Scaling Down the Image Pyramid: Markovian Dependencies for Efficient Next-Scale Prediction

Sinoué GAD
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

Edgar DUC
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

***Abstract*—"Scale-based" image generation models like VAR [1] offer superior structural coherence but suffer from a "token explosion" problem: the cumulative sequence length across all scales leads to quadratic memory overhead. This project explores a Markovian Pyramidal approach to decouple this dependency. By assuming each resolution scale only requires information from its immediate predecessor and from the target class, we significantly reduce the computational cost of adding refinement scales.**

I. Introduction

Multi-resolution autoregressive models such as VAR [1] and FlexVAR [2] achieve strong structural coherence by conditioning each scale on the full token history of all previous ones. While effective, this design entails a quadratic memory bottleneck: the cumulative sequence length grows faster than the target resolution, making high-fidelity synthesis increasingly costly. We question whether this full historical dependency is truly necessary. We argue that image generation can be cast as a Markovian process, each scale depending only on its immediate predecessor and a persistent class/target token without sacrificing global coherence. This assumption alone, however, shifts the burden onto the latent representation: without history, the model must encode both global structure and local texture within a single forward pass. We address this through Matryoshka capacity partitioning [3], dedicating nested embedding subspaces to increasing levels of visual frequency. Together, the Markovian constraint and the structured latent space form a principled trade-off between computational efficiency and generation quality, which this work evaluates on CIFAR-10.

II. Problem Formulation

A. Standard Visual Autoregressive Modeling

Let an image be represented by a multi-resolution pyramid of latent discrete token maps, denoted as $S = (S_1, S_2, \ldots, S_K)$, where K is the total number of scales. In standard Visual Autoregressive (VAR) models, the probability of generating the entire image is factorized as a dense sequence of conditional probabilities:

$$P(S|c) = \prod_{k=1}^{K} P(S_k|S_1, S_2, \ldots, S_{k-1}, c) \tag{1}$$

where c is the global condition, such as the target class token.

Let N_k be the number of tokens at scale k. To predict the tokens of scale k, the model must compute self-attention over a cumulative sequence length $L_k = 1 + \sum_{i=1}^{k} N_i$. Because self-attention has a quadratic memory and time complexity of $\mathcal{O}(L_k^2)$, this dense historical dependency creates a severe computational bottleneck as the number of scales increases. We refer to this scaling limitation as the pyramidal overhead.

B. Markovian Next-Scale Prediction

To mitigate this overhead, we propose formulating the generation as a Markovian process. We hypothesize that the structural and textural information required to generate scale S_k is sufficiently contained within the immediate preceding scale S_{k-1}, provided the global semantic context c is persistently maintained. The dense factorization is therefore approximated by:

$$P(S|c) \approx P(S_1|c) \prod_{k=2}^{K} P(S_k|S_{k-1}, c) \tag{2}$$

Under this Markovian framework, the context window at step k is strictly limited to the tokens of k tokens, and the class token c. Adding the class token at each scale ensure that the prediction doesn't drift through the process. The effective sequence length is drastically reduced to $L'_k = 1 + N_k$. This breaks the cumulative sequence growth, bounding the attention complexity and enabling the addition of refinement scales without prohibitive memory costs.

C. The Information Bottleneck Challenge

While computationally efficient, the Markovian assumption introduces a critical representation challenge. Restricting the context to S_{k-1} creates an information bottleneck. If an intermediate scale fails to retain the global structural intent established in earlier scales, the generation will suffer from semantic drift. Consequently, replacing dense historical dependencies with a short-term memory necessitates a highly structured latent space. The network must guarantee that low-frequency global patterns and high-frequency local textures are simultaneously preserved and efficiently transmitted through the Markovian chain.

III. Implementation Details

A. Inference and Scale Transition

The generation process follows a strict coarse-to-fine trajectory. To maintain the Markovian dependency during inference, the transition between scale S_{k-1} and S_k is handled through a spatial refinement pipeline:

1) **Latent Retrieval:** The discrete indices predicted at scale $k-1$ are first mapped back to their continuous latent representations using a VQ-VAE codebook.
2) **Spatial Upscaling:** These latent features are upscaled via bicubic interpolation to match the spatial resolution of the target scale S_k. This step provides the Transformer with a structural "template" of the image.
3) **Quantized Input Projection:** The upscaled features are then projected into the model's embedding space through a linear layer.
4) **Conditional Generation:** Finally, the Transformer uses this upscaled prior, combined with the global class token to predict the categorical distribution of the tokens at scale S_k.

