Kandinsky 3: Text-to-Image Synthesis for Multifunctional Generative Framework

Vladimir Arkhipkin[1], **Viacheslav Vasilev**[1, 2], **Andrei Filatov**[1, 3], **Igor Pavlov**[1,*],
Julia Agafonova[1], **Nikolai Gerasimenko**[1], **Anna Averchenkova**[1], **Evelina Mironova**[1],
Anton Bukashkin[1, 4], **Konstantin Kulikov**[1, 5], **Andrey Kuznetsov**[1, 6], **Denis Dimitrov**[1, 6]
[1]Sber AI, [2]MIPT, [3]Skoltech, [4]HSE University, [5]NUST MISIS, [6]AIRI
{dimitrov}@airi.net

Abstract

Text-to-image (T2I) diffusion models are popular for introducing image manipulation methods, such as editing, image fusion, inpainting, etc. At the same time, image-to-video (I2V) and text-to-video (T2V) models are also built on top of T2I models. We present Kandinsky 3, a novel T2I model based on latent diffusion, achieving a high level of quality and photorealism. The key feature of the new architecture is the simplicity and efficiency of its adaptation for many types of generation tasks. We extend the base T2I model for various applications and create a multifunctional generation system that includes text-guided inpainting/outpainting, image fusion, text-image fusion, image variations generation, I2V and T2V generation. We also present a distilled version of the T2I model, evaluating inference in 4 steps of the reverse process without reducing image quality and 3 times faster than the base model. We deployed a user-friendly demo system in which all the features can be tested in the public domain. Additionally, we released the source code and checkpoints for the Kandinsky 3 and extended models. Human evaluations show that Kandinsky 3 demonstrates one of the highest quality scores among open source generation systems.

1 Introduction

Text-to-image (T2I) models play a dominant role in generative computer vision technologies, providing high quality results and language understanding along with near real-time inference speed. This led to their popularity and accessibility for many applications through graphic AI editors and web-platforms, including chatbots. At the same time, T2I models are also used outside the image domain, e.g. as a backbone for text-to-video (T2V) generation models. Similar to trends in natural language processing (NLP) (et al, 2024), in generative computer vision there is increasing interest in systems that solve many types of generation tasks. The growing computational complexity of such methods is raising interest in distillation and inference speed up approaches.

Contributions of this work are as follows:

- We present Kandinsky 3, a new T2I generation model and its distilled version, accelerated by 3 times. We also propose an approach using the distilled version as a refiner for the base model. Human evaluation results demonstrate the quality of refined model is comparable to the state-of-the-art (SotA) solutions.

- We create one of the first feature-rich generative frameworks with open source code and public checkpoints[1][2]. We also extend Kandinsky 3 model with a number of generation options, such as inpainting/outpainting, editing, and image-to-video and text-to-video.

- We deploy a user-friendly web editor that provides free access to both the main T2I model and all the extensions mentioned[3]. The video demonstration is available on YouTube[4].

2 Related Works

To date, diffusion models (Ho et al., 2020) are de facto standard in the text-to-image generation task (Saharia et al., 2022; Balaji et al., 2022; Arkhipkin et al., 2024). Some models, such as Stable Diffusion (Rombach et al., 2022; Podell et al., 2023), are publicly available and widespread in the research community (Deforum, 2022). From the user's point of view, the most popular models are those that

[*] Work done during employment at Sber AI.

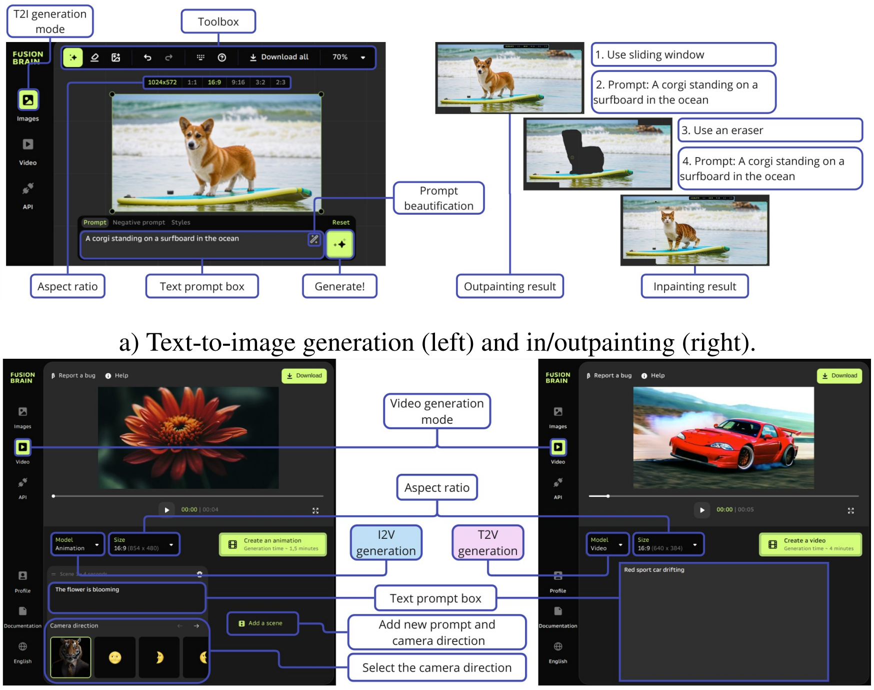
[1] https://github.com/ai-forever/Kandinsky-3

[2] ,https://huggingface.co/kandinsky-community/kandinsky-3

[3] https://fusionbrain.ai/en/editor

[4] https://youtu.be/I-7fhQNy4yI

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing System Demonstrations, pages 475–485
November 12-16, 2024 ©2024 Association for Computational Linguistics



a) Text-to-image generation (left) and in/outpainting (right).

b) Image-to-video generation or animation (left) and text-to-video generation (right).

