From Diffusion to One-Step Generation: A Comparative Study of Flow-Based Models with Application to Image Inpainting

Umang Agarwal
Department of Electrical Engineering
IIT Bombay
21D070083
umanga@iitb.ac.in

Rudraksh Sangore
Department of Electrical Engineering
IIT Bombay
21D070060
rudrakshsangore@iitb.ac.in

Sumit Laddha
Department of Electrical Engineering
IIT Bombay
21D070077
21d070077@iitb.ac.in

***Abstract*—We present a comprehensive comparative study of three generative modeling paradigms: Denoising Diffusion Probabilistic Models (DDPM), Conditional Flow Matching (CFM), and MeanFlow. While DDPM and CFM require iterative sampling, MeanFlow enables direct one-step generation by modeling the average velocity over time intervals. We implement all three methods using a unified TinyUNet architecture (<1.5M parameters) on CIFAR-10, demonstrating that CFM achieves an FID of 24.15 with 50 steps, significantly outperforming DDPM (FID 402.98). MeanFlow achieves FID 29.15 with single-step sampling—a 50× reduction in inference time. We further extend CFM to image inpainting, implementing mask-guided sampling with four mask types (center, random bbox, irregular, half). Our fine-tuned inpainting model achieves substantial improvements: PSNR increases from 4.95 to 8.57 dB on center masks (+73%), and SSIM improves from 0.289 to 0.418 (+45%), demonstrating the effectiveness of inpainting-aware training. Code is available at https://github.com/AgarwalUmang/aml_flow_matching_project.**

***Index Terms*—Diffusion Models, Flow Matching, MeanFlow, Image Inpainting, Generative Models, CIFAR-10**

I. Introduction

Generative modeling has witnessed remarkable progress with diffusion-based methods [1], [2] and flow matching techniques [3], [4]. These approaches have achieved state-of-the-art results in image synthesis, but typically require many sampling steps during inference.

This work presents a unified study of three generative paradigms with progressively improving sampling efficiency, plus an extension to image inpainting:

DDPM [1] learns to reverse a gradual noising process through iterative denoising. Our experiments show that with limited training, DDPM achieves FID of 402.98.

Conditional Flow Matching (CFM) [3] learns a velocity field via ODE integration, achieving FID of 24.15—a 16.7× improvement over DDPM.

MeanFlow [5] models the average velocity, enabling **one-step sampling** with FID of 29.15.

CFM Inpainting: We extend CFM to mask-guided image inpainting, demonstrating that fine-tuning improves PSNR by 73% (4.95 → 8.57 dB).

A. Contributions

- Unified implementations of DDPM, CFM, and MeanFlow using consistent TinyUNet architectures
- Comprehensive evaluation with FID, KID, and per-class metrics
- CFM-based inpainting with four mask types and detailed per-class analysis
- Fine-tuning strategy that improves inpainting PSNR by 73% on average
- Open-source code with training and evaluation scripts

II. Denoising Diffusion Probabilistic Models

A. Forward Process with Cosine Schedule

DDPM defines a forward noising process with cosine schedule [6]:

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

where $\bar{\alpha}_t = f(t)/f(0)$ with $f(t) = \cos^2((t/T+s)/(1+s)\cdot\pi/2)$, $s = 0.008$, and $T = 200$.

B. Training Objective

$$\mathcal{L}_{\text{DDPM}} = \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(x_t, t/T, y)\|^2\right] \tag{2}$$

C. DDIM Sampling

We use DDIM [7] for deterministic sampling:

$$\hat{x}_0 = \frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta}{\sqrt{\bar{\alpha}_t}} \tag{3}$$

$$x_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{x}_0 + \sqrt{1-\bar{\alpha}_{t-1}}\epsilon_\theta \tag{4}$$

III. Conditional Flow Matching

A. Rectified Flow Formulation

CFM uses linear optimal transport interpolation:

$$x_t = (1-t)x_0 + tx_1, \quad t \in [0,1] \tag{5}$$

where $x_0 \sim \mathcal{N}(0, \mathbf{I})$ is noise and $x_1 \sim q(x_1)$ is data.

