Summer Internship Project Report

on

“ AI Text to Image Generator ”

at

IBM

by

AKHILESH TRIPATHI

B.Tech, CSE-(AI/ML) Engineering

Under the Guidance of

Mr. Harendra Singh

Ministry of Electronics and Information Technology Government of India

NASSCOM®

futureskills prime

A MeitY - NASSCOM Digital Skilling Initiative

Abstract:

- The "AI Text to Image Generator" is a desktop-based application developed using Python's Tkinter and the Stable Diffusion model (v1.5) from Hugging Face.

- The goal of the project is to provide users with a user-friendly interface to transform any natural language text prompt into high-quality, AI-generated images.

- The model leverages deep learning techniques and runs efficiently on both CPU and GPU using PyTorch.

- By integrating a powerful AI backend with an intuitive frontend, this project brings cutting-edge generative capabilities directly to the desktop.

- The application also includes image preview and download functionality, making it suitable for students, researchers, and hobbyists interested in AI and digital creativity.

- This project bridges the gap between complex deep learning models and easy accessibility for end users, showcasing how advanced AI models can be used interactively in day-to-day desktop environments.

Introduction:

- Recent advancements in artificial intelligence have enabled machines to generate highly realistic images from textual descriptions.

- One such breakthrough model is Stable Diffusion, which enables powerful text-to-image generation.

- This project introduces a desktop-based GUI application that utilizes the Stable Diffusion v1.5 model to convert user prompts into visual art.

- Designed with simplicity in mind, the application is built using Tkinter, making it lightweight and easy to use across systems.

- The user simply inputs a textual prompt, and the model generates a relevant image, which is displayed within the application and can be downloaded for later use.

- This application offers not just a creative tool but also a practical demonstration of AI's capabilities in the field of multimodal learning, bridging text and vision.

- It provides a stepping stone for beginners in AI while offering functionality powerful enough for advanced use cases like content creation, storytelling, and concept visualization.

Objectives:

The primary objectives of this project are:

- To develop a simple, user-friendly GUI application for generating images from text prompts using AI.
- To integrate the Stable Diffusion v1.5 model via Hugging Face's diffusers library.
- To ensure compatibility and efficiency on both CPU and GPU using PyTorch.
- To demonstrate real-world application of generative AI in visual content creation.
- To allow users to interactively generate and download high-resolution images.
- To promote understanding and usability of deep learning models among beginners.
- To make AI-based image generation accessible without needing technical expertise.
- To encourage creativity and artistic exploration through AI tools.
- To bridge the gap between abstract AI concepts and usable software tools.
- To highlight the importance of multimodal AI systems in future innovations.

Literature Review / Background:

- Generative models have evolved significantly over the past few years, with GANs (Generative Adversarial Networks) and diffusion models leading the charge.

- Among diffusion models, **Stable Diffusion**, released by Stability AI and integrated with Hugging Face, stands out for its balance between quality and computational efficiency. This model takes a denoising approach to generate images conditioned on text inputs.

- The inspiration behind this project stems from the growing popularity of AI in the creative industry. Tools like DALL·E, Midjourney, and Stable Diffusion have shown how deep learning can empower artists and developers alike.

- Tkinter, a standard Python GUI library, was chosen for its simplicity and availability across platforms.

- The use of PyTorch allows leveraging CUDA for GPU acceleration, making the image generation faster.

- This project combines these elements to build a tool that reflects current trends in applied AI and democratizes access to powerful generative technology.

Methodology:

The development of this project follows a modular and practical methodology:

- First, the Stable Diffusion v1.5 model is loaded using the diffusers library and initialized on the appropriate device (CPU or GPU) using PyTorch.

- The frontend is created using Tkinter, which handles prompt input, button interactions, and image display.

- When the user enters a prompt and clicks on "Generate Image", the application calls the model pipeline to generate an image based on that text using a guidance scale of 7.5.

- The result is then resized for GUI compatibility and shown on a Tkinter canvas.