Figure 1: Kandinsky 3 interface on the FusionBrain website.

offer a high level of generation quality and an interaction web-system via API (Midjourney, 2022; Pika, 2023; Betker et al., 2023).

The development of diffusion models has enabled the design of a wide range of image manipulation techniques, such as editing (Parmar et al., 2023; Liew et al., 2022; Mou et al., 2023; Lu et al., 2023), in/outpainting (Xie et al., 2023), style transfer (Zhang et al., 2023b), and image variations (Ye et al., 2023). These approaches are of particular interest to the community and are also being implemented in user interaction systems (Midjourney, 2022; Betker et al., 2023; Razzhigaev et al., 2023).

T2I models have extensive knowledge of the relationship between visual and textual concepts. This allows them to be used as a backbone for models that expand the scope of generative AI to I2V (Karras et al., 2023), T2V (Singer et al., 2023; Blattmann et al., 2023; Arkhipkin et al., 2023; Gupta et al., 2023), text-to-3D generation (Poole et al., 2023; Lin et al., 2023; Raj et al., 2023), etc.

For a long time, the key disadvantage of diffusion models remained the speed of inference, which requires a large number of steps in the reverse diffusion process. Recently these limitations have been significantly overcome by the speed-up and distillation methods for diffusion models (Meng et al., 2023; Sauer et al., 2023). This increases the prospects for creating multifunctional generative frameworks based on diffusion models and their use through online applications and web editors.

3 Demo System

Kandinsky 3 model underlies a comprehensive user interaction system with free access. The system contains different modes for image and video generation, and for image editing. Here we describe the functionality and capabilities of our two key user interaction resources — Telegram bot and FusionBrain website.

FusionBrain is a web-editor that supports loading images from the user, and saving generated images and videos (Figure 1). The system accepts text prompts in Russian, English and other languages. It is also allowed to use emoji in the text description. The maximum prompt size is 1000 characters[5]. In terms of generation tasks, this web editor provides the following options:

- **Text-to-image generation** with maximum resolution 1024 × 1024 and the ability to choose the aspect ratio. In the `Negative prompt` field, the user can specify which information (e.g., colors) the model should not use

[5] A detailed API description can be found at `https://fusionbrain.ai/docs/en/doc/api-dokumentaciya/`.

for generation. There are also options for zoom in/out, choosing the generation style and prompt beautification (Section 5.1). For details of the base T2I model, see Section 4.

- **Inpainting/outpainting** are tools for editing an image by adding or removing individual objects or areas. Using the eraser allows one to highlight areas that can be filled in with or without a new text description. The sliding window can expand the image boundaries and further generate new areas of image. The web editor allows user to upload starting image or reuse the generation result. For implementation description see Section 5.3.

- **Animation.** This is an **image-to-video** generation based on the T2I scene generation using Kandinsky 3. The user can set up to 4 scenes by describing each scene using a text prompt. Each scene lasts 4 seconds, including the transition to the next. For each scene, it is possible to choose the direction of camera movement. For more details see Section 5.6.

- **Text-to-video generation.** Creating smooth and realistic videos in a 512×512 resolution with FPS $= 32$ using the text-to-video model Kandinsky Video (Arkhipkin et al., 2023), which is based on the Kandinsky 3 model. See also Section 5.7.

Telegram bot provides all the same options as the FusionBrain website, except in/outpainting. It also has a number of additional features:

- **Distilled model.** There is a choice of Kandinsky 2.2 (Razzhigaev et al., 2023), Kandinsky 3 or distilled version (Section 5.2).

- **Image editing.** This includes: style transfer using a guidance image or text prompt, image fusion, image-text fusion, and creation of the image variations (Section 5.4). We also deployed Custom Face Swap 5.5 for generating images using photos with real people.

Table 1: Kandinsky 3 models parameters.

Architecture part	Params	Freeze
Text encoder (Flan-UL2 20B)	8.6B	True
Denoising U-Net	3.0B	False
Image decoder (Sber-MoVQGAN)	0.27B	True
Total parameters	11.9B	

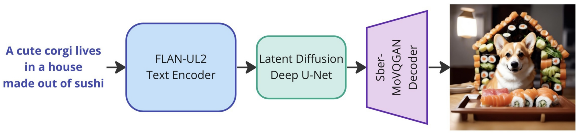

A cute corgi lives in a house made out of sushi
FLAN-UL2 Text Encoder
Latent Diffusion Deep U-Net
Sber-MoVQGAN Decoder

Figure 2: Architecture of the text-to-image model Kandinsky 3. It consists of a text encoder, a latent conditioned diffusion U-Net, and an image decoder.

