DeepCache: Accelerating Diffusion Models for Free

Supplementary Material

A. Pseudo algorithm

We present the pseudocode for our algorithm in Algorithm 1. It illustrates the iterative generation process over N steps, involving one step of complete model inference and N-1 steps of partial model inference. Here, we employ the sampling algorithm from DDPM [19] as an example. Our algorithm is adaptable to other fast sampling methods.

Algorithm 1: DeepCache

Input: A U-Net Model with down-sample blocks $\{D_i\}_{i=1}^{d}$, up-sample blocks$\{U_i\}_{i=1}^{d}$ and middle blocks M
Input: Caching Interval N, Branch Index m
Input: Output from step x_t, timestep t
Output: predicted output at $t-N$ step

▷ 1. Cache Step - Calculate $\epsilon_\theta\left(\mathbf{x}_t, t\right)$ and x_{t-1}
$\mathbf{h}_0 \leftarrow \mathbf{x}_t$ ▷ $\mathbf{h}_i$ for down-sampling features
for $i = 1, \ldots, d$ **do**
  $\mathbf{h}_i \leftarrow D_i(\mathbf{h}_{i-1})$
$\mathbf{u}_{d+1} \leftarrow M(\mathbf{h}_d)$ ▷ $\mathbf{u}_i$ for up-sampling features
for $i = d, \ldots, 1$ **do**
  if $i = m$ **then**
    Store $\mathbf{u}_{i+1}$ in Cache
  $\mathbf{u}_i \leftarrow U_i(\text{Concat}(\mathbf{u}_{i+1}, \mathbf{h}_i))$
$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\mathbf{u}_1\right) + \sigma_t \mathbf{z}$ ▷ $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
▷ 2. Retrieve Step - Calculate $\{x_{t-i}\}_{i=2}^{N}$
for $n = 2, \ldots, N$ **do**
  $\mathbf{h}_0 \leftarrow \mathbf{x}_{t-n+1}$
  for $i = 1, \ldots, m$ **do**
    $\mathbf{h}_i \leftarrow D_i(\mathbf{h}_{i-1})$
  Retrieve $\mathbf{u}_{i+1}$ from Cache
  for $i = m, \ldots, 1$ **do**
    $\mathbf{u}_i \leftarrow U_i(\text{Concat}(\mathbf{u}_{i+1}, \mathbf{h}_i))$
  $\mathbf{x}_{t-n} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\mathbf{u}_1\right) + \sigma_t \mathbf{z}$ ▷ $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
return x_{t-N}

B. Varying Hyper-parameters in Non-uniform 1:N Strategy

In the non-uniform 1:N strategy, the two hyper-parameters involved are the center c and the power p, which is used to determine the sequence of timesteps for conducting the entire model inference. We test on LDM-4-G for the impact of these hyper-parameters. Results are shown in Table 7 and Table 8. From these two tables, a clear trend is evident in the observations: as the parameters p and c are incremented, there is an initial improvement in the generated image quality followed by a subsequent decline. This pattern affirms the effectiveness of the strategy and also aligns with the location of the significant decrease in similarity observed in Figure 2(c).

ImageNet 256 × 256					
Center	**FID ↓**	**sFID ↓**	**IS ↑**	**Precision ↑**	**Recall ↑**
$c = 10$	8.26	8.47	160.3	75.69	48.93
$c = 20$	8.17	8.46	161.18	75.77	48.95
$c = 50$	7.77	8.16	163.74	76.23	49.18
$c = 80$	7.36	7.76	166.21	76.93	49.75
$c = 100$	7.16	7.51	167.52	77.30	49.64
$c = 120$	**7.11**	**7.34**	**167.85**	**77.44**	**50.08**
$c = 150$	7.33	7.36	166.04	77.09	49.98
$c = 200$	8.09	7.79	160.50	75.85	49.69

Table 7. Varying Center c with the power p equals to 1.2. Here the caching interval is set to 20.

ImageNet 256 × 256					
Power	**FID ↓**	**sFID ↓**	**IS ↑**	**Precision ↑**	**Recall ↑**
$p = 1.05$	7.36	7.52	166.12	77.06	50.38
$p = 1.1$	7.25	7.44	166.82	77.17	50.13
$p = 1.2$	7.11	**7.34**	167.85	77.44	50.08
$p = 1.3$	**7.09**	7.35	**167.97**	**77.56**	**50.34**
$p = 1.4$	7.13	7.39	167.68	77.42	50.26
$p = 1.5$	7.25	7.44	166.82	77.17	50.13

Table 8. Varying Power p with the center c equals to 120. Here the caching interval is also set to 20.

