# Comparative Analysis of Text-to-Image Generation Models

Komal[1], Sarbjeet Singh[2]

{komalaul28@gmail.com[1], sarbjeet@pu.ac.in[2]}

University Institute of Engineering and Technology, Panjab University, Chandigarh, India[1,2].

**Abstract.** Text-to-image generation is one of the emerging fields in the generative artificial intelligence (GAI). Most of the models focus on the realism and the visual fidelity of the image but fail to capture semantic meaning behind it. So, the major challenge for the models is to ensure semantic coherence and contextual relevance between the text and the generated image. This work utilizes two state-of-the-art text-to-image generative models, Stable Diffusion & Kandinsky for comparative analysis. Both of these models are fine-tuned on the MJ-Bench alignment dataset using LoRA. Different quantitative measures, including CLIP score, SSIM, and LPIPS, and qualitative evaluation through visual comparison and human evaluation are used. The results of the experiment provide an insight into the strengths and limitations of the different models, and also show the impact of fine-tuning on the quality and fidelity of the image, providing a better understanding of how each model behaves in text-to-image generation.

**Keywords:** Generative Artificial Intelligence (GAI) models, stable diffusion, kandinsky, low-rank adaptation, SSIM, CLIP score, LPIPS.

## 1 Introduction

Text-to-Image generation is one of the subfields of Generative Artificial Intelligence (GAI) that mainly focuses on automatically creating realistic or imaginative images based on the natural language descriptions given in the prompt by the user [1],[2],[3]. In simple terms, it can be described as the process of interpreting the textual input, which can either be a simple phrase or a complex narrative, and producing corresponding visual output that matches with the description provided by the user as the input [4].

A text-to-image generation model typically consists of a number of key components, such as a language encoder, which is used to understand and process the text prompts; a generative or machine learning model (like a GAN, VAE, transformer, or diffusion model), which generates the images, and evaluation modules to check how the generated images correspond to the text prompt given by the user [1],[2], [4]. The formatters will have to create these components, considering the applicable criteria that follow.

## 2 Background

Early attempts to the text-to-image generation relied on models such as the VAEs and the basic architectures of GAN [3],[6], but these models struggled in providing precise semantic relation between the prompt and the generated image [4],[5], [10]. These models often face challenges in order to understand complex textual prompts, generating images of low quality or semantically

IICTDS 2025, December 19-20, Chandigarh, India
Copyright © 2026 EAI
DOI 10.4108/eai.19-12-2025.2366220

incoherent visuals. The performance of the models was greatly dependent on the quality of training datasets; thus, they did not generalize text prompts among different linguistic inputs very often [4].

The first text-to-image models focused on modeling pixel distributions or layouts of scenes without paying attention to deep language and semantic understanding [3]. These models were conceptually simpler, but in general they were poor at capturing high dimensional relationships between text prompts and visual concept(s), thus worsening the ability to produce meaningful and coherent images and especially with abstract or imaginative prompts [20].

Recent developments in the text-to-image generation processes, such as the combination of transformer architectures and diffusion models, improved the capability to generate high-quality and high-fidelity images from given textual prompts [9],[11]. Both the models, like Stable Diffusion v1.5 and Kandinsky 3.0, operate in the latent space, which is a high-dimensional space, instead of direct pixels, and use vision encoders like CLIP for maintaining the semantic relation between the text and the generated images. These architectures are capable of performing implicit learning of multi-modal relations and do not totally rely on explicit feature engineering [6],[8].

These generative models frequently encounter some problems, like the need for high computational resources for training as well as for inference. It needs a high demand of GPUs for efficient processing. Methods such as Low-Rank Adaptation (LoRA) and Dreambooth have been developed to overcome some of these limitations by enabling lightweight fine-tuning of them for generating text-to-image, making them more adaptable to a certain domain or dataset like MJ-Bench [16],[17],[18].

Combining various techniques such as engineered prompts, prompt tuning, or fine-tuning, when applied to specific datasets, can result in similar performance as full retraining [21].Such approaches to these models can provide better interpretability of prompts, reduced training costs, and faster experimentation, which is highly beneficial in both academic and resource-constrained research [19].