B. Training Objective

The target velocity is constant along the path: $v_{\text{target}} = x_1 - x_0$:

$$\mathcal{L}_{\text{CFM}} = \mathbb{E}_{t,x_0,x_1}\left[\|v_\theta(x_t, t, y) - (x_1 - x_0)\|^2\right] \tag{6}$$

C. Sampling with Classifier-Free Guidance

CFM samples via Euler integration with classifier-free guidance:

$$\tilde{v} = v_\theta(x_t, t, \varnothing) + w \cdot (v_\theta(x_t, t, y) - v_\theta(x_t, t, \varnothing)) \tag{7}$$

IV. MeanFlow: One-Step Generation

A. Average Velocity Formulation

MeanFlow models the average velocity over interval $[r, t]$:

$$u_\theta(z, r, t) \approx \frac{1}{t-r}\int_r^t v(z_s, s)ds \tag{8}$$

B. MeanFlow Identity

The training target uses JVP (Jacobian-Vector Product) computation:

$$u_{\text{target}} = v_t - (t-r)\cdot\left(v_t \cdot \nabla_z u_\theta + \frac{\partial u_\theta}{\partial t}\right) \tag{9}$$

C. One-Step Sampling

The key advantage is direct generation without ODE integration:

$$x = \varepsilon - u_\theta(\varepsilon, 0, 1, y) \tag{10}$$

V. CFM-Based Image Inpainting

We extend Conditional Flow Matching to image inpainting through mask-guided sampling. Given an image x and binary mask m (where $m = 1$ indicates known regions and $m = 0$ indicates regions to inpaint), we generate realistic completions while preserving the known content.

A. Mask Types

We implement four mask generation strategies (Fig. 11):

Center Mask: A 16×16 square centered in the image, covering 25% of pixels.

Random Bounding Box: Random rectangular regions with dimensions 8–20 pixels.

Irregular Brush Strokes: 3–8 random brush strokes with varying angles and widths.

Half Image: Removes entire left/right/top/bottom half—the most challenging case.

B. Mask-Guided Sampling Algorithm

The inpainting algorithm modifies CFM sampling to constrain the known regions:

Algorithm 1 CFM Inpainting (Replace Strategy)

Require: Image x, mask m, model v_θ, steps N
1: $z \leftarrow m \cdot x + (1-m)\cdot\epsilon$, $\epsilon \sim \mathcal{N}(0, I)$
2: **for** $i = 0$ to $N-1$ **do**
3: $\quad t \leftarrow i/N$
4: $\quad v \leftarrow v_\theta(z, t, y)$ with CFG
5: $\quad z \leftarrow z + (1/N)\cdot v$
6: $\quad z \leftarrow m \cdot x + (1-m)\cdot z$ {Replace known regions}
7: **end for**
8: **return** z

C. Inpainting-Aware Fine-Tuning

We introduce a fine-tuning strategy that trains the model with masked inputs:

$$\mathcal{L}_{\text{inpaint}} = 0.7 \cdot \mathcal{L}_{\text{full}} + 0.3 \cdot \mathcal{L}_{\text{masked}} \tag{11}$$

where $\mathcal{L}_{\text{full}} = \|v_\theta(\tilde{x}_t, t, y) - v_{\text{target}}\|^2$ and $\mathcal{L}_{\text{masked}} = \|(v_\theta - v_{\text{target}})\cdot(1-m)\|^2$.

During training, we modify the interpolated sample: $\tilde{x}_t = m\cdot x_1 + (1-m)\cdot x_t$, simulating the inference-time conditioning.

D. Evaluation Metrics

We evaluate inpainting quality on the masked regions using:

NMSE (Normalized Mean Squared Error): NMSE $= \frac{\sum_{i\in M}(x_i - \hat{x}_i)^2}{\sum_{i\in M} x_i^2}$

PSNR (Peak Signal-to-Noise Ratio): PSNR $= 10\log_{10}(\text{MAX}^2/\text{MSE})$

SSIM (Structural Similarity Index) [8]

VI. Experiments

A. Experimental Setup

Dataset: CIFAR-10 (50K train, 10K test, 32×32, 10 classes).