	N=2	N=3	N=5	N=10	N=20
Center - c	120	120	110	110	120
Power - p	1.2	1.2	1.4	1.2	1.4

Table 9. Hyper-parameters for the non-uniform 1:N strategy in LDM-4-G

		N=2	N=3	N=4	N=5	N=6	N=7	N=8	
PartiPrompts	Center - c	15	15	15	10	15	15	10
	Power - p	1.5	1.3	1.4	1.5	1.3	1.4	1.4
COCO2017	Center - c	20	20	20	15	15	15	20
	Power - p	1.3	1.4	1.4	1.3	1.5	1.5	1.3

Table 10. Hyper-parameters for the non-uniform 1:N strategy in Stable Diffusion v1.5.

Selected Hyper-parameters for non-uniform 1:N For experiments in LDM, the optimal hyper-parameters and shown in Table 9. For experiments in Stable Diffusion, we chose center timesteps from the set {0, 5, 10, 15, 20, 25} and power values from the set {1.1, 1.2, 1.3, 1.4, 1.5, 1.6}.

ImageNet 256 × 256 (250 DDIM Steps)											
Method	**FID ↓**	**sFID ↓**	**IS ↑**	**Precision ↑**	**Recall ↑**	**Method**	**FID ↓**	**sFID ↓**	**IS ↑**	**Precision ↑**	**Recall ↑**
Baseline - LDM-4*	3.37	5.14	204.56	82.71	53.86	Baseline - LDM-4	3.60	-	247.67	87.0	48.0
Uniform - N=2	3.39	5.11	204.09	82.75	54.07	Non-uniform - N=2	3.46	5.14	204.12	83.21	53.53
Uniform - N=3	3.44	5.11	202.79	82.65	53.81	Non-uniform - N=3	3.49	5.13	203.22	83.18	53.44
Uniform - N=5	3.59	5.16	200.45	82.36	53.31	Non-uniform - N=5	3.63	5.12	200.04	83.07	53.25
Uniform - N=10	4.41	5.57	191.11	81.26	51.53	Non-uniform - N=10	4.27	5.42	193.11	81.75	51.84
Uniform - N=20	8.23	8.08	161.83	75.31	50.57	Non-uniform - N=20	7.36	7.76	166.21	76.93	49.75

Table 11. Comparing non-uniform and uniform 1:N strategy in class-conditional generation for ImageNet using LDM-4-G. *We regenerate the images using the official checkpoint of LDM-4-G.

Dataset	PLMS Steps	**50**	**45**	**40**	**35**	**30**	**25**	**20**	**15**	**10**
COCO2017	PLMS	30.45 1.00×	30.34 1.10×	30.40 1.27×	30.26 1.46×	30.26 1.62×	30.19 1.96×	30.06 2.46×	29.69 3.23×	28.03 4.88×
	w/ DeepCache	30.45 2.01×	30.33 2.23×	30.32 2.51×	30.18 2.88×	30.11 3.36×	29.94 4.02×	29.59 5.03×	28.82 6.69×	26.17 10.06×
PartiPrompts	PLMS	29.61 1.00×	29.61 1.11×	29.53 1.25×	29.44 1.43×	29.44 1.67×	29.38 2.00×	29.31 2.50×	28.95 3.33×	27.25 5.00×
	w/ DeepCache	29.60 2.01×	29.58 2.23×	29.50 2.51×	29.41 2.88×	29.32 3.36×	29.19 4.02×	28.89 5.03×	28.06 6.69×	25.57 10.06×

Table 12. The CLIP Score with DeepCache under different timesteps. Here we use the uniform 1:4 setting. The speedup is based on the latency of 50-step PLMS.

The optimal hyper-parameter values employed in our experiments are detailed in Table 10.

From the selected hyper-parameters, we found out that the optimal values vary slightly across different datasets. A noticeable trend is observed, indicating that the majority of optimal parameters tend to center around the 15th timestep, accompanied by a power value of approximately 1.4.

C. Non-uniform 1:N v.s. Uniform 1:N

We have shown the comparison of the non-uniform 1:N versus uniform 1:N strategy on Stable Diffusion in Figure 6. Here, we extend the comparison to ImageNet with LDM-4-G, and the corresponding results are detailed in Table 11.

In accordance with the observations from Table 11, a consistent pattern emerges compared to the findings on Stable Diffusion. Notably, when employing a substantial caching interval, the non-uniform strategy demonstrates a notable improvement, with the FID increasing from 8.23 to 7.36 with N=20. However, when dealing with a smaller caching interval (N<5), the strategy does not yield an enhancement in image quality. In fact, in certain cases, it may even lead to a slight degradation of images, as evidenced by the FID increasing from 3.39 to 3.46 for N=2.