4 Text-to-Image Model Architecture

Overview. Kandinsky 3 is a latent diffusion model, which includes a text encoder for processing a prompt from the user, a U-Net-like network (Ronneberger et al., 2015) for predicting noise, and a decoder for image reconstruction from the generated latent (Figure 2). For the text encoder, we use the encoder of the Flan-UL2 20B model (Tay, 2023; Tay et al., 2022), which contains 8.6 billion parameters. As an image decoder, we use a decoder from Sber-MoVQGAN (Arkhipkin et al., 2024). The text encoder and image decoder were frozen during the U-Net training. The whole model contains 11.9 billion parameters (Table 1).

Diffusion U-Net. To decide between large transformer-based models (Dosovitskiy et al., 2021; Liu et al., 2021; Ramesh et al., 2021) and convolutional architectures, both of which have demonstrated success in computer vision tasks, we conducted more than 500 experiments and noted the following key insights:

- Increasing the network depth while reducing the total number of parameters gives better results in training. A similar idea of residual blocks with bottlenecks was exploited in the ResNet-50 (He et al., 2016) and BigGAN-deep architecture (Brock et al., 2019);

- We decided to process the latents at the first network layers using convolutional blocks only. At later stages, we introduce transformer layers in addition to convolutional ones. This choice of architecture ensures the global interaction of image elements.

Thus, we settled on the ResNet-50 block as the main block for our U-Net. Using bottlenecks in residual blocks made it possible to double the number of convolutional layers, while maintaining approximately the same number of parameters as without bottlenecks. At the same time, the depth

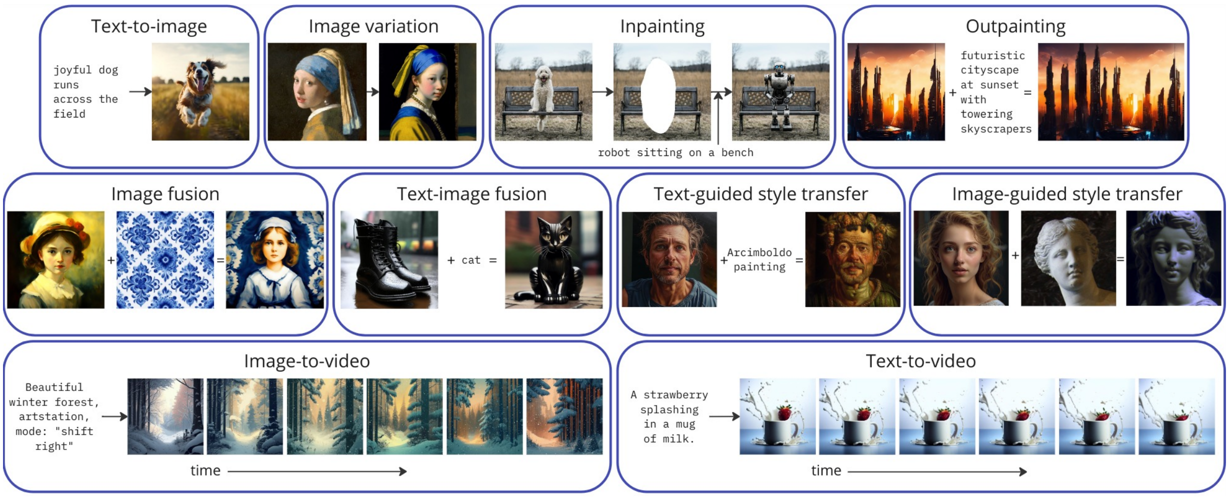

Text-to-image
joyful dog runs across the field
Image variation
Inpainting
robot sitting on a bench
Outpainting
futuristic cityscape at sunset with towering skyscrapers
+
=
Image fusion
+
=
Text-image fusion
+ cat =
Text-guided style transfer
+
Arcimboldo painting
=
Image-guided style transfer
+
=
Image-to-video
Beautiful winter forest, artstation, mode: "shift right"
time
Text-to-video
A strawberry splashing in a mug of milk.
time

Figure 3: Inference regimes of Kandinsky 3 model.

of our new architecture has increased by 1.5 times compared to Kandinsky 2 (Razzhigaev et al., 2023).

At the higher levels of the upscale and downsample parts, we placed our implementation of convolutional residual BigGAN-deep blocks. At lower resolutions, the architecture includes self-attention and cross-attention layers. The complete scheme of our U-Net architecture and a description of our residual BigGAN-deep blocks can be found in Appendix A.

5 Extensions and Features

5.1 Prompt Beautification

Many T2I diffusion models suffer from the dependence of the visual generation quality on the level of detail in the text prompt. In practice, users have to use long, redundant prompts to generate desirable images. To solve this problem, we have built a function to add details to the user's prompt using LLM. A prompt is sent to the input of the language model with a request to improve the prompt, and the model's response is sent as the input into Kandinsky 3 model. We used Neural-Chat-7b-v3-1 (Lv et al., 2023), based on Mistral 7B (Jiang et al., 2023)), with the following instruction: ### System:\nYou are a prompt engineer. Your mission is to expand prompts written by user. You should provide the best prompt for text to image generation in English. \n### User:\n{prompt}\n### Assistant:\n. Here {prompt} is the user's text. Example of generation for the same prompt with and without beautification are presented in the Appendix D.1. In general, human preferences are more inclined towards generations with prompt beautification (Section 7).