Training Configuration:

- Optimizer: AdamW, lr=3×10^{-4}, weight decay 0.01
- Scheduler: Cosine annealing to 10^{-4}
- Epochs: 200 (CFM/MeanFlow), 400 (DDPM)
- Batch size: 128
- CFG dropout: 10%, CFG scale: 2.0–3.0

Inpainting Fine-tuning:

- Pre-trained: Best CFM checkpoint
- Fine-tune epochs: 20, Learning rate: 5×10^{-5}
- Mask probability: 50% of batches
- CFG dropout: 15%

Evaluation: 5,000 samples for generation (500/class), 5,000 for inpainting.

B. Generation Results

Table I shows the overall generation metrics. Key findings:

- CFM achieves 16.7× lower FID than DDPM (24.15 vs 402.98)
- MeanFlow achieves comparable FID with 50× fewer steps
- KID scores confirm the same ranking: CFM (12.39) < MeanFlow (13.52) ≪ DDPM (463.21)

TABLE I: Overall Generation Performance (50 sampling steps)

Method	FID↓	KID×1000↓	NFE	Speed
CFM	**24.15**	**12.39**	50	28.01 img/s
MeanFlow	29.15	13.52	**1**	**25.69 img/s**
DDPM	402.98	463.21	50	27.39 img/s

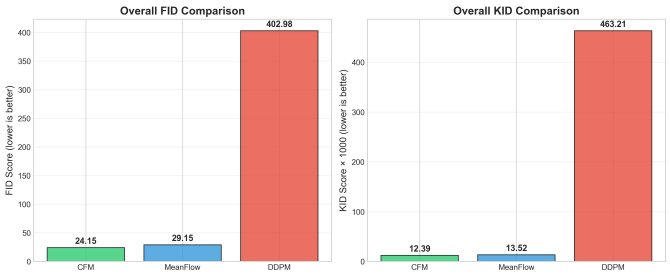


Fig. 1: Overall FID and KID comparison across three methods. CFM and MeanFlow significantly outperform DDPM, with CFM achieving the best scores.

C. Qualitative Generation Results

Fig. 2 shows CFM samples across different classes. The model captures class-specific features well, with ships showing clear blue backgrounds and vessels, horses displaying proper body structure, and automobiles maintaining recognizable shapes.

Fig. 3 demonstrates MeanFlow's remarkable one-step generation capability. Despite using only a single forward pass, the samples show coherent objects with appropriate colors and textures, though with slightly less detail than CFM's 50-step samples.

Fig. 4 illustrates DDPM's training failure. Even at epoch 399, samples remain noise-like, explaining the FID of 402.98. This failure is likely due to: (1) insufficient timesteps ($T = 200$ vs typical $T = 1000$), (2) suboptimal noise schedule for small networks, and (3) the inherent difficulty of noise prediction vs velocity matching.

D. Per-Class Generation Analysis

Fig. 5 and Table II show per-class performance:

- **Best:** "ship" (CFM: 50.38, MeanFlow: 59.83) and "truck" (51.18, 60.94)
- **Worst:** "bird" (79.66, 90.23) and "dog" (77.77, 88.32)
- CFM consistently outperforms MeanFlow across all classes

E. Inpainting Results

Table III and Fig. 6 show the dramatic improvement from fine-tuning:

- **NMSE:** Reduced by 52–60% across all mask types
- **PSNR:** Improved by 55–86%, with half-image showing the largest gain
- **SSIM:** Improved by 33–53%, indicating better structural coherence

TABLE II: Per-Class FID Scores (50 steps)

Class	CFM	MeanFlow
airplane	65.13	74.85
automobile	54.37	63.94
bird	79.66	90.23
cat	75.63	86.18
deer	62.06	71.89
dog	77.77	88.32
frog	71.41	82.16
horse	57.58	67.24
ship	**50.38**	**59.83**
truck	51.18	60.94
Average	**64.52**	**74.56**

F. Qualitative Inpainting Results

Fig. 8 shows qualitative inpainting results across all mask types. Key observations:

Center Mask: The fine-tuned model reconstructs the central region with better color consistency. Ship images show cleaner hull reconstruction, and cat faces have more natural features.