D. Small Number of Sampling Steps

We demonstrate the effectiveness of DeepCache under different sampling steps in Table 12. To provide a comparison of our results, we also report the results of PLMS under different sampling steps and their corresponding speedups. From the table, we can see that DeepCache exhibits a significant improvement over the baseline at nearly equivalent speedups. For instance, at close to a 5-fold speedup, PLMS's Clip Score is 28.03 and 27.25 (corresponding to COCO2017 and PartiPrompts, respectively), while DeepCache achieves 29.59 and 28.89.

CIFAR-10 32 × 32				
Skip Branch	**MACs ↓**	**Throughput ↑**	**Speed ↑**	**FID ↓**
1	1.60G	29.60	3.023×	7.14
2	2.24G	22.24	2.272×	5.94
3	3.01G	18.11	1.850×	5.73
4	3.89G	15.44	1.577×	5.69
5	4.58G	13.15	1.343×	5.51
6	5.31G	11.46	1.171×	4.93
7	5.45G	11.27	1.151×	4.92
8	5.60G	11.07	1.131×	4.76
9	5.88G	10.82	1.105×	4.54
10	5.95G	10.73	1.096×	4.57
11	5.99G	10.67	1.089×	4.52
12	6.03G	10.59	1.082×	4.48

Table 13. Effect of different skip branches. Here we test the impact under the uniform 1:5 strategy.

E. Varying Skip Branches

In Table 13, we show the impact on image quality as we vary the skip branch for executing DeepCache. For our experiments, we employ the uniform 1:N strategy with N=5, and the sampling of DDIM still takes 100 steps. From the results in the table, we observe that the choice of skip branch introduces a trade-off between speed and image fidelity. Specifically, opting for the first skip branch with no down-sampling blocks and one up-sampling block yields approximately 3× acceleration, accompanied by a reduction in FID to 7.14. Additionally, certain skip branches exhibit significant performance variations, particularly the 6-th branch. The results emphasize an extra trade-off between speed and image quality, complementing the earlier noted trade-off linked to different sampling steps. This particular trade-off operates at the level of model size granularity and can be achieved without incurring additional costs.

PLMS				DeepCache				
Steps	Throughput	Speed	CLIP Score	N	Throughput	Speed	Uniform 1:N	Non-Uniform 1:N
50	0.230	1.00	29.51	1	-	-	-	-
45	0.251	1.09	29.40	2	0.333	1.45	29.54	29.51
40	0.307	1.34	29.35	3	0.396	1.72	29.50	29.59
35	0.333	1.45	29.24	4	0.462	2.01	29.53	29.57
30	0.384	1.67	29.24	5	0.494	2.15	29.41	29.50
25	0.470	2.04	29.32	6	0.529	2.30	29.30	29.46
20	0.538	2.34	29.15	7	0.555	2.41	29.11	29.42
15	0.664	2.89	28.58	8	0.582	2.53	28.97	29.26

Table 14. Stable Diffusion v1.5 on PartiPrompt

PLMS				DeepCache				
Steps	Throughput	Speed	CLIP Score	N	Throughput	Speed	Uniform 1:N	Non-Uniform 1:N
50	0.237	1.00	30.24	1	-	-	-	-
45	0.252	1.06	30.14	2	0.356	1.50	30.31	30.37
40	0.306	1.29	30.19	3	0.397	1.68	30.33	30.34
35	0.352	1.49	30.09	4	0.448	1.89	30.28	30.31
30	0.384	1.62	30.04	5	0.500	2.11	30.19	30.23
25	0.453	1.91	29.99	6	0.524	2.21	30.04	30.18
20	0.526	2.22	29.82	7	0.555	2.34	29.90	30.10
15	0.614	2.59	29.39	8	0.583	2.46	29.76	30.02