5.2 Distilled Model

Inference speed is one of the key challenges for using diffusion models in online-applications. To speed up our T2I model we used the approach from (Sauer et al., 2023), but with a number of significant modifications (see Appendix A). We trained a distilled model on a dataset with 100k highly-aesthetic image-text pairs, which we manually selected from the pretraining dataset (Section 6). As a result, we speed up Kandinsky 3 by 3 times, making it possible to generate an image in only 4 passes through U-Net. However, like in (Sauer et al., 2023), we had to sacrifice the text comprehension quality, which can be seen by the human evaluation (Figure 5). Generation examples by distilled version can be found in Appendix D.2.

Refiner. We observed that the distilled version generated more visually appealing examples than the base model. Therefore, we propose an approach that uses the distilled version as a refiner for the base model. We generate the image using the base T2I model, after which we noise it to the second step out of the four that the distilled version was trained on. Next, we generate the enhanced image by doing two steps of denoising using the distilled version.

5.3 Inpainting and Outpainting

We initialize the in/outpainting model by the Kandinsky 3 weights in GLIDE manner (Nichol et al., 2022). We modify the input convolution layer of U-Net so that it takes 9 channels as input: 4 for the original latent, 4 for the image latent, and one channel for the mask. We zeroed the additional weights, so training starts with the base model. For training, we generate random masks of the following forms: rectangular, circles, strokes, and arbitrary form. For every image sample we use up to 3 unique masks. We use the same dataset as for the training base model (Section 6) with generated masks. Additionally, we finetune our model using object detection datasets and LLaVA (Liu et al., 2023) synthetic captions.

5.4 Image Editing

Kandinsky 2 (Razzhigaev et al., 2023) natively supported images fusion technique through a complex architecture with image prior. Kandinsky 3 has a simpler structure (Figure 2), allowing it to be easily adapted to existing image manipulation approaches.

Fusion and variations. Kandinsky 3 also provides generation using an image as a visual prompt. To do this, we extended an IP-Adapter-based approach (Ye et al., 2023). To implement it based on our T2I generation model, we used ViT-L-14, finetuned in the CLIP pipeline (Radford et al., 2021), as an encoder for visual prompt. For image-text fusion, we get CLIP-embeddings for input text and image, and sum up the cross-attention outputs for them. To create image variations, we get the visual prompt embeddings and feed them to the IP-Adapter. For image fusion, the embeddings for each image are summed with weights and fed into the model. Thus, we have three inference options (Figure 3). We trained our IP-Adapter on the COYO 700m dataset (Byeon et al., 2022).

Style transfer. We found that the IP Adapter-based approach does not preserve the shape of objects, so we decided to train ControlNet (Zhang et al., 2023a) in addition to our T2I model to consistently change the appearance of the image, preserving more information compared to the original one (Figure 3). We used the HED detector (Xie and Tu, 2015) to obtain the edges in the image fed to the ControlNet. We train model on the COYO 700m dataset (Byeon et al., 2022).

5.5 Custom Face Swap

This service allows one to generate images with real people who are not present in the Kandinsky 3 training set without additional training. The pipeline consists of several steps, including: creating a description of a face on an uploaded photo using the OmniFusion VLM model (Goncharova et al., 2024), generating an image based on it using Kandinsky 3, and finally face detection and then transferring the face from the uploaded photo to generated one using GHOST models (Groshev et al., 2022). Also at the end, enhancement of the transferred face images is done using the GFPGAN model (Wang et al., 2021). Examples are presented in Appendix D.3.

5.6 Animation

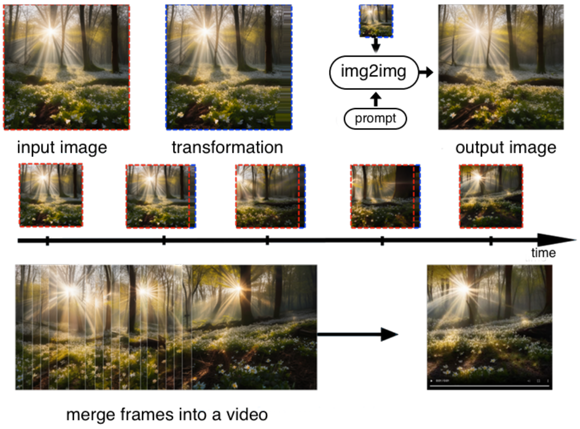


Figure 4: Image-to-Video generation. The input image undergoes a right shift transformation. The result enters the image-to-image process to eliminate transformation artifacts and update the semantic content guided by the text prompt.

Our I2V generation pipeline is based on the Deforum technique (Deforum, 2022) and consists of several stages as shown in Figure 4. First, we convert the image into a 2.5D representation using a depth map, and apply spatial transformations to the resulting scene to induce an animation effect. Then, we project a 2.5D scene back onto a 2D image, eliminate translation defects and update semantics using image-to-image (I2I) techniques. More details can be found in Appendix C.