This pipeline ensures that each scale is not generated in isolation but as a direct high-frequency refinement of its predecessor's upscaled geometry.

B. Markovian Block Masking and Parallelization

To train the model efficiently, we introduce a block-diagonal attention mask as illustrated in Fig. 1. Unlike the cumulative causal mask in standard VAR, our mask restricts each scale S_k to attend only to the tokens of the previous scale S_{k-1} and its own generated tokens.

A critical component of this architecture is the **global class token (cls)**, which is exempted from the Markovian constraint. By allowing every scale to maintain full attention to the class token, we provide a persistent semantic anchor that ensures the subject of the image remains consistent throughout the refinement process. This masking strategy allows for the parallel training of all scales in a single forward pass, significantly reducing training time while strictly enforcing the Markovian dependency during inference.

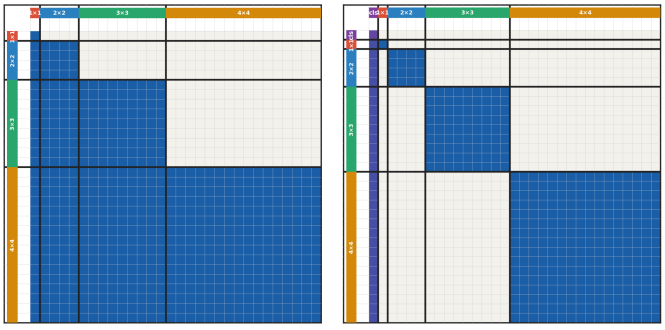


Fig. 1: Comparison between the standard VAR causal block mask (left) and our proposed diagonal block mask (right) for Markovian training with the cls token in purple.

C. Scale-wise Loss Optimization

Since the scales are architecturally decoupled, applying a standard cross-entropy loss over the entire flattened sequence is no longer optimal. Instead, we compute an independent cross-entropy loss for each scale k:

$$\mathcal{L}_{total} = \sum_{k=1}^{K} \lambda_k \cdot \mathrm{CE}(S_k, \hat{S}_k) \tag{3}$$

where λ_k is a scale-specific weight. This formulation allows us to monitor the convergence of each resolution level independently, which is vital given that different scales operate in different frequency domains.

D. Mitigating Scale Conflicts with Matryoshka Partitioning

A major challenge in multi-scale modeling that we identified is the gradient conflict: coarse scales prioritize global structure (low frequencies), while fine scales focus on local textures (high frequencies). When sharing a single embedding space, we observed that these objectives interfere significantly, as they lack the regular causal dependencies found in standard VAR. In our experiments, this led to either an 'aquarelle effect', where textures remain blurry, or a 'noisy effect', where high-frequency gradients dominate the generation at the expense of global structural coherence (see Appendix B).
To resolve this, we implement **Matryoshka Capacity Partitioning** (see Fig. 2). We partition the embedding dimension C into nested subspaces. For instance, the coarsest scale S_1 is restricted to the first C/K dimensions, while the finest scale S_K utilizes the full capacity. This nesting ensures that:

1) Coarse structural information is "protected" in the lower dimensions.
2) High-frequency details have dedicated "bandwidth" in the higher dimensions, preventing them from being overwritten by the stronger gradients of the structural scales.

This hierarchical organization of the latent space is the necessary counterpart to the Markovian assumption, providing the model with a structured "short-term memory."

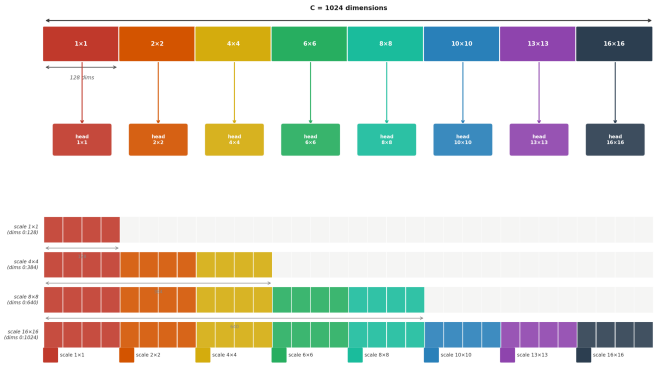


Fig. 2: Matryoshka Representation in our model: Each scale i activates the first (i+1)×128 dimensions of the 1024-dimensional embedding, with a dedicated prediction head operating on the active subspace only.

