilistic Model (TDPM), the diffusion chain is truncated to the first T_{trunc} steps:

$$\{\beta_1, \beta_2, \ldots, \beta_{T_{trunc}}\} \subset \{\beta_1, \beta_2, \ldots, \beta_T\} \tag{5}$$

This truncation reduces the computational burden while maintaining the quality of the generated samples.

To handle the unknown distribution at the truncation point, a deep neural network-based generator G_ψ is introduced to approximate this distribution:

$$\mathbf{x}_{T_{trunc}} = G_\psi(\mathbf{z}), \quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{6}$$

When $\mathbf{x}_{T_{trunc}}$ is generated from the implicit generative model, we can start the reverse diffusion chain at time T_{trunc} from the implicit generated distribution $p_\psi(\mathbf{x}_{T_{trunc}})$.

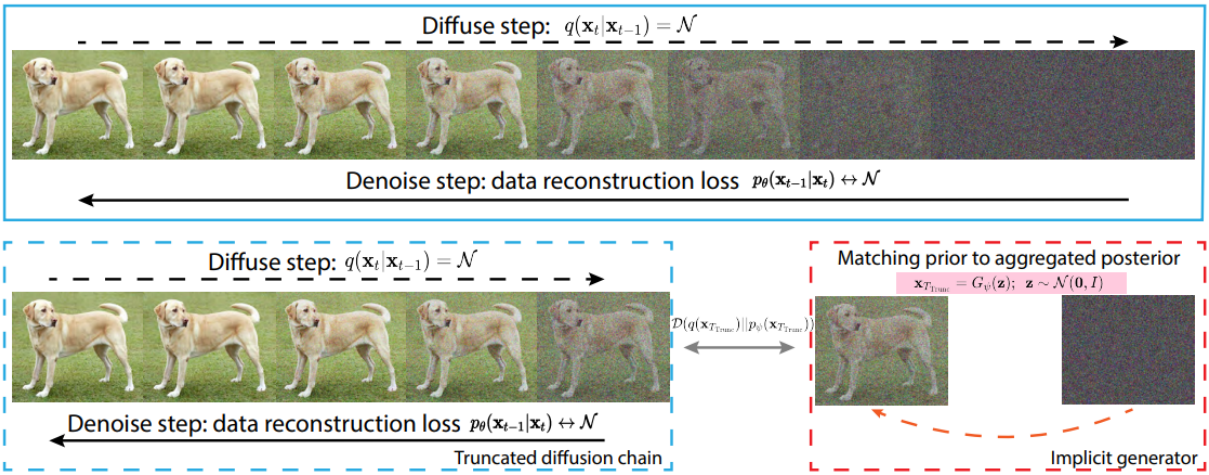

Figure 1. The full process of Truncated Diffusion Probabilistic Models. [1]

2.3 Deep Image Prior

The concept of the Deep Image Prior (DIP) [3] emerged from the realization that the structure of a deep convolutional neural network (ConvNet) alone, without any learning, can serve as an effective prior for various image restoration tasks.

In the DIP framework, the ConvNet is used as a generator that maps a random noise input to an image. The network parameters are then optimized to minimize a loss function defined on the given degraded image, effectively fitting the network to restore the image.

The algorithm of DIP is particularly suitable for tasks like image denoising, super-resolution, and inpainting, providing high-quality restorations based on the network's innate biases alone.

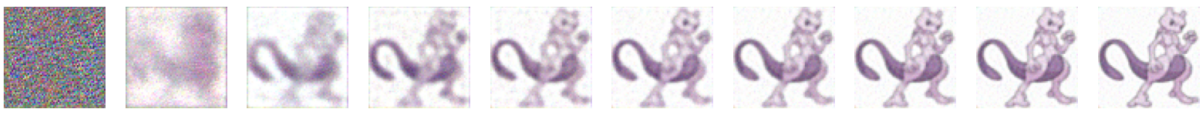

Figure 2. The full process of Deep Image Prior from the noisy image to the ground truth.

3 Truncated Diffusion and Deep Image Prior (TDPM-DIP)

This report introduce the idea of combining DDPM and DIP within the TDPM framework to accelerate the sampling speed. By truncating the diffusion chains and apply DIP framework as the implicit generator G_ψ, This algorithm not only can significantly reduce the reverse diffusion steps but also maintaining or even improving the quality of images in denoising and generation tasks.

3.1 Algorithm Explaination

In this report, TDPM-DIP framework can be split into 3 algorithms. In the DDPM training process, each image $\mathbf{x_0}$ sampled from the dataset will be added to a random noise ϵ by the noise scheduler, and $\mathbf{x_t}$ is dependent on the chosen time step t. We will train an image denoiser $\epsilon_\theta(x(t), t)$ which is built by the **U-Net**[4] from **HuggingFace Diffusers**.