5.7 Text-to-Video Generation

We created the T2V generation pipeline (Arkhipkin et al., 2023), consisting of two models – for keyframes generation and for interpolation. Both of them use the pretrained Kandinsky 3 as a backbone. Please refer to the main paper for additional details and results regarding the T2V model.

6 Data

Our dataset for the T2I model training consists of popular open-source datasets and our internal

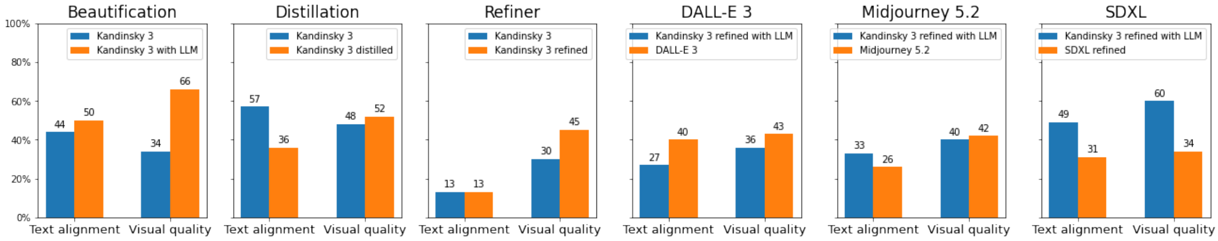


Figure 5: Human evaluation results on DrawBench (Saharia et al., 2022).

data of approximately 150 million text-image pairs. To improve data quality, we used several filters: aesthetics quality[6], watermarks detection[7], CLIP similarity of the image with the text (Radford et al., 2021), and detection of duplicates with perceptual hash (Zauner, 2010). Using these filters, we created multimodal datasets processing framework[8].

We divided all the data into two categories. We used the first at the initial stages of low-resolution pretraining and the second for mixed and high-resolution fine-tuning. The first category includes open text-image datasets such as LAION-5B (Schuhmann et al., 2022) and COYO-700M (Byeon et al., 2022), and data that we collected from the Internet. The second category contains the same datasets but with stricter filters, especially for the image aesthetics quality. For training details, please refer to the Appendix B.

7 Human Evaluation

We found that when a high level of generation quality is achieved, FID values do not correlate well with visually noticeable improvements. For the previous version of Kandinsky model (Razzhigaev et al., 2023) we reported FID, but in this work we focused on human evaluation results for model comparison.

We conducted side-by-side (SBS) comparisons between the refined version of Kandinsky 3 with beautification and other competing models: Midjourney 5.2 (Midjourney, 2022), SDXL (Podell et al., 2023) and DALL-E 3 (Betker et al., 2023). For SBS we used generations by prompts from DrawBench dataset (Saharia et al., 2022). We also compared our base T2I model with a distilled and refined version, as well as a version with prompt beautification. Each of the 12 people chose the best image from the displayed image pairs based on two criteria separately: 1) alignment between image content and text prompt, and 2) visual quality of the image. Each pair was shown to 5 different people out of 12. The group of estimators included people with various educational backgrounds, such as an economist, engineer, manager, philologist, sociologist, programmer, financier, lawyer, historian, journalist, psychologist, and editor. The participants ranged in age from 19 to 45. We also compared our base T2I model with a distilled version. Each of the 12 people chose the best image according to alignment between image content and text prompt, and visual quality of the image.

According to the results for all categories (Figure 5), prompt beautification has significantly improved the visual quality of the images. Distillation led to an increase in visual quality, but a deterioration in text comprehension. Using a distilled model as a refiner improves visual quality, while ensuring text comprehension is comparable to the base model. The low percentage values for text alignment here are due to the fact that people often chose both models.

Kandinsky 3 demonstrates competitive results for well-known SotA models, noting the complete openness of our solution, including code, checkpoints, implementation details, and the ease of adapting our model for various kinds of generative tasks.

8 Conclusion

We presented Kandinsky 3, a new open source text-to-image generative model. Based on this model, we presented our multifunctional generative framework that allows users to solve a variety of generative tasks, including inpainting, image editing, and video generation. We also presented and deployed an accelerated distilled version of our model, which, when used as a refiner for the base

[6] https://github.com/christophschuhmann/improved-aesthetic-predictor

[7] https://github.com/boomb0om/watermark-detection

[8] https://github.com/ai-forever/DataProcessingFramework

T2I model, produces SotA results among open-source solutions, according to human evaluation quality. We have implemented our framework on several platforms, including FusionBrain website and Telegram bot. We have made the code and pre-trained weights available on Hugging Face under a permissive license with the goal of making broad contributions to open generative AI development and research.

9 Ethical Considerations

We performed multiple efforts to ensure that the generated images do not contain harmful, offensive, or abusive content by (1) cleansing the training dataset from samples that were marked to be harmful/offensive/abusive, and (2) detecting abusive textual prompts.

To prevent NSFW generations we use filtration modules in our pipeline, which works both on the text and visual levels via OpenAI CLIP model (Radford et al., 2021).