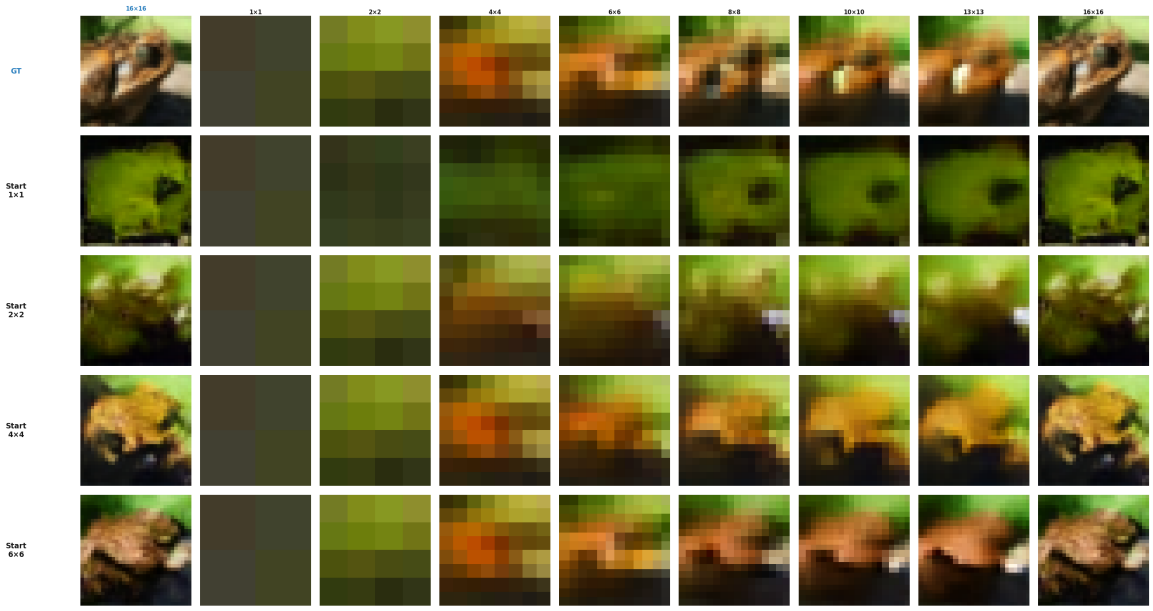


Fig. 3: Partial scale seeding for conditional generation. The first row shows the ground truth decomposed across all scales. Each subsequent row is generated autoregressively from the ground truth tokens up to scale k, with finer scales freely sampled by the model. See more examples in Appendix. A.

IV. Experimental Setup

A. Experimental Setup

All experiments are conducted on the CIFAR-10 dataset. We evaluate our framework by measuring the structural consistency and the quality of the generated samples.

B. VQ-VAE Configuration

The visual tokenizer is a Vector Quantized Variational Autoencoder [4] optimized for the CIFAR-10 resolution. We employ a reduction factor of 2 with a patch size of 2, resulting in a latent grid of 16x16 tokens. The codebook size is set to 1024 entries.

After training for 20 epochs on the CIFAR-10 training set, the VQ-VAE achieves a r-FID-10K of 4.2.

C. Transformer Architecture

The generative backbone is a custom Transformer model derived from the FlexVAR architecture, adapted to enforce the Markovian next-scale prediction constraint. The model consists of 12 layers with an embedding dimension (C) of 512, totaling approximately 50 million parameters.

To handle conditional information efficiently, we utilize Adaptive Layer Normalization (AdaLN) layers applied to the condition embedding (*cond_emb*). This allows the global class target to modulate the hidden states across all layers of the network. During training, we use the block-diagonal masking strategy described in Section III to parallelize the learning of the 8 resolution scales while strictly maintaining the Markovian dependencies between consecutive scales.