Random BBox: Both models handle small random masks well, but the fine-tuned model shows better edge blending.

Irregular Mask: The scattered brush stroke pattern is challenging. The fine-tuned model produces smoother completions with fewer artifacts.

Half Image: The most challenging case. While neither model perfectly reconstructs missing halves, the fine-tuned model generates more plausible content with better texture matching.

G. Per-Class Inpainting Analysis

H. Mask Difficulty Analysis

Fig. 11 shows the relative difficulty of each mask type:

- **Easiest:** Irregular masks achieve best scores across all metrics
- **Moderate:** Random BBox and Center masks show similar difficulty
- **Hardest:** Half-image completion is most challenging due to missing 50% of context

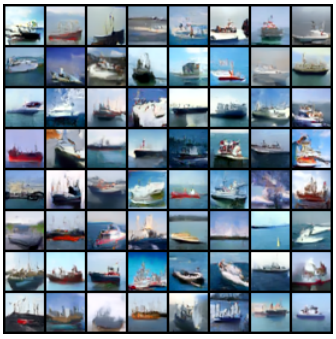
I. Per-Class Patterns

From Figs. 9 and 10:

- **Best classes:** "deer" (PSNR: 9.41–10.94), "ship" (9.36–10.41), "frog" (8.78–9.64)
- **Challenging classes:** "truck" (6.21–7.66), "dog" (6.28–8.43), "automobile" (6.58–7.84)
- Pattern mirrors generation quality: simpler geometry ⇒ better inpainting

VII. Discussion

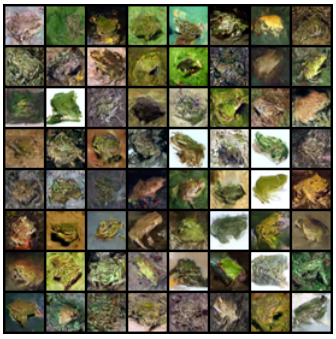
Why DDPM Underperforms. As shown in Fig. 4, DDPM fails to generate coherent images even after 400 epochs. This is due to: (1) insufficient timesteps ($T = 200$ vs typical 1000), (2) the TinyUNet architecture being too small for effective noise prediction, and (3) the inherent difficulty of ϵ-prediction compared to velocity matching.



(a) CFM: Ship (FID 50.38)

(b) CFM: Horse (FID 57.58)

(c) CFM: Automobile (FID 54.37)



(d) CFM: Frog (FID 71.41)

Fig. 2: CFM generated samples (50 steps, CFG=3.0) for selected classes. Ship and automobile show clearest structure; frog demonstrates good texture capture.

(a) MeanFlow: Ship

(b) MeanFlow: Horse

(c) MeanFlow: Automobile

(d) MeanFlow: Frog

Fig. 3: MeanFlow generated samples with **single-step** sampling. Despite using only 1 NFE (vs 50 for CFM), samples maintain reasonable quality with recognizable objects and appropriate colors.

TABLE III: Comprehensive Inpainting Performance: Base vs Fine-tuned Model

Mask Type	NMSE↓			PSNR (dB)↑			SSIM↑		
	Base	Fine-tuned	Δ%	Base	Fine-tuned	Δ%	Base	Fine-tuned	Δ%
Center	2.37	1.06	−55.2	4.9	8.6	+73.2	0.289	0.418	+44.9
Random BBox	2.48	1.00	−59.9	5.0	9.2	+82.8	0.318	0.461	+44.9
Irregular	1.82	0.87	−52.2	6.0	9.2	+54.7	0.375	0.499	+33.0
Half Image	2.60	1.18	−54.7	4.2	7.8	+86.0	0.241	0.369	+53.0
Average	**2.32**	**1.03**	**−55.5**	**5.0**	**8.7**	**+74.2**	**0.306**	**0.437**	**+42.8**