While obvious queries, according to our tests, almost never generate abusive content, technically it is not guaranteed that certain carefully engineered prompts may not yield undesirable content. We, therefore, recommend using an additional layer of classifiers, depending on the application, which would filter out the undesired content and/or use image/representation transformation methods tailored to a given application.

Acknowledgments

The authors express their gratitude to Mikhail Shoytov, Said Azizov, Tatiana Nikulina, Anastasia Yaschenko, Sergey Markov, Alexander Kapitanov, Victoria Wolf, Denis Kondratiev, Julia Filippova, Evgenia Gazaryan, Vitaly Timofeev, Emil Frolov, Sergey Setrakov as well as Tagme and ABC Elementary Markup Commands.

References

Abien Fred Agarap. 2019. Deep learning using rectified linear units (relu). *Preprint*, arXiv:1803.08375.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223. PMLR.

Vladimir Arkhipkin, Andrei Filatov, Viacheslav Vasilev, Anastasia Maltseva, Said Azizov, Igor Pavlov, Julia Agafonova, Andrey Kuznetsov, and Denis Dimitrov. 2024. Kandinsky 3.0 technical report. *Preprint*, arXiv:2312.03511.

Vladimir Arkhipkin, Zein Shaheen, Viacheslav Vasilev, Elizaveta Dakhova, Andrey Kuznetsov, and Denis Dimitrov. 2023. Fusionframes: Efficient architectural aspects for text-to-video generation pipeline. *Preprint*, arXiv:2311.13073.

Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. 2022. ediff-i: Text-to-image diffusion models with ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*.

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, Wesam Manassra, Prafulla Dhariwa, Casey Chu, Yunxin Jiao, and Aditya Ramesh. 2023. Improving image generation with better captions.

Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. 2020. Adabins: Depth estimation using adaptive bins. *arXiv:2011.14141 [cs.CV]*.

Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. 2023. Align your latents: High-resolution video synthesis with latent diffusion models. *CoRR*, abs/2304.08818.

Andrew Brock, Jeff Donahue, and Karen Simonyan. 2019. Large scale gan training for high fidelity natural image synthesis. *Preprint*, arXiv:1809.11096.

Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. 2022. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset.

Deforum. 2022. Deforum. https://deforum.art/.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. 2017. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Preprint*, arXiv:1702.03118.

OpenAI et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Elizaveta Goncharova, Anton Razzhigaev, Matvey Mikhalchuk, Maxim Kurkin, Irina Abdullaeva, Matvey Skripkin, Ivan Oseledets, Denis Dimitrov, and Andrey Kuznetsov. 2024. Omnifusion technical report. *Preprint*, arXiv:2404.06212.

Alexander Groshev, Anastasia Maltseva, Daniil Chesakov, Andrey Kuznetsov, and Denis Dimitrov. 2022. Ghost—a new face swap approach for image and video domains. *IEEE Access*, 10:83452–83462.

Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Li Fei-Fei, Irfan Essa, Lu Jiang, and José Lezama. 2023. Photorealistic video generation with diffusion models. *Preprint*, arXiv:2312.06662.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

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

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 448–456. JMLR.org.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Johanna Karras, Aleksander Holynski, Ting-Chun Wang, and Ira Kemelmacher-Shlizerman. 2023. Dreampose: Fashion image-to-video synthesis via stable diffusion. *CoRR*, arXiv:2304.06025.

Jun Hao Liew, Hanshu Yan, Daquan Zhou, and Jiashi Feng. 2022. Magicmix: Semantic mixing with diffusion models. *CoRR*, abs/2210.16056.

Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. 2023. Magic3d: High-resolution text-to-3d content creation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *NeurIPS*.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10012–10022.

Shilin Lu, Yanzhu Liu, and Adams Wai-Kin Kong. 2023. TF-ICON: diffusion-based training-free cross-domain image composition. *CoRR*, abs/2307.12493.

Kaokao Lv, Wenxin Zhang, and Haihao Shen. 2023. Supervised fine-tuning and direct preference optimization on intel gaudi2. Medium post.

Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik P. Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. 2023. On distillation of guided diffusion models. In *CVPR*, pages 14297–14306. IEEE.

Midjourney. 2022. Midjourney. `https://www.midjourney.com/`.

Chong Mou, Xintao Wang, Jiechong Song, Ying Shan, and Jian Zhang. 2023. Dragondiffusion: Enabling drag-style manipulation on diffusion models. *CoRR*, abs/2307.02421.

Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. 2022. GLIDE: towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 16784–16804. PMLR.

Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. 2023. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 Conference Proceedings, SIGGRAPH 2023, Los Angeles, CA, USA, August 6-10, 2023*, pages 11:1–11:11. ACM.

Pika. 2023. Pika. `https://pika.art/`.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. 2023. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *Preprint*, arXiv:2307.01952.

Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. 2023. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763.

Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Ben Mildenhall, Nataniel Ruiz, Shiran Zada, Kfir Aberman, Michael Rubenstein, Jonathan Barron, Yuanzhen Li, and Varun Jampani. 2023. Dreambooth3d: Subject-driven text-to-3d generation. *ICCV*.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8821–8831. PMLR.