D. Training Details

The Transformer is trained using the scale-wise cross-entropy loss with Matryoshka capacity partitioning. We use the AdamW optimizer with a learning rate of $1e-4$ and a weight decay of 0.05. Due to limited compute, the model has been trained for 64 epochs (roughly 50K steps), which limits the quality of the generated images.

V. Results

Our model achieves an FID-10K of **62.0** on the CIFAR-10 training set. Even though this result does not rival with SOTA models, it is a very promising start taking into account our limited computational budget.

In table I, we detail the accuracy of the next-scale prediction in the teacher forcing setting, i.e. on average at what frequency do we have $\arg\max f_\theta(S_k^{gt}) = S_{k+1}^{gt}$. Please note that the bigger S_{k+1} is, the more difficult it is to get a perfect match.
We observe that, counter-intuitively, some bigger scales have been much better memorized than some lower scales (namely scale $\text{acc}(S_{k+1} = 4 \times 4) > \text{acc}(S_{k+1} = 2 \times 2)$ and $\text{acc}(S_{k+1} = 13 \times 13) > \text{acc}(S_{k+1} = 10 \times 10) > \text{acc}(S_{k+1} = 8 \times 8)$. This is what we denote as "conflict between scales" during training : during optimization, the model has focused on fitting some scales more than others. This interpretation is justified by the figure 10b. The scale $S_{k+1} = 4 \times 4$ is indeed the one for which the loss is most dramatically decreasing.

In figure 3, we report the generation quality of the model when the auto-regressive sampling starts from a given ground-truth scale.

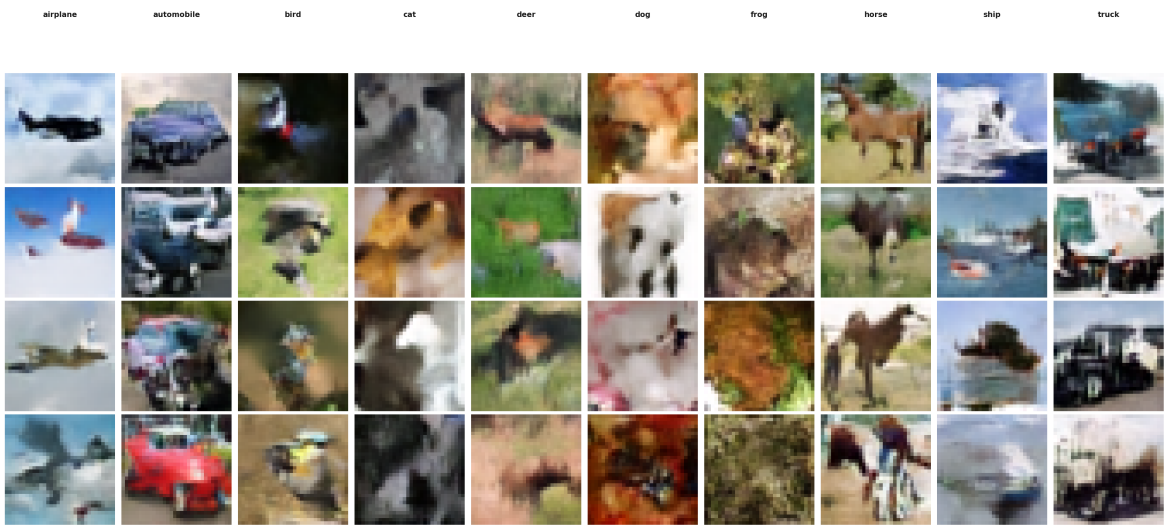


Fig. 4: Free generation sampling on different classes.

TABLE I: Next-Scale Prediction Performance across Resolutions

Resolution	Accuracy (%)	Unique Predicted Tokens
1×1	0.0%	5 / 1024
2×2	12.5%	19 / 1024
4×4	92.2%	102 / 1024
6×6	43.8%	160 / 1024
8×8	17.0%	229 / 1024
10×10	41.1%	312 / 1024
13×13	56.6%	414 / 1024
16×16	14.0%	582 / 1024

We can qualitatiely see that the bigger are the ground-truth scales fed to the model, the more the final reconstruction resembles the original image from the dataset. Also, the bigger the starting ground-truth scale, the more structured and plausible is the image.