### 2.1 Key Models and Their Evolution

The evolution of T2I models has quickly demonstrated the progression of different architectural approaches, with each one built upon the advancements of the previous one in the models:

Generative Adversarial Networks (GANs): It was introduced in 2014, and these models' operations involve a generator and discriminator in an adversarial process in the image generation. GAN models greatly enhanced the realism of generated images; expecting that the models produce realistic-looking images came with challenges such as training instability and mode collapse, which leads to a lack of variety amongst generated images. Further improvements such as AttnGAN (adding attention mechanisms for finer control) and GAN-CLS (Conditional Latent Space for stable training) have been done in order to handle these limitations to some extent. Later versions like StackGAN, StackGAN++, HDGAN, etc., improved the image quality and realism of the output image. DualAttn-GAN proposed dual attention mechanisms for improving the fine-grained text-image alignment [5], [13].

Variational Autoencoders (VAEs) provided a probabilistic way so diverse images could be generated but sometimes at the expense of image quality [4], producing blurry images as a result.

Building on a transformer architecture used in natural language, the OpenAI DALL-E model was one of the first models to adopt an autoregressive approach for image generation on massive datasets, generating a diversity of outputs [10]. Later versions like DALL-E 2 made it more

realistic and capable [9]. Google Brain's Imagen is also a hybrid of transformer language modeling and diffusion processes [8], [11].

Currently diffusion models include Stable Diffusion, Imagen, and DALLE 2, which are regarded as state-of-the-art. They overcome the shortcomings of GANs and VAEs by producing images from an iterative denoising process [7][8][9]. Diffusion models are known for being stable, producing high-quality output, and being able to generate detailed and semantically correct pictures. Stable Diffusion is an open-source model that provides the ability to customize and make it more accessible to the researchers and students [2]. Latent diffusion models further improve the efficiency by working in the latent space and not dealing with the pixels directly [8].

Stable Diffusion is one of the powerful text-to-image generation models that is open source and uses a particular type of deep learning model called Latent Diffusion Model (LDM). In contrast with other diffusion models that process directly on image pixels, Stable Diffusion operates in a lower-dimensional space called latent space, which minimizes the computational resources [2], thus revealing its capability to run on consumer hardware with a fair GPU.

Kandinsky 3.0 —Another advanced latent diffusion model for text-to-image generation written in Python. It is an evolution of the previous models of Kandinsky 2-x and focuses on a better understanding of text, better visual quality, and multilingual prompts. Kandinsky 3.0 simplifies the architecture by implementing a one-stage generation pipeline; this means that it uses text embedding directly for generation, and no intermediate "prior" models are needed [1], [22].

## 3 Methodology

To answer the objectives of this work, the work is generally divided into two main phases which are given below :

1. Both Stable Diffusion and Kandinsky are fine-tuned on the alignment component of the MJ-Bench dataset. The main aim of the fine-tuning process is mainly to improve the semantic alignment between the input text prompt and the generated image and maintain high visual quality. This is achieved through the implementation of Low-Rank Adaptation (LoRA) procedures of effective parameter optimisation [15][16][17].

2. The fine-tuned models go for systematic evaluation to make the benchmarking consistent. This analysis brings together both quantative analysis such as CLIP score, LPIPS and SSIM, and qualitative visual analysis. The integrated methodology enables a more thorough comparison of the models, thus explaining the strengths and weaknesses of each model variant [17].

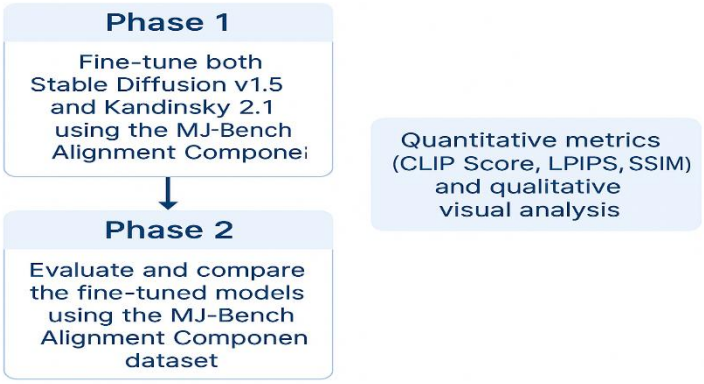



**Fig. 1** Different Phases of the Research

### 3.1 Phase 1

For this study, fine-tuning is carried out making use of the alignment part of the MJ-Bench dataset. This dataset has been specially designed for the purpose of assessing and refining the process of semantic correspondence between the text prompts and the generated images. The alignment subset includes textual prompts, the relevant images and annotations on the extent to which the generated images follow the desired meaning. This dataset is ideal for training models to understand the semantic relationships and improve the quality in terms of relevance and coherence with the given text prompt [16].

