Readme

Model Selection and Justification :

For this project, I selected **`CompVis/stable-diffusion-v1-4`** as the foundational model for both **text-to-image** and **image-to-image** tasks. This version of Stable Diffusion is highly optimized for general-purpose generative tasks, and its open-source nature makes it ideal for customizable workflows in domain-specific applications like pharmaceutical imagery.

To enable structural control over the outputs and enforce realistic spatial layouts, I integrated **`lllyasviel/sd-controlnet-canny`**, a ControlNet variant trained on Canny edge maps. This allowed the model to condition generation on edge-detected versions of input images—an essential feature when enhancing or transforming laboratory scenes while maintaining their physical structure.

I also implemented the **`StableDiffusionImg2ImgPipeline`** from `diffusers` for refining or regenerating images based on a prompt. This pipeline enabled real-time, user-guided editing and enhancement without the need for model retraining. Together, this architecture satisfies all project constraints: open-source tooling, support for both text-to-image and image-conditioned generation, and the ability to generate content that is clearly grounded in pharmaceutical environments—such as R&D labs, cleanrooms, or bioreactor facilities.

Also added an option for **ControlNet** Just in case users want to enhance the current image while keeping the structure of outputimage same as input image.

Why not other Models :

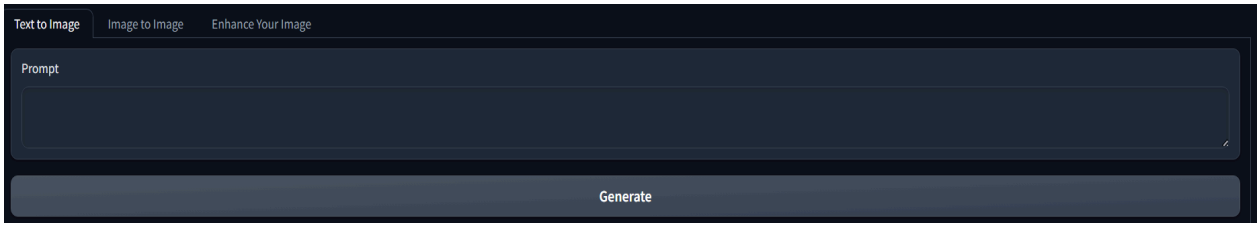
I considered other approaches such as **DreamBooth** , but they introduce additional training complexity and resource demands. While **DreamBooth** excels at teaching a model new concepts or identities from a small set of reference images, it requires GPU-intensive fine-tuning and a well-labeled dataset—making it less suitable for dynamic or user-driven generation. I tried tuning on my own dataset , but the free version of google colab crashes in between . Therefore I choose to go with the base model and for specific use cases I have prompted it in that way.

Installation & usage instructions :