In figure 4 (class-conditional free generation), we see that the generated samples show a great variety of colors, sharp shapes, and neither blur nor noise. Matryushka representation has therefore successfully solved the *"aquarelle vs. noisy effect"*.

Some patterns in the per-scale loss and gradient norm plots (fig. 10b, 11b) are very intriguing, namely the stair-shaped decrease. We do not yet have valuable explanation for this phenomenon.

The gradually increasing global gradient norm (fig. 11a) is also unexpected and entails funnel-shaped global loss landscape.

VI. Limitations

Despite the promising results of our Markovian pyramidal formulation, several limitations remain.

Limited model capacity and training budget. Due to computational constraints, our experiments were conducted with a relatively small Transformer (approximately 50M parameters) trained for only 64 epochs. This limited capacity and training duration are insufficient to reach the performance levels of larger-scale autoregressive models, and likely explain the relatively weak quantitative scores observed at finer resolutions.

Patch size choice. 2×2 patch size for CIFAR-10 tokenization is perhaps too small and induces pixel-level prediction. We should try bigger patches.

Sensitivity to scale-wise loss weighting. The proposed scale-wise objective introduces a set of hyperparameters $\{\lambda_k\}$ that require careful tuning. We observed that different weighting schemes can substantially affect both convergence and final sample quality. We were unable to perform a systematic hyperparameter sweep. As a result, the reported results likely underestimate the achievable performance of the method under optimal weighting.

Dependence on VQ-VAE tokenization quality. The performance of the model is tightly coupled to the quality of the VQ-VAE tokenizer. Any reconstruction error or information loss in the discrete latent space directly impacts the subsequent autoregressive modeling. In particular, the relatively small codebook and limited tokenizer training may restrict the fidelity of high-frequency details.

Evaluation limited to CIFAR-10. All experiments are conducted on CIFAR-10, a low-resolution dataset with limited diversity. It remains unclear whether the proposed Markovian formulation scales effectively to higher resolutions or more complex datasets such as ImageNet, where long-range dependencies may play a more critical role.

VII. Future works

Distillation An important advantage of Matryushka representation is that it turns our model into an optimal teacher for scale-specific distillation. Because each scale has its own embedding sub-space, you can train a smaller model from a teacher with distillation by only looking at the right heads and embedding sub-spaces. You can basically extract the

sub-codebook corresponding to sub-embeddings, extract the heads corresponding to sub-embeddings, and re-train smaller MLPs/LayerNorms.

Without Matryushka ? We do not know yet if the model without Matryushka representation is viable, i.e. if it is possible to fix the loss/gradient explosion problem so that, with extended training, all scales losses are correctly minimized and we therefore get rid of the scale conflict problem.

VIII. Conclusion

In this work, we introduced a Markovian pyramidal formulation for multi-scale image generation, aiming to alleviate the quadratic memory overhead inherent to standard visual autoregressive models. By constraining each scale to depend only on its immediate predecessor and a persistent global class token, we successfully decouple the cumulative sequence dependency and enable more efficient scaling to higher resolutions.

To address the information bottleneck induced by this Markovian assumption, we proposed a structured latent representation based on Matryoshka capacity partitioning. This design enforces a frequency-aware organization of the embedding space, allowing coarse structural information and fine-grained textures to coexist without destructive interference.

Experimental results on CIFAR-10 demonstrate that, despite limited computational resources, the proposed approach is capable of generating structurally coherent samples while significantly reducing computational complexity. Qualitative analyses confirm that the model preserves global semantics and progressively refines details across scales, validating the effectiveness of the Markovian next-scale prediction paradigm.

However, the current performance remains below state-of-the-art methods, highlighting several important challenges. In particular, the dynamics of loss and gradient, and the sensitivity to loss weighting open several directions for future work.

References

[1] K. Tian, Y. Jiang, Z. Yuan, B. Peng, and L. Wang, "Visual autoregressive modeling: Scalable image generation via next-scale prediction," *Advances in neural information processing systems*, vol. 37, pp. 84 839–84 865, 2024.