Table 15. Stable Diffusion v1.5 on MS-COCO 2017

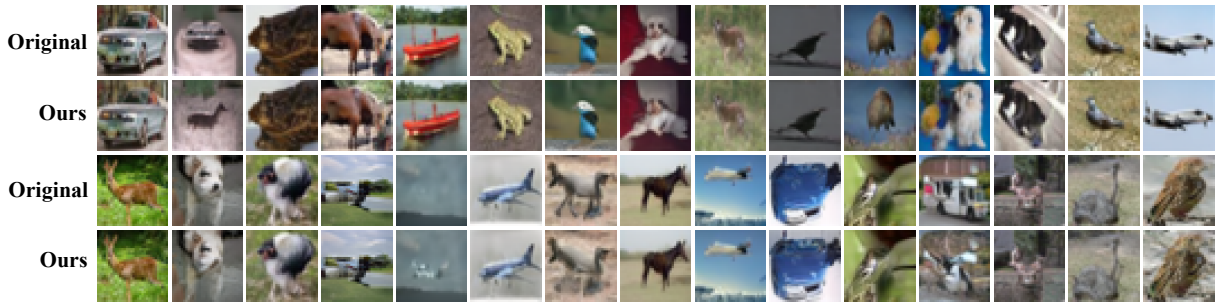


Figure 9. DDPM for LSUN-Churches: Samples with DDIM-100 steps (upper line) and DDIM-100 steps + DeepCache with N=5 (lower line). The speedup Ratio here is 1.85×.

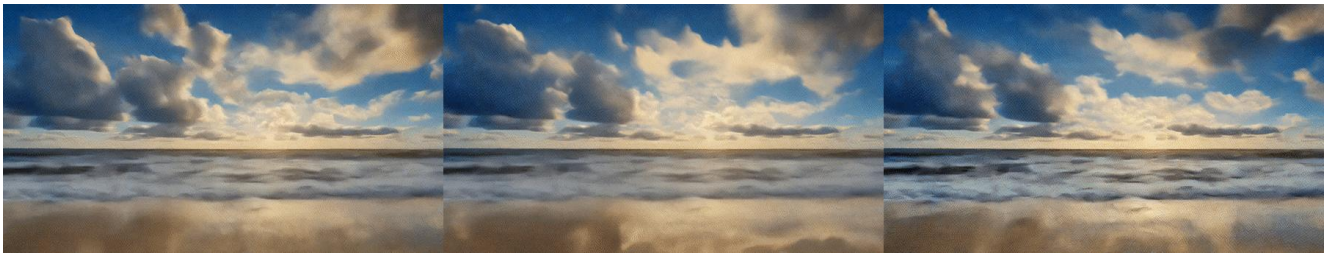
F. Prompts

Prompts in Figure 1(a):

- A bustling city street under the shine of a full moon
- A picture of a snowy mountain peak, illuminated by the first light of dawn
- dark room with volumetric light god rays shining through window onto stone fireplace in front of cloth couch
- A photo of an astronaut on a moon
- A digital illustration of a medieval town, 4k, detailed, trending in artstation, fantasy
- A photo of a cat. Focus light and create sharp, defined edges

Prompts in Figure 5:

- A person in a helmet is riding a skateboard
- There are three vases made of clay on a table
- A very thick pizza is on a plate with one piece taken.
- A bicycle is standing next to a bed in a room.
- A kitten that is sitting down by a door.
- A serene mountain landscape with a flowing river, lush greenery, and a backdrop of snow-capped peaks, in the style of an oil painting.
- A delicate floral arrangement with soft, pastel colors and light, flowing brushstrokes typical of watercolor paintings.
- A magical winter wonderland at night. Envision a landscape covered in fresh snow, with twinkling stars above, a cozy cabin with smoke rising from its chimney, and a



(a) Generated video of SVD-XT without acceleration.

(b) 1.7x acceleration of SVD-XT by DeepCache

(c) Generated video of SVD-XT without acceleration.

(d) 1.7x acceleration of SVD-XT by DeepCache

Figure 10. Acceleration of Stable Video Diffusion with DeepCache. We show 3 frames of the video, which is evenly spaced out.

gentle glow from lanterns hung on the trees

- A photograph of an abandoned house at the edge of a forest, with lights mysteriously glowing from the windows, set against a backdrop of a stormy sky. high quality photography, Canon EOS R3.
- A man holding a surfboard walking on a beach next to the ocean.

G. Detailed Results for Stable Diffusion

We furnish the elaborate results corresponding to Figure 6 in Table 14 and Table 15. Given the absence of a definitive N for aligning the throughput of PLMS, we opt for an alternative approach by exploring results for various N values. Additionally, we assess the performance of the PLMS algorithm across different steps. Analyzing the data from these tables reveals that for $N < 5$, there is minimal variation in the content of the image, accompanied by only slight fluctuations in the CLIP Score.

H. Results on Video Diffusion

We provide the generated video for Stable Video Diffusion [5] accelerated by DeepCache in Figure 10.

I. More Samples for Each Dataset

We provide the generated images for each model and each dataset in Figure 9, Figure 11, Figure 12, Figure 13 and Figure 14.

Figure 11. Stable Diffusion v1.5: Samples with 50 PLMS steps (upper line) and 50 PLMS steps + DeepCache with N=5 (lower line). The speedup Ratio here is 2.15×. Here we select prompts from the MS-COCO 2017 validation set.