While training DIP as the implicit generator, we need to sample a Gaussian Noise $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and a specific truncated time step T_{trunc}. We can acquire the truncated noisy image $\mathbf{x}_{\mathbf{T}_{\text{trunc}}}$ from the noise scheduler provided by DDPM. The algorithm uses **Mean-Square Error** for the loss function that can calculate the distance $\mathcal{D}(q(\mathbf{x}_{\mathbf{T}_{\text{trunc}}})||p_\psi(\mathbf{x}_{\mathbf{T}_{\text{trunc}}}))$, where $\mathcal{D}(q||p)$ is a statistical distance between distributions q and p:

$$\tilde{\mathcal{L}}_{DIP} \quad = \quad \| \left(\sqrt{\bar{\alpha}_{\mathbf{T}_{\text{trunc}}}} \mathbf{x_0} + (\sqrt{1 - \bar{\alpha}_{\mathbf{T}_{\text{trunc}}}}) \mathbf{z} \right) - G_\psi(\mathbf{z}) \|^2 \quad := \quad \mathcal{D}(q(\mathbf{x}_{\mathbf{T}_{\text{trunc}}})||p_\psi(\mathbf{x}_{\mathbf{T}_{\text{trunc}}})) \quad (7)$$

where T_{trunc} is the truncated time step, G_ψ is the DIP model whose parameter parameterized by ψ, and $\mathbf{z}$ is the Gaussian Noise.

The sampling process starts by randomly sampling a Gaussian noise $\mathbf{z}$ (same as **Algorithm 2**). Through the denoising steps of the pre-trained DIP, we can get $\hat{\mathbf{x}}_{\mathbf{T}_{\text{trunc}}}$ in fewer steps than the full DDPM reverse steps. Thus, the TDPM-DIP sampler can be used to solve the reverse diffusion steps from the truncated point T_{trunc} to the noise-free image step.

This report summarize the training and sampling processes of TDPM-DIP in Algorithm 1, Algorithm 2, and Algorithm 3 respectively.

Algorithm 1 Training DDPM

repeat
  $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
  $t \sim Uniform(\{1, \ldots, T_{trunc}\})$
  $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take gradient step on
    $\nabla_\theta \|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, t)\|^2$
until converged

Algorithm 2 Training DIP as G_ψ

Require:
Gaussian Noise $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
Truncated noisy image $\mathbf{x}_{\mathbf{T}_{\text{trunc}}}$
Lemma: $\mathbf{x}_{\mathbf{T}_{\text{trunc}}} = \sqrt{\bar{\alpha}_{\mathbf{T}_{\text{trunc}}}} \mathbf{x_0} + (\sqrt{1 - \bar{\alpha}_{\mathbf{T}_{\text{trunc}}}}) \mathbf{z}$
Ensure: Predicted Truncated noisy image $\hat{\mathbf{x}}_{\mathbf{T}_{\text{trunc}}}$
repeat
  $\mathbf{x}^{(i)} = G_\psi^{(i)}(\mathbf{z})$
  $MSE_{loss} = \|\mathbf{x}_{\mathbf{T}_{\text{trunc}}} - \mathbf{x}^{(i)}\|^2$
  Update $G_\psi^{(i)}$ using the ADAM algorithm.
  $i \leftarrow i + 1$
until $i > i_{max}$
return $G_\psi^{(i_{max})}$

Algorithm 3 Sampling

Gaussian Noise $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ▷ same as **Algorithm 2**
$\hat{\mathbf{x}}_{\mathbf{T}_{\text{trunc}}} = G_\psi(\mathbf{z})$
for $t = T_{\text{trunc}}$ to 1 **do**
  $\mathbf{z_t} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z_1} = \mathbf{0}$
  $\hat{\mathbf{x}}_{\mathbf{t-1}} = \frac{1}{\sqrt{\alpha_t}} \left(\hat{\mathbf{x}}_\mathbf{t} - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(\hat{\mathbf{x}}_\mathbf{t}, t) \right) + \beta_t \mathbf{z_t}$
end for
return $\hat{\mathbf{x}}_\mathbf{0}$

3.2 Potential Benefits and Limitations

The integration of Denoising Diffusion Probabilistic Models (DDPM) and Deep Image Prior (DIP) offers a compelling solution that combines the strengths of both methods, but there are also some limitations on the proposed TDPM-DIP framework.
Here are some potential benefits of TDPM-DIP:

1. **Enhanced Image Quality.** The combination of DDPM's iterative refinement with the strong priors embedded in the DIP framework can result in superior image quality. The DIP can provide a strong initialization for the DDPM, leading to faster convergence and better final results.