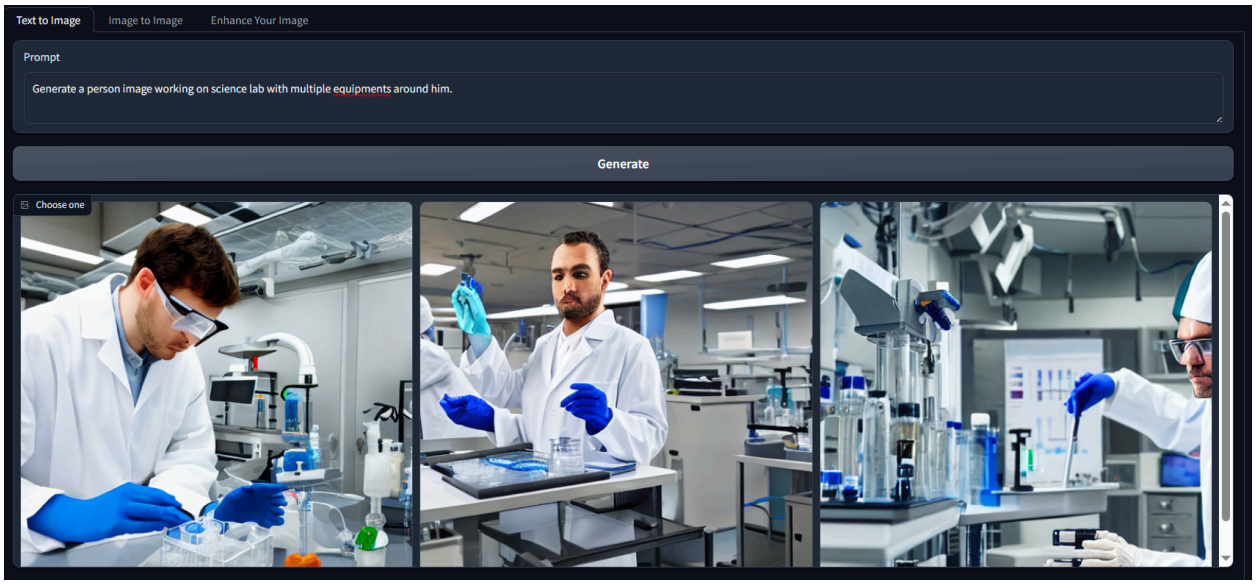
- I have shared a google colab notebook , where you have to run the cells in sequential manner .
- A HF_TOKEN is proved there in google colab in the text section , please use it and **Do not share it with anyone else .**
- Also create a HF_TOKEN key and value and allow access to it.
- You need to connect the runtime to T4 GPU
- Once to run all the cells successfully , at the end you will get a temporary link to the deployed Gradio app . you can open it in the browser or can interact in google colab notebook also . Both options are available. (for example : Running on public URL: https://e64223d300ac0cb430.gradio.live , it will be different in your case)
- **While running cells containing !pip install , you have to restart the session so that modules load properly . Make sure to connect to the T4 GPU again. And make sure all the models get installed , if you face any issue related to import error , again run the !pip install cells and restart the session .**

Screenshots :

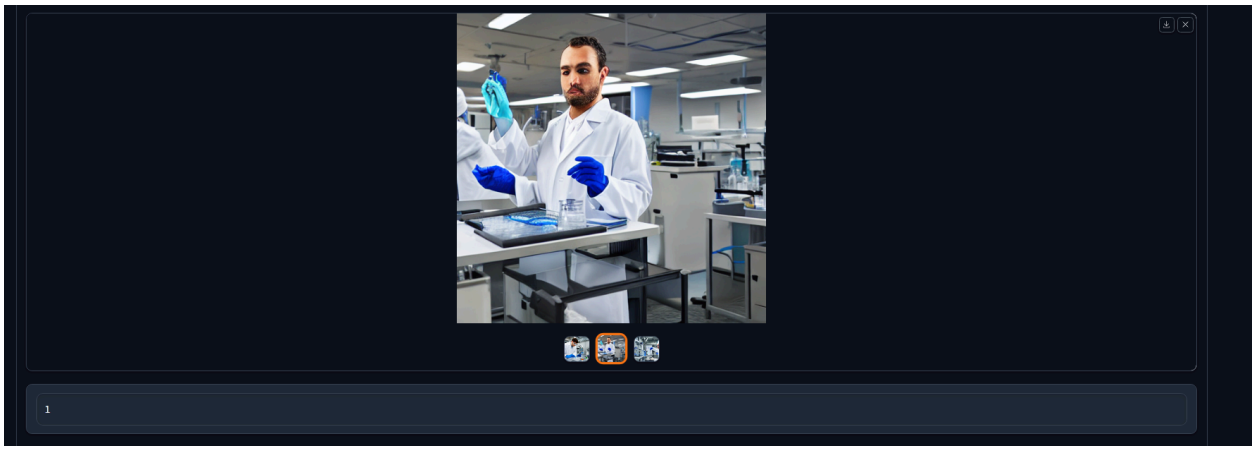
1. ***You have three tabs*** : Text to image , image to image and Enhance your image