The fine tuning procedure requires the preprocessing of the dataset. The alignment part of the MJ- bench dataset can be accessed from the Hugging Face datasets library. Each example includes a prompt and a corresponding image. These pictures are resized to 512 by 512 pixels, normalized and then converted to an appropriate format to be trained. The text tokens are then tokenized using the corresponding base model tokenizer i.e. CLIP-Tokenizer. This step makes sure that both the text prompt and the image are compatible with the model architecture in training mode [11], [17].

The LoRA method of fine-tuning is being used in this work. It is a parameter-efficient fine tuning algorithm that injects few trainable parameters into each model layer. Only these small sets of parameters are changed during training, which significantly decreases the memory consumption and the computation time. When the fine-tuning is finished, the adapted model is used for inference, and it produces a new image based on the text prompts [15]. The model is now fine-tuned and ideally produces output that is more semantically accurate to the input text [16], [17].

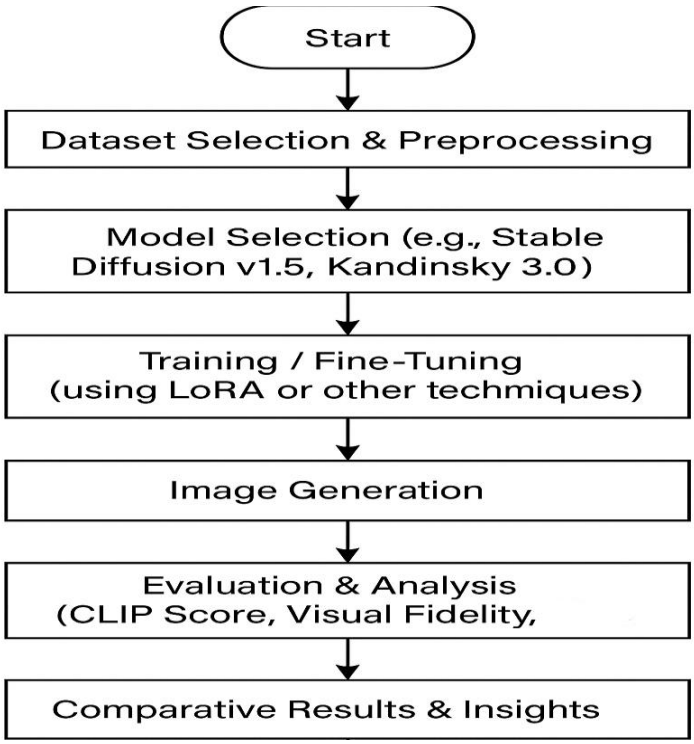



**Fig. 2** Workflow of the Model

### 3.2 Phase 2

Phase 2 of the study is focused on evaluating and comparing these fine-tuned models of text-to-image generation using the MJ-Bench dataset, whose alignment component is used to ensure consistent and reliable benchmarking. The evaluation process combines a variety of quantitative and qualitative metrics, such as the CLIP Score to measure semantic alignment [17], LPIPS to measure perceptual similarity [5], and SSIM to measure structural consistency [5], to measure the image realism, coherence and consistency to the prompt. This approach allows one to easily identify the strengths and limitations of each model and suggest a complete comparative analysis that highlights the performance of each fine-tuned model.

Quantitative techniques are, however, insufficient to convey the quality of the produced images. In this method we basically examine the realism, coherence, and aesthetic of the images. Suppose in any case qualitative metrics fail to identify any issue; then qualitative methods can be used. For example, a model may be scored high on CLIP, but it still has distorted facial features or texture incoherence problems, but we can use our qualitative inspection to spot out these types of problems [12].

## 4 Result and Evaluation

All the experiments in this work have been conducted on Google Collaboratory which offers an environment with an Intel(r) Xeon(r) CPU @ 2.20GHz, 13GB RAM, and 100GB disk space, also with GPU acceleration to allow the training and inference on the model. The implementation is

carried out using Python 3 and utilizing the Hugging Face Transformers and the Diffusers Libraries along with the supporting packages such as PyTorch, Datasets, PIL, and PEFT.