- The image is also stored in memory for potential download. Additionally, a "Download Image" button allows the user to save the result as a .png file using Python's file dialog.

- Exception handling ensures robustness.

- This clear separation of logic, model, and interface ensures modularity, ease of debugging, and extendability in the future.

Results and Analysis:

- The application performs reliably across different systems, demonstrating successful integration of AI with a graphical interface.

- On a system with a GPU, image generation takes approximately 5–8 seconds per prompt, whereas on a CPU it ranges from 10–15 seconds.

- The output images are visually appealing and accurately reflect the semantic meaning of the text prompts. The Tkinter interface effectively renders the images and facilitates easy download.

- The results validate that Stable Diffusion is suitable for real-time generation tasks even outside of a cloud environment.

- During testing, various types of prompts—abstract, object-based, landscape, and imaginative—were used, and the model showed consistent quality across them.
- Users reported that the app was intuitive and required no prior technical knowledge.

- The analysis confirms that with minimal resources, generative AI can be deployed on local machines for educational, artistic, or prototyping purposes.

Challenges:

Developing this application posed several technical and functional challenges:

- The first was handling the model's size and performance, especially on systems without GPU support.

- Running Stable Diffusion on CPU was considerably slower, which required optimizing the workflow to prevent the GUI from freezing.

- Secondly, integrating high-resolution image display in Tkinter required resizing images without compromising quality.

- Memory management was also essential, as multiple images could be generated in one session. Ensuring that the image reference persisted on the canvas was a challenge resolved using global variables.

- Additionally, handling user errors (e.g., empty prompts or interrupted downloads) needed robust exception handling.

- The application also had to deal with package compatibility across environments.

- Overcoming these hurdles helped enhance the project’s usability and robustness, ensuring it works across different system configurations.

Future Scope:

This project lays the foundation for many future enhancements:

- One major upgrade could be adding **prompt customization options** such as guidance scale, image resolution, and seed value for reproducibility.

- Users could be allowed to select different diffusion models (like Stable Diffusion XL or custom fine-tuned models). Support for **image-to-image** generation or **inpainting** could also be added.

- A preview gallery of generated images or a history tracker could enhance user experience. Migrating the application to a web-based GUI using frameworks like Flask or Streamlit could improve accessibility.

- Another potential is to introduce voice input for prompts, making it even more interactive. Additionally, image style transfer or filters could be integrated for artistic effects.

- As hardware capabilities improve and diffusion models evolve, such tools can become faster, more flexible, and more creative, making AI art generation mainstream in personal and professional spaces.

Conclusion:

- The AI Text to Image Generator project successfully demonstrates how powerful generative models like Stable Diffusion can be deployed in user-friendly desktop applications.

- By leveraging open-source tools and libraries, it brings cutting-edge AI capabilities to everyday users.

- The integration of Tkinter ensures platform independence and simplicity, while PyTorch and Hugging Face enable robust model performance.

- Despite challenges related to performance and system compatibility, the application remains functional, intuitive, and effective in generating high-quality images.

- This project not only showcases the power of multimodal AI but also contributes toward democratizing AI for creativity and learning.

- It reflects how the intersection of deep learning and UI design can make complex models accessible and practical for a broader audience.

- Overall, this project is a stepping stone for future innovations in AI-driven visual content generation.

References:

- Hugging Face Diffusers Library – https://huggingface.co/docs/diffusers/
- Stability AI – Stable Diffusion v1.5 Model – https://huggingface.co/runwayml/stable-diffusion-v1-5
- PyTorch – https://pytorch.org/
- Tkinter GUI Library – https://docs.python.org/3/library/tkinter.html
- PIL (Python Imaging Library) – https://pillow.readthedocs.io/en/stable/
- High-Resolution Image Synthesis with Latent Diffusion Models – https://arxiv.org/abs/2112.10752
- Official GitHub: diffusers – https://github.com/huggingface/diffusers
- Stack Overflow and Python documentation for error handling and GUI support