[2] S. Jiao, G. Zhang, Y. Qian, J. Huang, Y. Zhao, H. Shi, L. Ma, Y. Wei, and Z. Jie, "Flexvar: Flexible visual autoregressive modeling without residual prediction," *arXiv preprint arXiv:2502.20313*, 2025.

[3] A. Kusupati, G. Bhatt, A. Rege, M. Wallingford, A. Sinha, V. Ramanujan, W. Howard-Snyder, K. Chen, S. Kakade, P. Jain *et al.*, "Matryoshka representation learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 30 233–30 249, 2022.

[4] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," *Advances in neural information processing systems*, vol. 30, 2017.

APPENDIX A
PARTIAL SCALE SEEDING — ADDITIONAL EXAMPLES

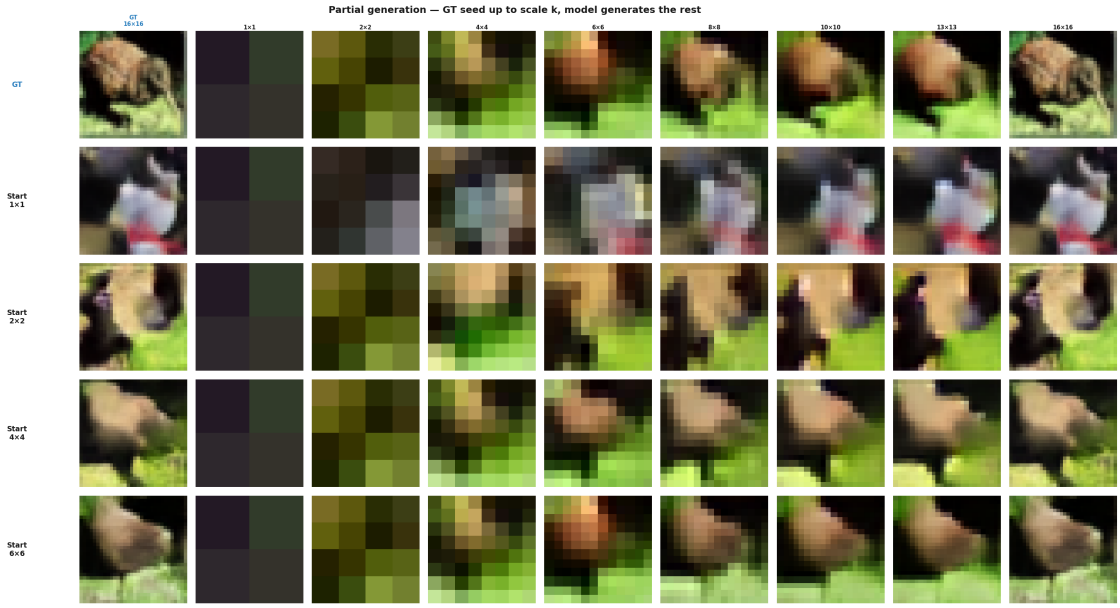


Fig. 5: Partial scale seeding, sample 2.



Fig. 6: Partial scale seeding, sample 3.