The experiments are conducted using the dataset MJ-Bench (alignment component) present in a variety of the text-image prompt pairs that can be used for the benchmarking and fine-tuning of the text-to-image generation model. This dataset offers good coverage across the semantic alignment, visual consistency and compliance with the prompt, so it can be used for evaluating the performance of generative models. The dataset is loaded from the Hugging Face Datasets library, and the default train-test set configuration was used for fine-tuning and evaluating the dataset.

For the fine-tuning, LoRA (Low-Rank Adaptation) was used on the model. So instead of training the entire model, LoRA only allows efficient tuning of the parameters by only updating specific components of the UNet attention layers. This set up provides the ability to train the LoRA parameters with the rest of the model frozen.

The result was obtained in the research work. In this section, the finding is structured into two parts: quantitative evaluation and qualitative assessment. By putting these results together, it provides insights into the relative strengths and weaknesses of both the models and also shows the impact of the fine-tuning on the performance of the text-to-image generation.

Quantitative Evaluation: The quantitative performance of the models is evaluated by using different metrics, i.e., CLIP Score, LPIPS, and SSIM. Results of both the models before fine-tuning and after fine-tuning by using different evaluation metrics are given below.

**Table 1.** Performance Comparison of Stable Diffusion Model before and after fine-tuning by using different evaluation metrics

| PROMPT | CLIP (BEFORE) | CLIP (AFTER) | LPIPS | SSIM |
|---|---|---|---|---|
| An old man reading a glowing book under a cherry blossom tree at sunset | 0.363037 | 0.386719 | 0.709526 | 0.034817 |
| A golden retriever surfing on a giant wave, cinematic lighting, hyperrealistic detail | 0.35376 | 0.381348 | 0.817169 | 0.182389 |
| A medieval knight made of glass fighting a dragon in a storm, concept art style. | 0.3137 | 0.3435 | 0.746084 | 0.1946 |

**Table 2.** Performance Comparison of Kandinsky Model before and after fine-tuning by using different evaluation metrics

| PROMPT | CLIP (BEFORE) | CLIP (AFTER) | LPIPS | SSIM |
|---|---|---|---|---|
| An old man reading a glowing book under a cherry blossom tree at sunset | 0.382324 | 0.391357 | 0.696262 | 0.141864 |

| | | | | |
|---|---|---|---|---|
| A golden retriever surfing on a giant wave, cinematic lighting, hyperrealistic detail | 0.396729 | 0.411865 | 0.650084 | 0.287942 |
| A medieval knight made of glass fighting a dragon in a storm, concept art style. | 0.27002 | 0.352783 | 0.663148 | 0.096566 |

Qualitative Assessment: While the quantitative evaluation gives the insights towards the numerical performance of the generative models, they do not fully capture the image perceptual quality, semantic accuracy, and artistic fidelity. So, in order to calculate that, we conduct the qualitative assessment of both the model outputs in **Figure [3].**




**Fig 3.** Qualitative Result of Stable Diffusion Model before and after fine-tuning

The fine-tuning demonstrates all the generated images with clearer prompt alignment and enhanced visual quality with better detail. In the first image, the old man under cherry blossoms shows a more accurately posed subject with improved facial clarity and naturalness as compared

to the earlier distant and less expressive rendering. The second picture features the surfing dog that appears to have a more dynamic and realistic surfing pose with improved fur and wave structure. The character in the third image, where it is shown as a knight and a dragon, is drawn with more details, and the figure is drawn clearly, thus rectifying the problem of lack of details and the blurred edges that are apparent in the previous image. Overall, fine-tuning helps to improve the details and realism of the images, along with the prompt fidelity of every single one of them.




**Fig 4.** Qualitative Result of Kandinsky Model before and after fine-tuning

Fine-tuning makes all the generated images more prompt-aligned, more realistic, and has a better visual coherence as shown in **Figure[4].** The old man under the cherry blossoms looks more natural with more sophisticated facial features and shows lighter lighting. The second picture, features the surfing dog that appears to have more pronounced textures, focus on the subject, and a better cinematic feel. In the third image, featuring a knight and dragon scene, the dragon becomes clearly visible after fine-tuning, making the scene match well with the desired prompt with support of the action's dynamism of the story. In general, fine-tuning makes it easier to create images with more details and better luminance balance and enriches the visual narration.