2. **Accelerate Sampling Speed.** By integrating the denoising diffusion process with the deep image prior, the combined model can achieve efficient image generation and restoration without large amounts of denoising steps for the reverse process in DDPM. The random Noise can be first denoised by DIP in fewer steps, thus TDPM-DIP can reduce the full reverse diffusion chain steps.

3. **Versatility and Robustness.** TDPM-DIP can adapt to a wider range of image restoration and generation tasks, leveraging the strengths of both methods to handle diverse scenarios more effectively.

4. **Improved Interpretability.** TDPM-DIP can offer better interpretability of the generated images, as the DIP component can elucidate the role of network architecture in the generation process.

Here are some potential limitations of TDPM-DIP:

1. **Complexity in Implementation** TDPM-DIP may be more complex to implement and tune compared to using DDPM or DIP alone. The integration of two distinct methods requires careful consideration of their respective hyperparameters and interaction dynamics.

2. **Limited Generalization Without Training** The DIP approach inherently does not generalize as well to completely unseen image types or styles since it does not learn from a wide variety of data. Although combining with DDPM can mitigate this to some extent, the reliance on the intrinsic structure of ConvNets in DIP might still limit the generalization capabilities of the TDPM-DIP in some contexts. The random noise $\mathbf{z}$ in **Algorithm 2** and **Algorithm 3** need to be the same.

4 Experiments

This report aims to demonstrate that TDPM-DIP can generate high quality faster by using fewer steps of reverse diffusion. The experiment uses the image dataset of **pokémon** to test the proposed method and compare it with standalone DDPM and DIP methods. There are two tasks to test the TDPM-DIP method: **Image Generation** and **Image Denoising**.

For **Image Generation**, we use FID (lower is better) to measure the fidelity and diversity of the generated images. We use the **pokémon** dataset [5] provided by HugGAN Community in HuggingFace. The images consist of 64×64 pixels for pokémon.

For **Image Denoising**, we use PSNR (higher is better) and SSIM (higher is better) to measure the similarity of the restored images and the ground truth images. We use the **pokémon** dataset [5] provided by HugGAN Community in HuggingFace. The images consist of 64 × 64 pixels for pokémon.

4.1 Image Generation

We first look at the results of image generation on the Pokémons. We use DDPM as our baseline and compare it with the TDPM-DIP framework. For the original DDPM, we use 1000 steps for the reverse diffusion chain. For the TDPM-DIP, we use DIP for the implicit generator G_ψ with training (denoising) steps equivalent to 100 reverse steps, and for the truncated DDPM, we use extended $T_{trunc} = 10$ steps for the rest reverse process.

Results in Table 1 show that TDPM-DIP outperformed DDPM. Notably, TDPM-DIP significantly improved DDPM's performance, decreasing FID from 69.05 to 20.12. TDPM-DIP also significantly improved computational efficiency compared to DDPM, reducing reverse sampling steps from 1000 steps to 110 steps per image and inference time from 38 minutes to 7 seconds on T4 GPU.

Method	FID@1k ↓	Reverse Steps↓	Inference Time↓	Speed Up (Steps)↑	Speed Up (Time)↑
DDPM	69.05	1000	38s	x1	x1
TDPM-DIP	**20.12**	**110**	**7s**	**x9.09**	**x5**

Table 1: Results of image generation from pure noise on Pokémon, with the best score in each metrics marked in bold.

Figure 3. Qualitative results of the Image Generation task by TDPM-DIP, with T_{trunc} = 10, 50, and 100. Each group was generated from TDPM-DIP (left) and implicit DIP as G_ψ (right).

4.2 Image Denoising

Since the TDPM-DIP framework will capture the structural image prior through the implicit generator, we can use this property for any image denoising or restoration task. We will also compare it with the standalone DIP and DDPM methods on this task.

The results presented in Table 2 indicate that TDPM-DIP significantly outperformed the DDPM models, as well as the other denoising method, Deep Image Prior (DIP). In terms of computational efficiency, TDPM-DIP reduced the inference time by approximately 5 times compared to DDPM on T4 GPU. By implementing an efficient denoising strategy, TDPM-DIP significantly improves the PSNR and SSIM on Pokemon, achieving the best performance between standalone DDPM and DIP methods.

Method	PSNR@200 ↑	SSIM@200 ↑	Reverse Steps↓	Inference Time↓
DDPM	13.40	0.369	600	32s
DIP	32.12	0.948	100	7s
TDPM-DIP	**32.63**	**0.970**	110	7s

Table 2: Results of Image Denoising task, with the best score in each metric marked in bold. Each noisy image is at the noisy level $t = 600$ on the Pokémon dataset.

Figure 4 shows the Zapdos images generated by various image denoising methods from the same noisy level $t = 600$. Notably, the Zapdos body and background appear clear and sharp in the image generated by TDPM-DIP, while the other methods exhibit noticeable disturbance and noise in the image.

