Task Title: AI Image Generator using Stable Diffusion

Full Name: Jacob Sam Jose

Batch: June 2025

Internship ID: VERVEOX25PY35

Submission Date: 05/ 07/2025

x

1. Objective

The objective of this task was to build an AI-based image generator using the Stable Diffusion v1.4 model in Python. The goal was to learn how to accept user prompts, generate realistic images, and handle model loading and output saving. This helped me understand diffusion-based generative AI models and practical deployment in Python.

2. Source Code

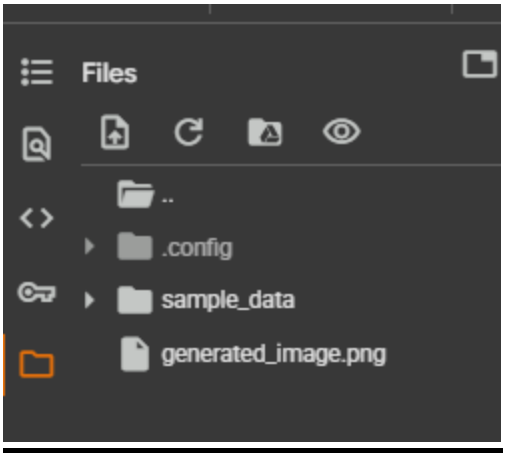
```python
from diffusers import StableDiffusionPipeline
import torch
import matplotlib.pyplot as plt

model_id = "CompVis/stable-diffusion-v1-4"
device = "cuda" if torch.cuda.is_available() else "cpu"

if device == "cuda":
    pipe = StableDiffusionPipeline.from_pretrained(
        model_id,
        torch_dtype=torch.float16
    ).to(device)
else:
    pipe = StableDiffusionPipeline.from_pretrained(model_id).to(device)

print("Model loaded successfully!")

prompt = input("Enter your prompt for image generation: ")

print("Generating image...")
result = pipe(prompt)
image = result.images[0]
print("Image generation complete.")

plt.imshow(image)
plt.axis('off')
plt.show()

image.save("generated_image.png")
print("Image saved as generated_image.png")
```

Program Output

```
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as secret in your Google Colab and restart your session.
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
  warnings.warn(
model_index.json: 100% 541/541 [00:00<00:00, 32.2kB/s]
Fetching 16 files: 100% 16/16 [02:23<00:00, 9.95s/it]
preprocessor_config.json: 100% 342/342 [00:00<00:00, 7.61kB/s]
config.json: 4.56k/? [00:00<00:00, 112kB/s]
scheduler_config.json: 100% 313/313 [00:00<00:00, 4.02kB/s]
config.json: 100% 592/592 [00:00<00:00, 7.12kB/s]
scheduler_config-checkpoint.json: 100% 209/209 [00:00<00:00, 2.47kB/s]
merges.txt: 525k/? [00:00<00:00, 6.04MB/s]
safety_checker/model.safetensors: 100% 1.22G/1.22G [00:58<00:00, 21.3MB/s]
text_encoder/model.safetensors: 100% 492M/492M [00:50<00:00, 10.2MB/s]
special_tokens_map.json: 100% 472/472 [00:00<00:00, 22.8kB/s]
tokenizer_config.json: 100% 806/806 [00:00<00:00, 10.2kB/s]
vocab.json: 1.06M/? [00:00<00:00, 8.20MB/s]
config.json: 100% 743/743 [00:00<00:00, 8.74kB/s]
config.json: 100% 551/551 [00:00<00:00, 6.65kB/s]
unet/diffusion_pytorch_model.safetensors: 100% 3.44G/3.44G [02:22<00:00, 102MB/s]
vae/diffusion_pytorch_model.safetensors: 100% 335M/335M [01:43<00:00, 3.40MB/s]
Loading pipeline components...: 100% 7/7 [00:23<00:00, 3.70s/it]
Model loaded successfully!
Enter your prompt for image generation: A futuristic server room with glowing fiber-optic cables and AI cores
Generating image...
100% 50/50 [00:07<00:00, 7.24it/s]
Image generation complete.
```

```
Image saved as generated_image.png
```

X



4 .Code Explanation

This Python script uses the **Stable Diffusion** model to generate an image based on a text prompt entered by the user. It checks if a GPU (CUDA) is available and loads the model accordingly for better performance. Once the user types a prompt, the model generates an image, displays it using matplotlib, and saves it as generated_image.png.

5. Challenges Faced

Problem: Running on CPU was very slow.

Solution: Ensured GPU is available by using Google Colab with GPU enabled (Runtime > Change runtime type > GPU).

Problem: Out of Memory Errors

Solution: torch_dtype=torch.float16