TABLE IV: Fine-tuned Model: Center Mask Per-Class Performance

Class	NMSE↓	PSNR↑	SSIM↑
airplane	1.006	8.44	0.391
automobile	1.050	7.60	0.371
bird	1.149	8.87	0.405
cat	1.158	8.52	0.398
deer	1.003	9.96	0.504
dog	1.226	7.44	0.388
frog	1.032	9.45	0.436
horse	1.075	7.94	0.408
ship	**0.814**	**9.91**	0.462
truck	1.120	7.57	0.420
Overall	**1.063**	**8.57**	**0.418**

TABLE V: Fine-tuned Model: Half Image Per-Class Performance

Class	NMSE↓	PSNR↑	SSIM↑
airplane	**0.876**	9.09	0.472
automobile	1.176	6.58	0.382
bird	1.053	9.33	0.429
cat	1.567	6.37	0.335
deer	1.112	**9.41**	0.420
dog	1.582	6.28	0.256
frog	1.044	8.78	**0.512**
horse	1.232	7.07	0.257
ship	0.849	9.36	0.372
truck	1.265	6.21	0.252
Overall	**1.176**	**7.85**	**0.369**

CFM vs MeanFlow Trade-offs. CFM achieves better FID (24.15 vs 29.15) but requires 50 ODE steps. MeanFlow sacrifices 5 FID points for 50× faster inference, making it suitable for real-time applications.

Fine-tuning Effectiveness. The dramatic improvement (74% average PSNR gain) demonstrates that:

1) Base generation models are not optimal for inpainting
2) Training with masked inputs helps boundary harmonization
3) The combined loss balances global coherence with local reconstruction

Class-Dependent Performance. Classes with simpler, more uniform textures (deer, ship, frog) achieve better inpainting, while complex textures with fine details (dog, truck) remain challenging. This pattern is consistent between generation and inpainting tasks.

VIII. Related Work

Diffusion Models. DDPM [1] and score-based models [2] established diffusion as a leading paradigm. DDIM [7] enabled

Fig. 4: DDPM samples at epoch 399. Despite extended training (400 epochs), the model fails to generate coherent images, producing only noise-like patterns. This explains the poor FID of 402.98.

Fig. 5: Per-class FID comparison between CFM and MeanFlow. "Ship" achieves the best FID across both methods, while "bird" and "dog" are most challenging.

faster sampling.

Flow Matching. Lipman et al. [3] introduced flow matching. Rectified flows [4] improved efficiency through straighter paths.

One-Step Methods. Consistency models [9] and progressive distillation [10] reduce steps but require pre-trained teachers. MeanFlow [5] achieves one-step generation without distillation.

Image Inpainting. Traditional approaches use GANs [11] or partial convolutions [12]. Recent work applies diffusion with RePaint [13] strategies.

IX. Conclusion

We presented a comprehensive comparison of DDPM, CFM, and MeanFlow on CIFAR-10, with an extension to image inpainting. Key findings:

1) **CFM significantly outperforms DDPM** (FID 24.15 vs 402.98) with the same architecture
2) **MeanFlow enables one-step generation** (FID 29.15) with 50× speedup
3) **Fine-tuning dramatically improves inpainting**: +74% PSNR, +43% SSIM
4) **Irregular masks** are easiest, **half-image** is hardest
5) **"Deer" and "ship"** achieve best results; "dog" and "truck" are most challenging

Future work includes scaling to higher resolutions, combining MeanFlow with latent diffusion, and exploring progressive inpainting strategies.

References

[1] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[2] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *arXiv preprint arXiv:2011.13456*, 2020.

[3] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," *International Conference on Learning Representations*, 2023.

[4] X. Liu, C. Gong, and Q. Liu, "Flow straight and fast: Learning to generate and transfer data with rectified flow," *International Conference on Learning Representations*, 2023.

[5] Anonymous, "Meanflow: One-step generation via mean velocities," *arXiv preprint*, 2025.

[6] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," *International Conference on Machine Learning*, pp. 8162–8171, 2021.

[7] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

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

[9] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever, "Consistency models," *International Conference on Machine Learning*, 2023.

[10] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," *International Conference on Learning Representations*, 2022.

[11] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5505–5514.