Anton Razzhigaev, Arseniy Shakhmatov, Anastasia Maltseva, Vladimir Arkhipkin, Igor Pavlov, Ilya Ryabov, Angelina Kuts, Alexander Panchenko, Andrey Kuznetsov, and Denis Dimitrov. 2023. Kandinsky: An improved text-to-image synthesis with image prior and latent diffusion. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 286–295, Singapore. Association for Computational Linguistics.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. 2023. Adversarial diffusion distillation. *Preprint*, arXiv:2311.17042.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Preprint*, arXiv:2210.08402.

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. 2023. Make-a-video: Text-to-video generation without text-video data. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Yi Tay. 2023. A new open source flan 20b with ul2. https://www.yitay.net/blog/flan-ul2-20b.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier García, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Huaixiu Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. 2022. Ul2: Unifying language learning paradigms. In *International Conference on Learning Representations*.

Xintao Wang, Yu Li, Honglun Zhang, and Ying Shan. 2021. Towards real-world blind face restoration with generative facial prior. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yuxin Wu and Kaiming He. 2018. Group normalization. *arXiv:1803.08494*.

Saining Xie and Zhuowen Tu. 2015. Holistically-nested edge detection. In *Proceedings of IEEE International Conference on Computer Vision*.

Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. 2023. Smartbrush: Text and shape guided object inpainting with diffusion model. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22428–22437.

Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. 2023. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *Preprint*, arXiv:2308.06721.

Christoph Zauner. 2010. Implementation and benchmarking of perceptual image hash functions. Master's thesis, Austria.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. 2023a. Adding conditional control to text-to-image diffusion models.

Yuxin Zhang, Nisha Huang, Fan Tang, Haibin Huang, Chongyang Ma, Weiming Dong, and Changsheng Xu. 2023b. Inversion-based style transfer with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10146–10156.

A Architecture details

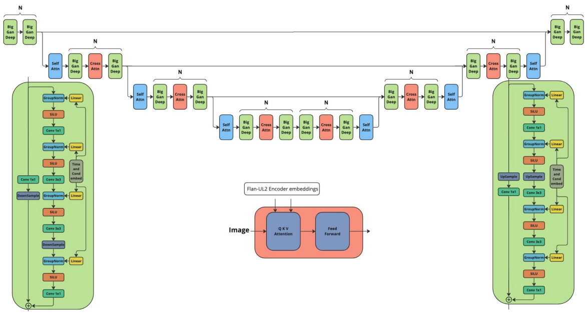


Figure 6: Kandinsky 3 U-Net architecture. The architecture is based on modified BigGAN-deep blocks (left and right – downsample and upsample blocks), which allows us to increase the depth of the architecture due to the presence of bottlenecks. The attention layers are arranged at levels with a lower resolution than the original image.

U-Net. Our version of the BigGAN-deep residual blocks (Figure 6) differs from the one proposed in (Brock et al., 2019). Namely, we use Group Normalization (Wu and He, 2018) instead of Batch Normalization (Ioffe and Szegedy, 2015) and use SiLU (Elfwing et al., 2017) instead of ReLU (Agarap, 2019). As skip connections, we implement them in the standard BigGAN residual block. For example, in the upsample part of the U-Net, we do not drop channels but perform upsampling and apply a convolution with 1×1 kernel.

Distillation. The key differences with (Sauer et al., 2023) are as follows:

- As a discriminator, we used the frozen downsample part of the Kandinsky 3 U-Net with trainable heads after each layer of resolution downsample (Figure 7);
- We added cross-attention on text embeddings from FLAN-UL2 to the discriminator heads instead of adding text CLIP-embeddings. This improved the text alignment using a distilled model;
- We used Wasserstein Loss (Arjovsky et al., 2017). Unlike Hinge Loss, it is unsaturated, which avoids the problem of zeroing gradients at the first stages of training, when the discriminator is stronger than the generator;
- We removed the regularization in the Distillation Loss, since according to our experiments it did not affect the quality of the model;
- We found that the generator quickly becomes more powerful than the discriminator, which leads to learning instability. To solve this problem, we have significantly increased the learning rate of the discriminator. For the discriminator the learning rate is $1e-3$, and for the generator it is $1e-5$. To prevent divergence, we used gradient penalty, as in the (Sauer et al., 2023).

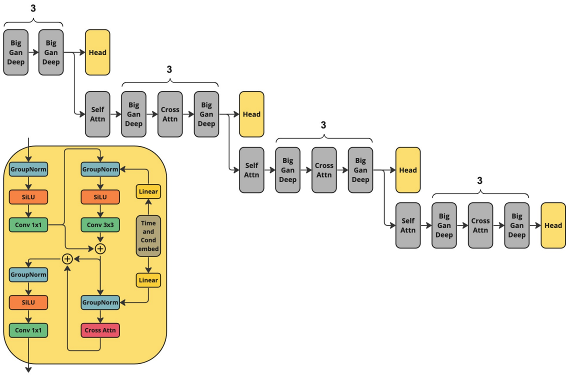


Figure 7: Discriminator architecture for distilled version of our model. Gray blocks inherit the weight of U-Net from T2I version Kandinsky 3 and remain frozen during training.