Normal Zapdos Image | DIP method | DDPM method | **TDPM-DIP method**

Figure 4. Qualitative results of the Pokémon image denoising task.

4.3 Training Details

TDPM-DIP adopted the UNet architecture from the **HuggingFace Diffusers library**. DDPM model was built from **UNet2DModel** with 6 downsampling blocks and 6 upsampling blocks, while DIP model was built from the same architecture but with 4 downsampling blocks and 4 upsampling blocks. The training batch size was set to 32 and the DDPM model was trained for 100 epochs. On the other side, DIP model was set to train on any specific image for 100 iterations, which can be seen as the partial "denoising" steps in the reverse process of TDPM-DIP. The AdamW optimizer with a learning rate of 10^{-4} was used for training DDPM. All experiments were conducted on Google Colab platform with one NVIDIA T4 GPU, with 15GB memory.

5 Ablation Studies and Analysis

This report has conducted two ablation studies to investigate the impact of different components on TDPM-DIP. For simplicity, We will only demonstrate these two important impacts on the **image denoising** task, while the ablation studies can still be conducted on the image generation or other interesting image translation tasks.

5.1 Impact of Truncated Step Selection

We investigated the impact of different truncated time steps on the performance of TDPM-DIP. For this ablation study, we conducted experiments using 0, 5, 10, and 50 truncated time steps for TDPM-DIP on the image denoising task. The results in Table 3 indicate that the number of truncated time steps significantly affects the performance of TDPM-DIP. Optimal performance for image denoising is achieved with 5 steps for the truncated point. From these findings, we can find that the selection of truncated steps is important for different needs. For the quick reverse process, we can select the fewer truncated steps for computational efficiency. On the other hand, we can select around 5 to 20 middle truncated time steps for image quality. For the higher truncated steps, the generated images will be more creative and interesting and it's useful for image generation or style translation. Hence, there is a trade-off between computational efficiency and image quality.

Truncated Time Step	PSNR@200 ↑	SSIM@200 ↑	Reverse Steps↓	Inference Time↓
$\mathbf{T_{trunc} = 0}$ **(DIP)**	32.12	0.948	**100**	7s
$\mathbf{T_{trunc} = 5}$	**32.77**	**0.970**	105	7s
$\mathbf{T_{trunc} = 10}$	32.63	**0.970**	110	7s
$\mathbf{T_{trunc} = 50}$	28.70	0.937	150	7s

Table 3: The impact of different truncated steps in TDPM-DIP on the Image Denoising Task.

5.2 Impact of Noise Scheduler

Table 4 illustrates the impact of the two proposed noise schedulers on the performance of TDPM-DIP and other methods. The cosine noise scheduler seems to be a better choice for the image denoising task on all the proposed methods. Hence, the discrepancy in the results between linear and cosine noise schedulers is obvious for our TDPM-DIP framework. Although these improvements are significant on the image denoising task, selecting an appropriate noise scheduler to the specific task is still an important issue, and also needs more experiments and discussions.

Noise Scheduler	Method	PSNR@200 ↑	SSIM@200 ↑	Reverse Steps↓	Inference Time↓
Linear	DDPM	13.40	0.369	600	32s
	DIP	32.12	0.948	100	7s
	TDPM-DIP	**32.63**	**0.970**	110	7s
Cosine	DDPM	18.68	0.659	600	32s
	DIP	34.35	0.962	100	7s
	TDPM-DIP	**34.76**	**0.979**	110	7s

Table 4: The impact of two proposed noise schedulers on the Image Denoising Task.

6 Discussion and Conclusion

This report introduces TDPM-DIP, a simple yet effective approach that accelerates the sampling of DDPM while significantly improving the image quality. Our evaluation demonstrates that TDPM-DIP achieves the best performance across image generation and image denoising tasks on the Pokémon dataset.

Limitations and Future Work The evaluation dataset we used is only for the Pokémons, the evaluation scores might be slightly difference on other datasets. Also, some image-to-image tasks like Super-Resolution, Image Inpainting, and Translation were not conducted. Future work should focus on experimenting the diverse image-to-image tasks and using image datasets from different fields for evaluation and comparison with the performance.

References

[1] Zheng, Huangjie and He, Pengcheng and Chen, Weizhu and Zhou, Mingyuan. Truncated Diffusion Probabilistic Models. *arXiv preprint arXiv:2202.09671*, 2022.

[2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[3] Ulyanov, Dmitry and Vedaldi, Andrea and Lempitsky, Victor. Deep Image Prior. *arXiv:1711.10925*, 2017.

[4] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI.

[5] HuggingFace Dataset Url: https://huggingface.co/datasets/huggan/pokemon