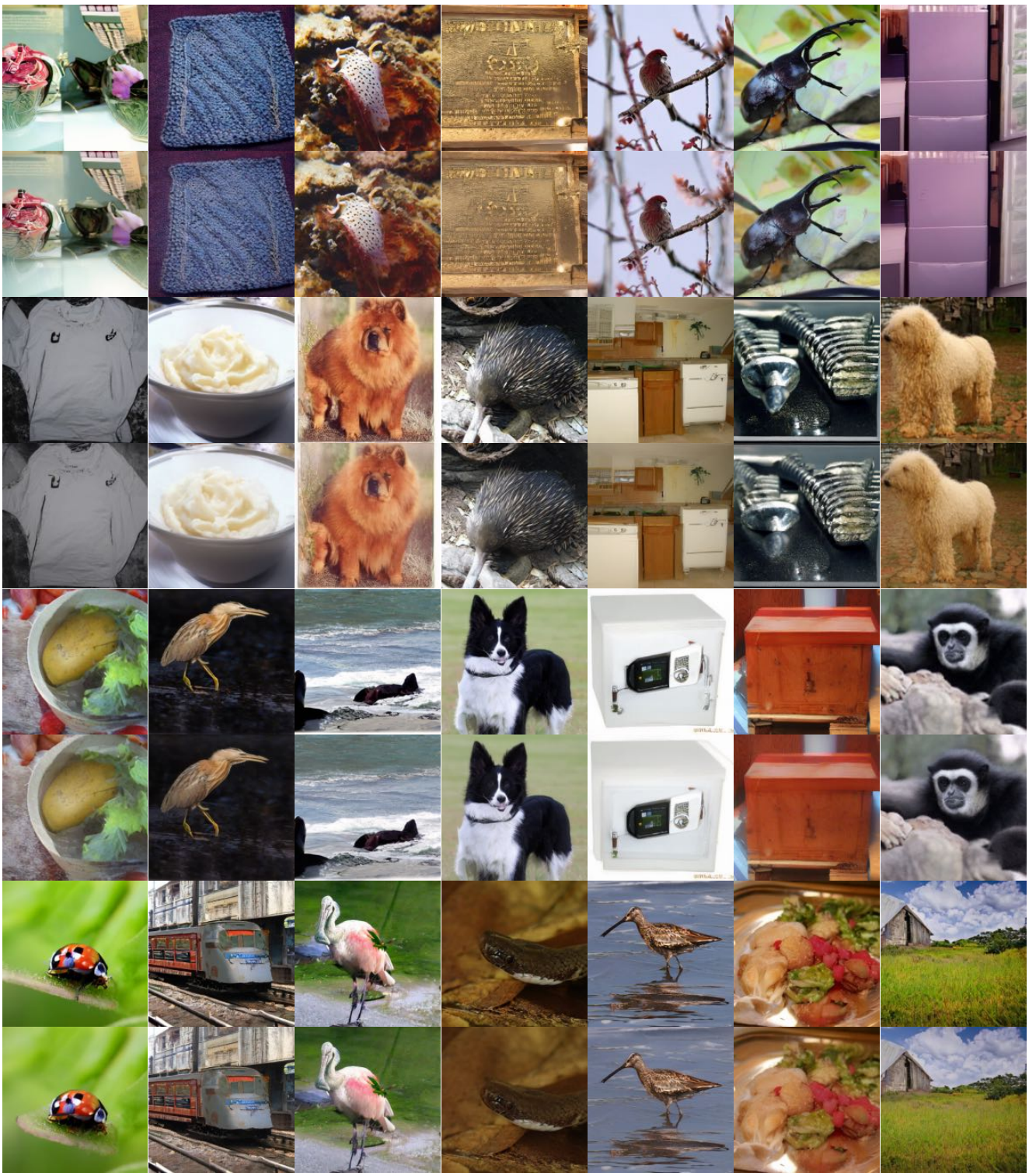


Figure 12. LDM-4-G for ImageNet: Samples with DDIM-250 steps (upper line) and DDIM-250 steps + DeepCache with N=10 (lower line). The speedup Ratio here is 6.96×.

Figure 13. DDPM for LSUN-Bedroom: Samples with DDIM-100 steps (upper line) and DDIM-100 steps + DeepCache with N=5 (lower line). The speedup Ratio here is 1.48×.

Figure 14. DDPM for LSUN-Churches: Samples with DDIM-100 steps (upper line) and DDIM-100 steps + DeepCache with N=5 (lower line). The speedup Ratio here is 1.48×.