B Training strategy

We divided the training process into several stages to use more data and train the T2I model to generate images in a wide range of resolutions:

1. **256×256 resolution:** 1.1 billions of text-image pairs, batch size = 20, 600k steps, 104 NVIDIA Tesla A100;
2. **384×384 resolutions:** 768 millions of text-image pairs, batch size = 10, 500k steps, 104 NVIDIA Tesla A100;
3. **512×512 resolutions:** 450 millions of text-image pairs, batch size = 10, 400k steps, 104 NVIDIA Tesla A100;
4. **768×768 resolutions:** 224 millions of text-image pairs, batch size = 4, 250k steps, 416 NVIDIA Tesla A100;
5. **Mixed resolution: $\mathbf{768^2 \leq W \times H \leq 1024^2}$**, 280 millions of text-image pairs, batch size = 1, 350k steps, 416 NVIDIA Tesla A100.

C Animation pipeline details

The scene generation process involves depth estimation along the z-axis in the interval $[(z_{\text{near}}, z_{\text{far}})]$. Depth estimation utilizes AdaBins (Bhat et al., 2020). The camera is characterized by the coordinates (x, y, z) in 3D space, and the direction of view, which is set by angles (α, β, γ). Thus, we set the trajectory of the camera motion using the dependencies $x = x(t)$, $y = y(t)$, $z = z(t)$, $\alpha = \alpha(t)$, $\beta = \beta(t)$, and $\gamma = \gamma(t)$. The camera's first-person motion trajectory includes perspective projection operations with the camera initially fixed at the origin and the scene at a distance of z_{near}. Then, we apply transformations by rotating points around axes passing through the scene's center and translating to this center. Due to the limitations of a single-image-derived depth map, addressing distortions resulting from camera orientation deviations is crucial. We adjust scene position through infinitesimal transformations and employ the I2I approach after each transformation. The I2I technique facilitates the realization of seamless and semantically accurate transitions between frames.

D Additional generation examples

D.1 Prompt beautification

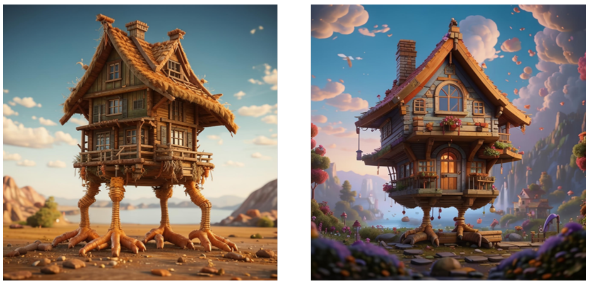


Figure 8: Prompt: `A hut on chicken legs`. Without/With LLM.

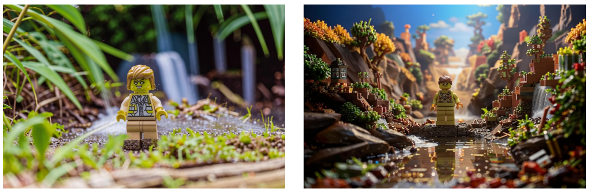


Figure 9: Prompt: `Lego figure at the waterfall`. Without/With LLM.

D.2 Distillation and prior works

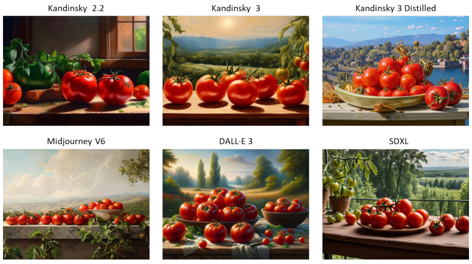


Figure 10: Prompt: `Tomatoes on a table, against the backdrop of nature, a still life painting depicted in a hyper realistic style.`

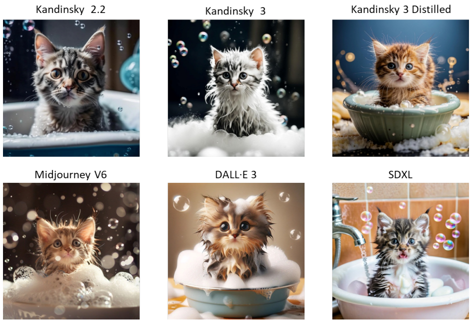


Figure 11: Prompt: `Funny cute wet kitten sitting in a basin with soap foam, soap bubbles around, photography.`

D.3 Custom Face Swap

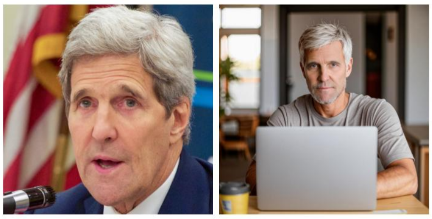


Figure 12: Real photo on the left. Name is anonymised. Prompt: `@Name is sitting at his laptop.`

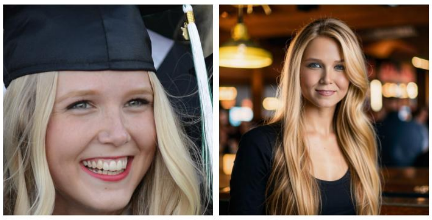


Figure 13: Real photo on the left. Name is anonymised. Prompt: `@Name at the bar, photo.`