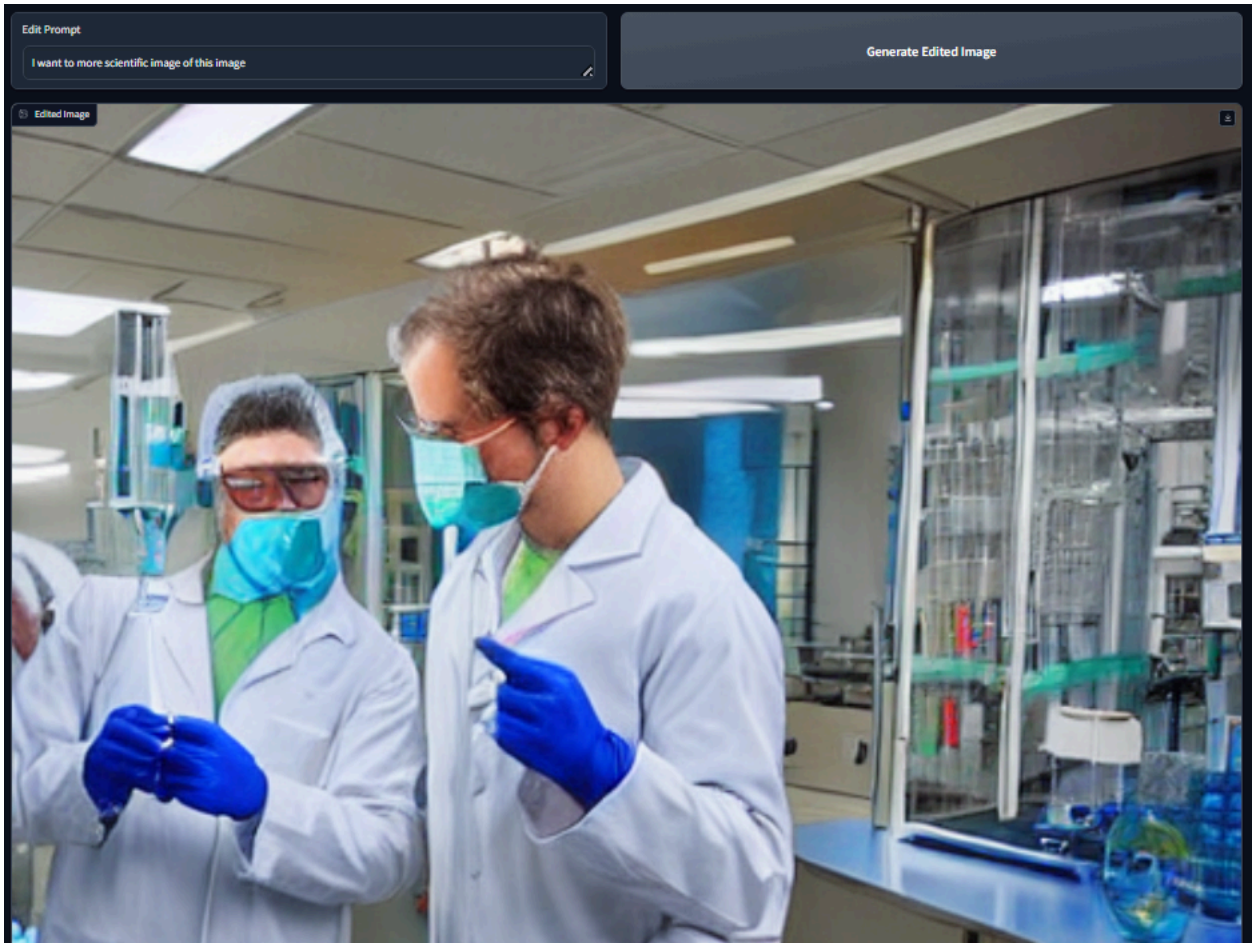
2. ***Text to Image*** : Here on entering the prompt you will get three images where you can choose one .



