```
# Step 1: Install required libraries
!pip install -q diffusers transformers accelerate safetensors torch torchvision matplotlib
```

```
# Step 2: Import libraries
import torch
from diffusers import AutoPipelineForText2Image
from PIL import Image
import os
import matplotlib.pyplot as plt
```

```
# Step 3: Set device
device = "cuda" if torch.cuda.is_available() else "cpu"
print("Using device:", device)
```

```
Using device: cpu
```

```
# Step 4: Load Stable Diffusion Turbo model
pipe = AutoPipelineForText2Image.from_pretrained(
    "stabilityai/sd-turbo",
    torch_dtype=torch.float16 if device == "cuda" else torch.float32
)

pipe = pipe.to(device)
pipe.safety_checker = None  # Disable safety checker
```

```
model_index.json: 100%                                                 616/616 [00:00<00:00, 24.4kB/s]
Fetching 12 files: 100%                                                12/12 [00:56<00:00,  4.86s/it]
text_encoder/model.safetensors: 100%                                                   1.36G/1.36G [00:50<00:00, 23.8MB/s]
config.json:        1.87k/? [00:00<00:00, 16.8kB/s]
config.json: 100%                                                      618/618 [00:00<00:00, 2.43kB/s]
tokenizer_config.json: 100%                                            855/855 [00:00<00:00, 3.94kB/s]
special_tokens_map.json: 100%                                          574/574 [00:00<00:00, 2.06kB/s]
merges.txt:         525k/? [00:00<00:00, 975kB/s]
vocab.json:         1.06M/? [00:00<00:00, 3.71MB/s]
scheduler_config.json: 100%                                            553/553 [00:00<00:00, 2.38kB/s]
unet/diffusion_pytorch_model.safetensors: 100%                                         3.46G/3.46G [00:55<00:00, 120MB/s]
vae/diffusion_pytorch_model.safetensors: 100%                                          335M/335M [00:51<00:00, 6.14MB/s]
config.json: 100%                                                      655/655 [00:00<00:00, 8.06kB/s]
Loading pipeline components...: 100%                                                   5/5 [00:01<00:00,  3.51it/s]
You have disabled the safety checker for <class 'diffusers.pipelines.stable_diffusion.pipeline_stable_diffusion.S
```

```
# Step 5: Prompts (Chest X-ray dataset generation)
prompts = [
    "Chest X-ray of a healthy elderly patient, mild age-related changes, clear lungs, no pathological findings."
    "Chest X-ray showing patchy infiltrates consistent with atypical pneumonia",
    "Chest X-ray with ground-glass opacities throughout both lungs.",
    "Chest X-ray with bilateral pleural effusion, fluid levels present",
    "Chest X-ray showing pulmonary fibrosis with reticular opacities and reduced lung volume.",
    "Chest X-ray with prominent pulmonary vessels and mild interstitial edema.",
    "Chest X-ray with chest tube placed for pneumothorax management.",
    "Chest X-ray in erect position with full lung expansion",
    "Chest X-ray showing variation in brightness and contrast due to different imaging equipment."
]
```

```
# Step 6: Create output directory
output_dir = "synthetic_image_dataset"
os.makedirs(output_dir, exist_ok=True)
```

```
# Step 7: Generate images
images = []

for idx, prompt in enumerate(prompts):
    image = pipe(
        prompt=prompt,
        num_inference_steps=4,
        guidance_scale=0.0,
        height=512,
        width=512
    ).images[0]

    image.save(f"{output_dir}/image_{idx+1}.png")
    images.append(image)
```

```
100%  4/4 [01:22<00:00, 18.80s/it]
100%  4/4 [01:07<00:00, 16.49s/it]
100%  4/4 [01:04<00:00, 16.23s/it]
100%  4/4 [01:04<00:00, 16.02s/it]
100%  4/4 [01:03<00:00, 16.09s/it]
100%  4/4 [01:03<00:00, 15.88s/it]
100%  4/4 [01:04<00:00, 16.08s/it]
100%  4/4 [01:03<00:00, 15.81s/it]
100%  4/4 [01:04<00:00, 16.05s/it]
```

```
# Step 8: Display generated images
cols = 3
rows = (len(images) + cols - 1) // cols

plt.figure(figsize=(12, 12))
for i, img in enumerate(images):
    plt.subplot(rows, cols, i + 1)
    plt.imshow(img, cmap="gray")
    plt.axis("off")
    plt.title(f"Image {i+1}")

plt.tight_layout()
plt.show()
```