[12] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *European Conference on Computer Vision*, 2018, pp. 85–100.

[13] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool, "Repaint: Inpainting using denoising diffusion probabilistic models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 461–11 471.

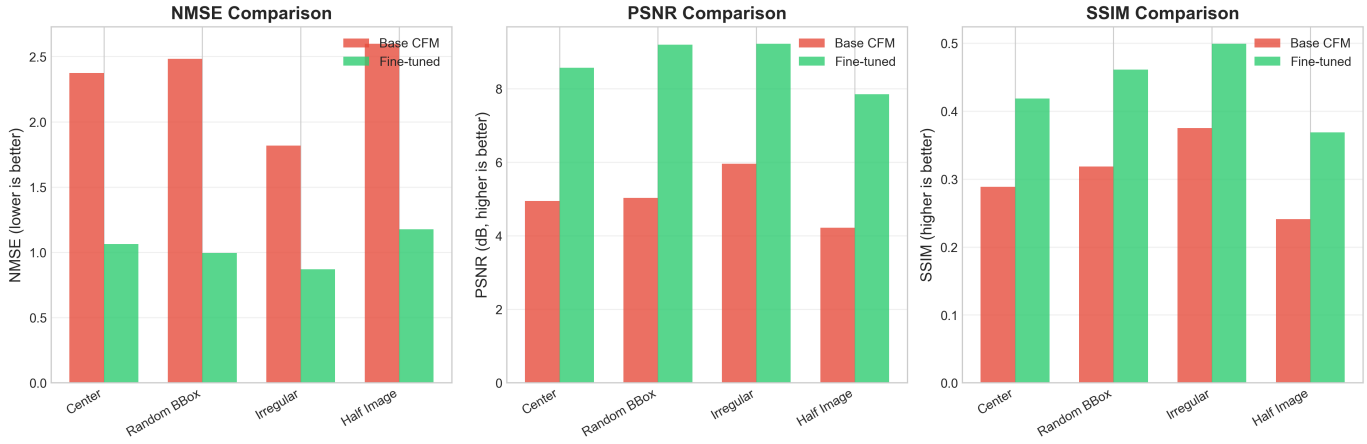


Fig. 6: Inpainting performance comparison: Base CFM (red) vs Fine-tuned (green) across all mask types. Fine-tuning dramatically improves all metrics—reducing NMSE by 52–60% and improving PSNR by 55–86%.