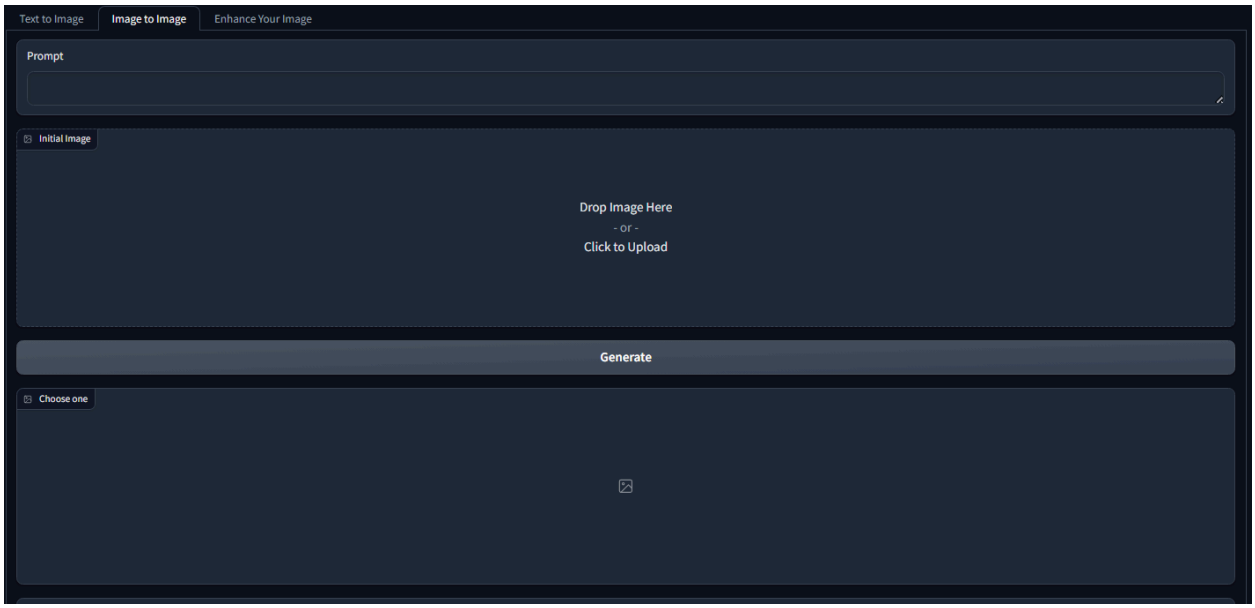
Chosen index will appear here like below image and you can download it for the enhancement using the Enhance image tab , and also you have option to regenerate in below tab.



After you have chosen the image , you can edit it using the prompt like below and it will regenerate it .