Image 1

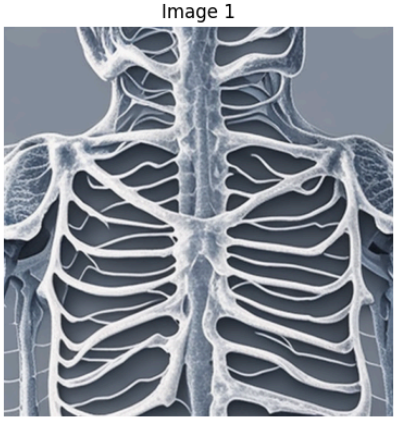

Image 2

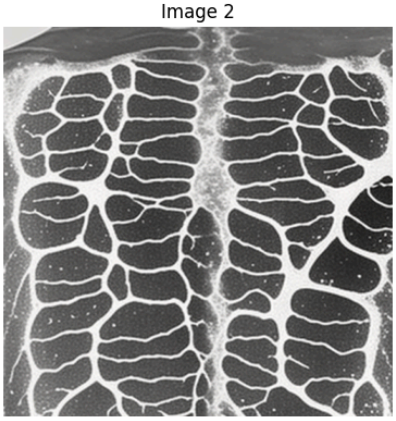

Image 3

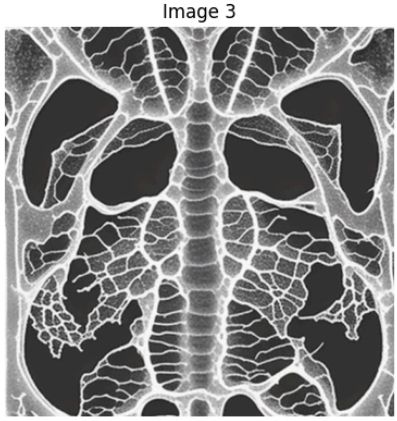

Image 4

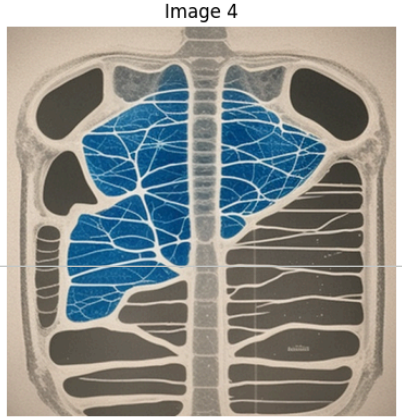

Image 5

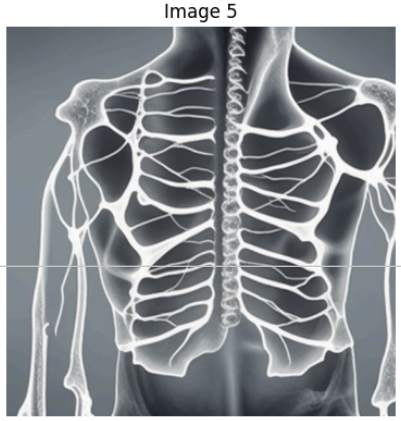

Image 6

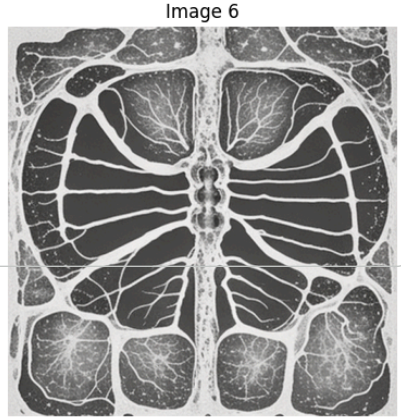

Image 7

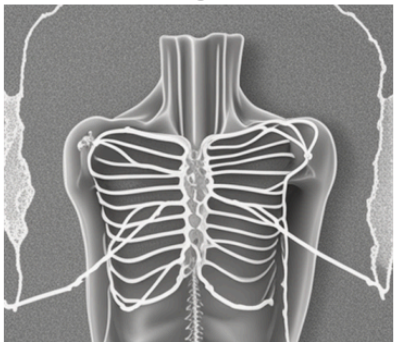

Image 8

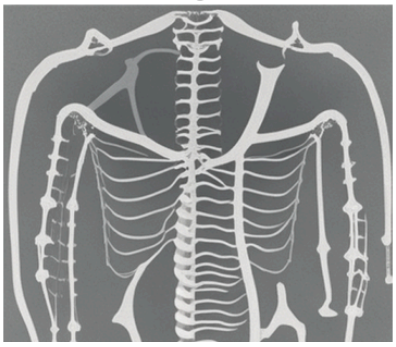

Image 9