## 5 Conclusion

In conclusion, the process of fine-tuning achieves improvements for both Kandinsky and Stable Diffusion for all prompts and contributes to higher CLIP scores that are associated with semantic alignment. Kandinsky exhibits better performance on imaginative and complex prompts and thus offering better semantic fidelity, whereas Stable Diffusion provides more outputs that are perceptually rich and creatively diverse. The two models have low scores in SSIM which only emphasizes their generative character. Kandinsky is more precise and conceptually detailed, whereas Stable Diffusion is more flexible in creativity.

## References


[1] Kandinsky-2 repository: Kandinsky 2—multilingual text-to-image latent diffusion model. GitHub. Available online (2025)

[2] Stable Diffusion repository: A latent text-to-image diffusion model. GitHub. Available online (2025)

[3] Zhang, C., Zhang, C., Zhang, M., Kweon, I.S., Kim, J.: Text-to-image diffusion models in generative AI: A survey. arXiv preprint arXiv:2303.07909 (2024)

[4] Zhou, J.: Text to image generation: A literature review focus on the diffusion model. ITM Web of Conferences. Vol. 73, pp. 02037 (2025)

[5] Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 586–595 (2018)

[6] Hoe, J.T., Jiang, X., Chan, C.S., Tan, Y.P., Hu, W.: InteractDiffusion: Interaction control in text-to-image diffusion models. arXiv preprint (2023)

[7] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10674–10685 (2022)

[8] Razzhigaev, A., et al.: Kandinsky: An improved text-to-image synthesis with image prior and latent diffusion. Proceedings of EMNLP System Demonstrations. pp. 286–295 (2023)

[9] Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with CLIP latents. arXiv preprint arXiv:2204.06125 (2022)

[10] Ramesh, A., Pavlov, M., Goh, G., et al.: Zero-shot text-to-image generation. arXiv preprint arXiv:2102.12092 (2021)

[11] Radford, A., Kim, J.W., Hallacy, C., et al.: Learning transferable visual models from natural language supervision. Proceedings of the International Conference on Machine Learning. pp. 8748–8763 (2021)

[12] Oppenlaender, J.: The creativity of text-to-image generation. Proceedings of the ACM Conference. pp. 192–202 (2022)

[13] Li, B., Qi, X., Lukasiewicz, T., Torr, P.H.S.: Controllable text-to-image generation. Advances in Neural Information Processing Systems. Vol. 32 (2019)

[14] Lee, S., et al.: Diffusion explainer: Visual explanation for text-to-image stable diffusion. Online resource (2023)

[15] Hu, E., Shen, Y., Wallis, P., et al.: LoRA: Low-rank adaptation of large language models. International Conference on Learning Representations (ICLR) (2022)

[16] Kasymov, A., Sendera, M., Stypułkowski, M., Zięba, M., Spurek, P.: AutoLoRA: AutoGuidance meets low-rank adaptation for diffusion models. arXiv preprint (2024)
[17] Hessel, J., Holtzman, A., Forbes, M., le Bras, R., Choi, Y.: CLIPScore: A reference-free evaluation metric for image captioning. Proceedings of EMNLP. pp. 7514–7528 (2021)
[18] Hayou, S., Ghosh, N., Yu, B.: LoRA+: Efficient low-rank adaptation of large models. arXiv preprint (2024)
[19] Gal, R., Alaluf, Y., Atzmon, Y., et al.: An image is worth one word: Personalizing text-to-image generation using textual inversion. arXiv preprint arXiv:2208.01618 (2022)
[20] Gafni, O., Polyak, A., Ashual, O., et al.: Make-A-Scene: Scene-based text-to-image generation with human priors. arXiv preprint arXiv:2203.13131 (2022)
[21] Dehouche, N., Dehouche, K.: What's in a text-to-image prompt? The potential of stable diffusion in visual arts education. Heliyon. Vol. 9, no. 6, e16757 (2023)
[22] Arkhipkin, V., et al.: Kandinsky 3: Text-to-image synthesis for multifunctional generative framework. arXiv preprint (2024)