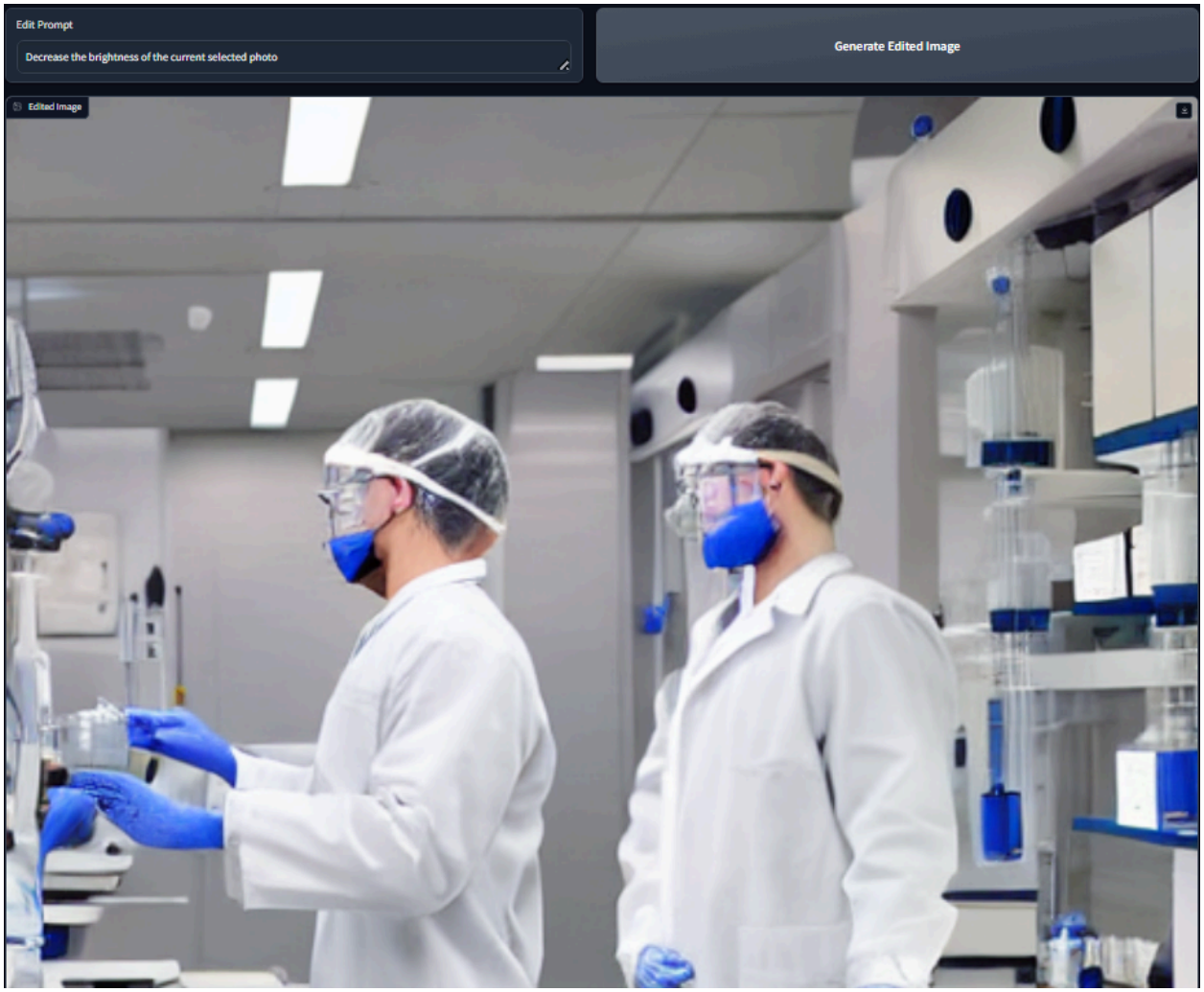

Edit Prompt
I want to more scientific image of this image
Generate Edited Image
Edited Image

3. ***Image to Image*** : Here you have option to input text as well as image

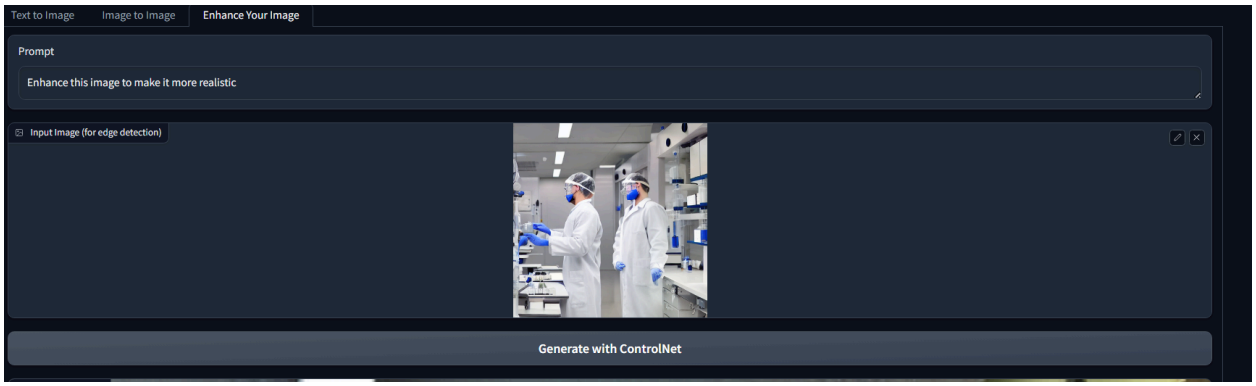


After input image and text , output looks like

It generates multiple images and here you can choose one image to go ahead and edit it using below tabs in UI.



4. ***Enhance your Image*** : Whatever image you want to enhance from above steps download it , and use that image here . This tab is especially if you want to make sure physical structure of output image remains same as input.


Text to Image
Image to Image
Enhance Your Image
Prompt
Enhance this image to make it more realistic
Input Image (for edge detection)
Generate with ControlNet

Output will look like : See how it is similar to the uploaded image, but enhanced version of it.