APPENDIX B
AQUARELLE VS NOISY EFFECT

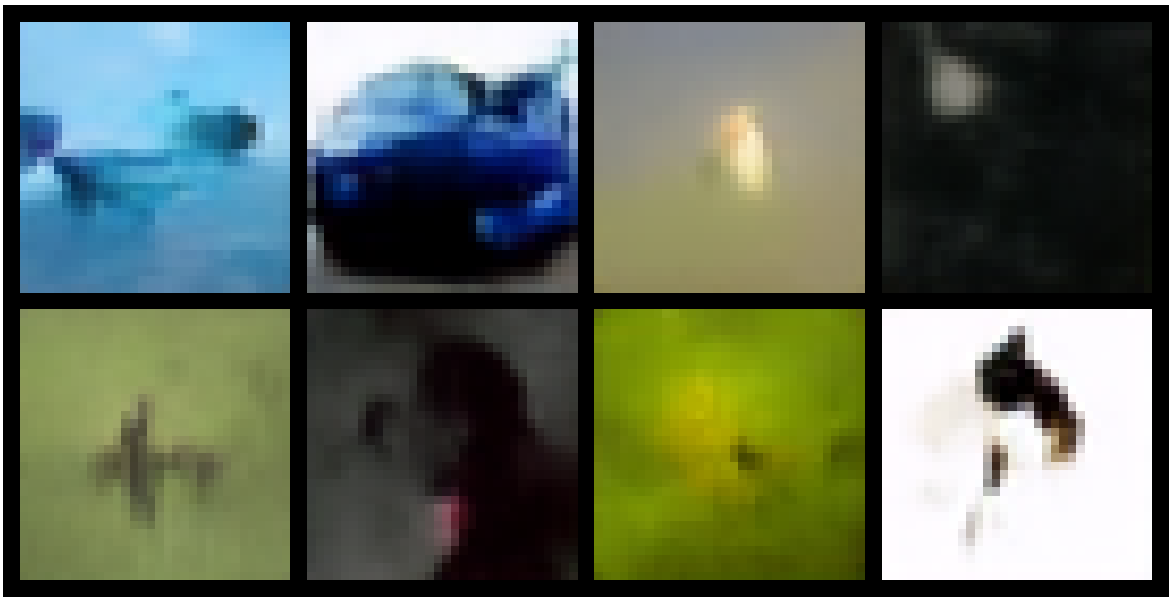


Fig. 7: Aquarelle effect

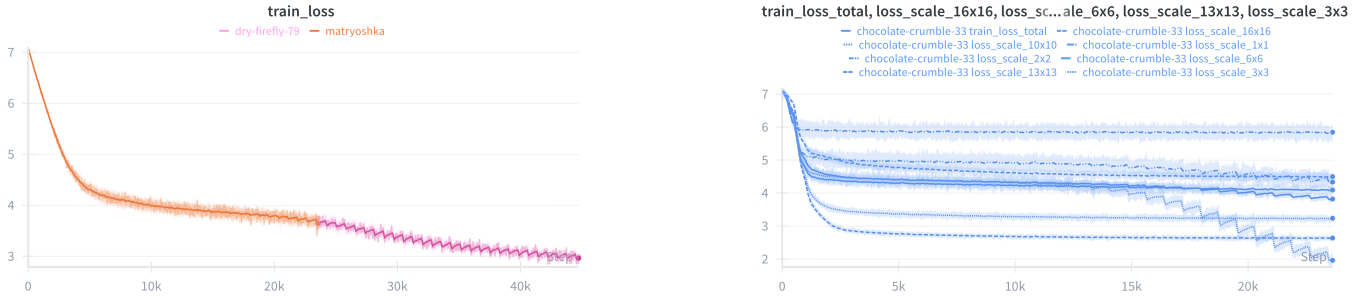


Fig. 8: Noisy effect

APPENDIX C
TRAINING DETAILS



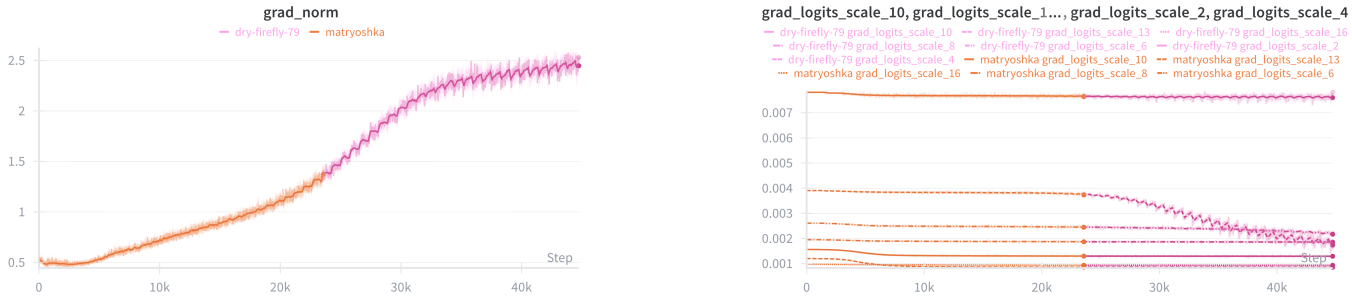
Fig. 9: Global training loss without using Matryushka representation



(a) Global training loss

(b) Per-scale training loss

Fig. 10: Training loss using Matryushka representation



(a) Global gradient norm

(b) Per-scale gradient norm

Fig. 11: Training gradient norm using Matryushka representation