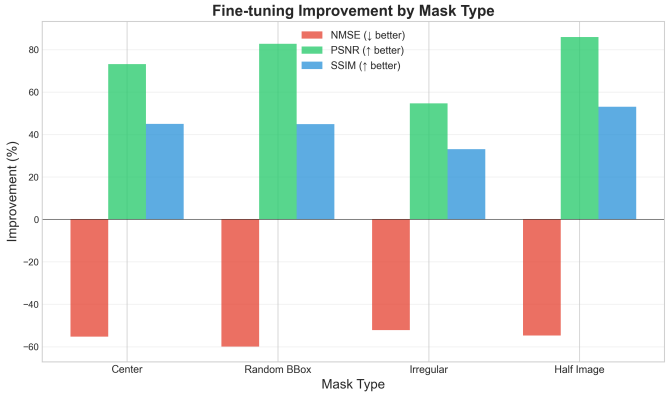
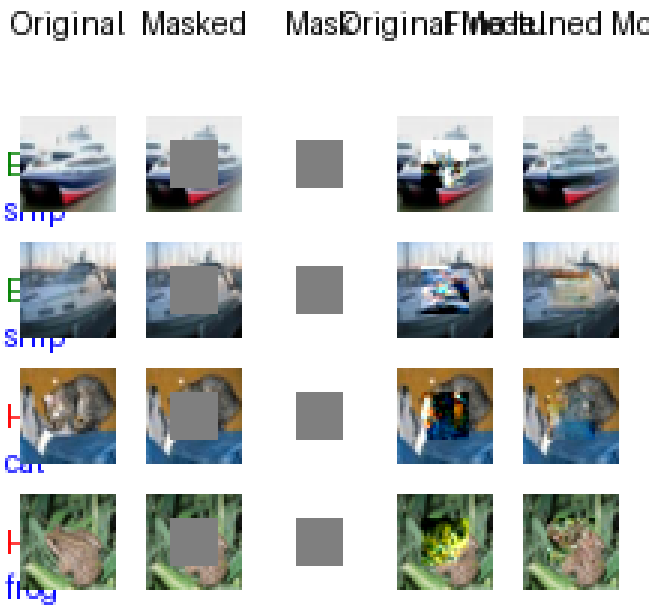
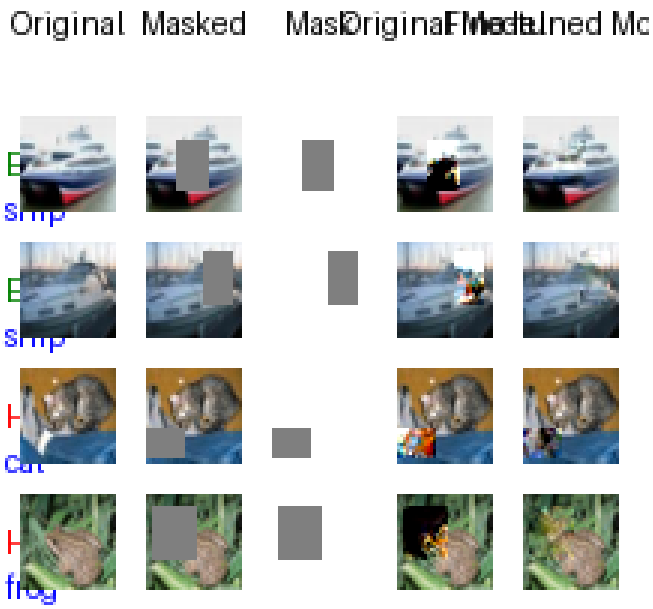


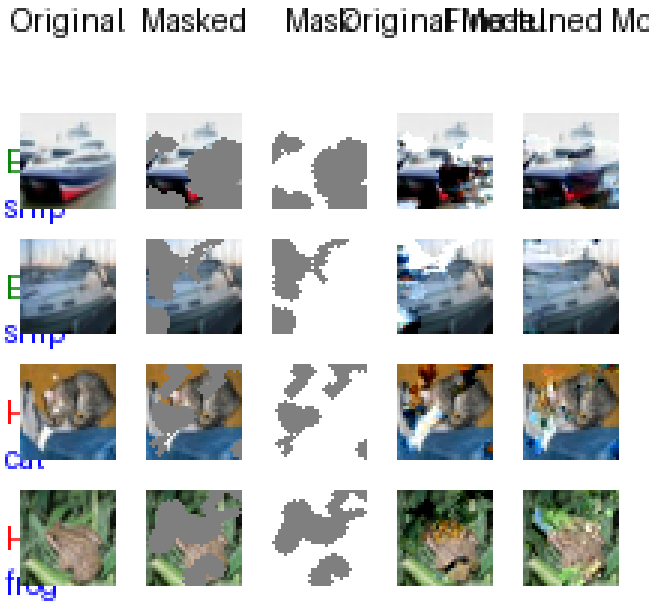
Fig. 7: Percentage improvement from fine-tuning by mask type. Half-image shows the largest PSNR improvement (+86%), while random bbox shows the largest NMSE reduction (−60%).



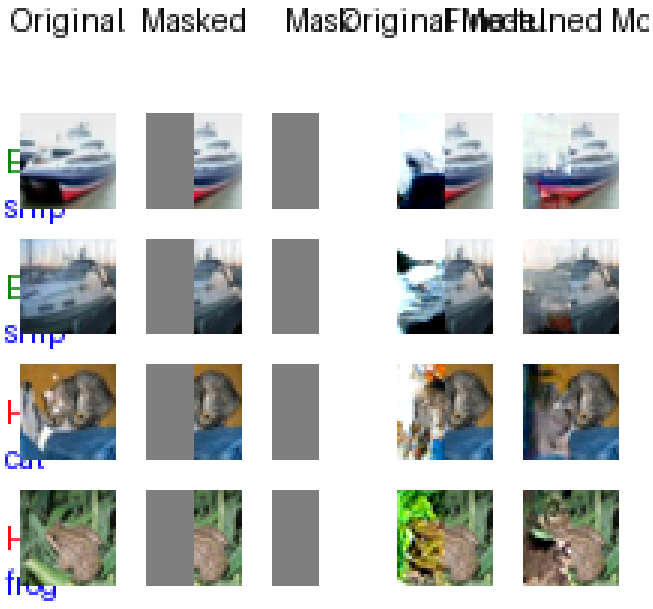
(a) Center Mask



(b) Random Bounding Box



(c) Irregular Brush Strokes



(d) Half Image

Fig. 8: Qualitative inpainting comparison across all four mask types. Each panel shows: Original → Masked → Base Model Result → Fine-tuned Model Result. The fine-tuned model produces more coherent completions with better color consistency and structural integrity. Note the improved ship reconstruction (row 1-2), better cat face completion (row 3), and more natural frog textures (row 4).

Fig. 9: Per-class PSNR heatmap for the fine-tuned model. "Deer" consistently achieves the highest PSNR across all mask types (9.41–10.94 dB), while "truck" and "dog" are most challenging.

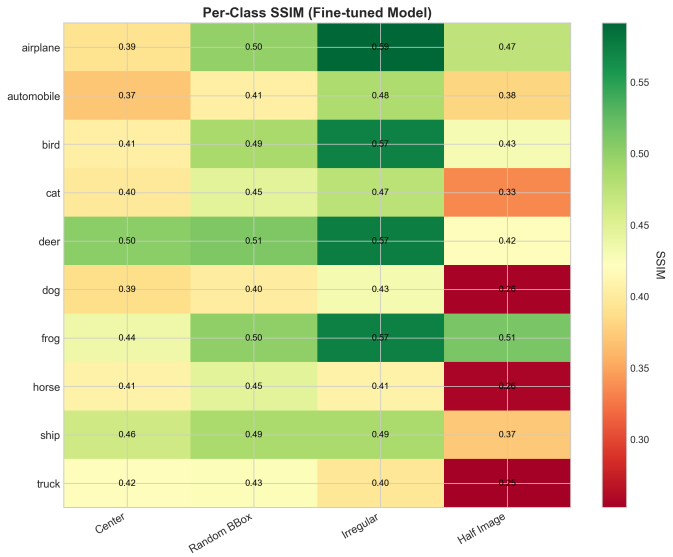


Fig. 10: Per-class SSIM heatmap for the fine-tuned model. Higher values (green) indicate better structural similarity. "Airplane" and "frog" achieve high SSIM on irregular masks.

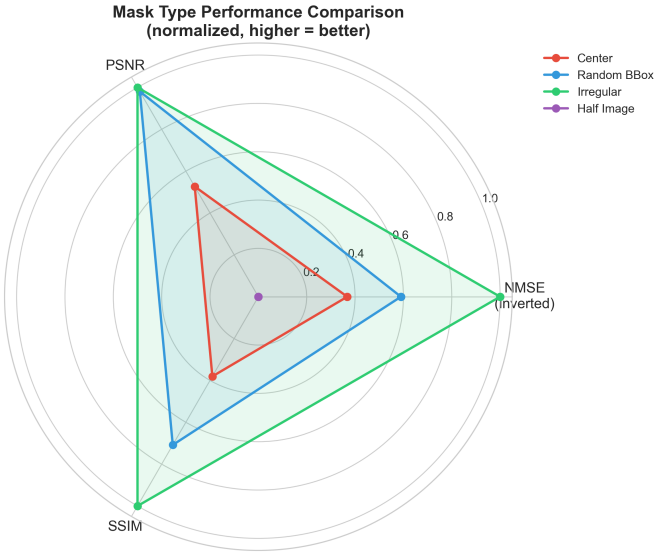


Fig. 11: Normalized performance comparison across mask types. Irregular masks (green) achieve the best overall performance, while half-image (purple) is most challenging. Metrics are normalized so higher values indicate better performance.