Stable Diffusion Capabilities and Limitations

Discuss what Stable Diffusion can and can't do, to determine if it's suitable for your project.

Stable Diffusion Capabilities

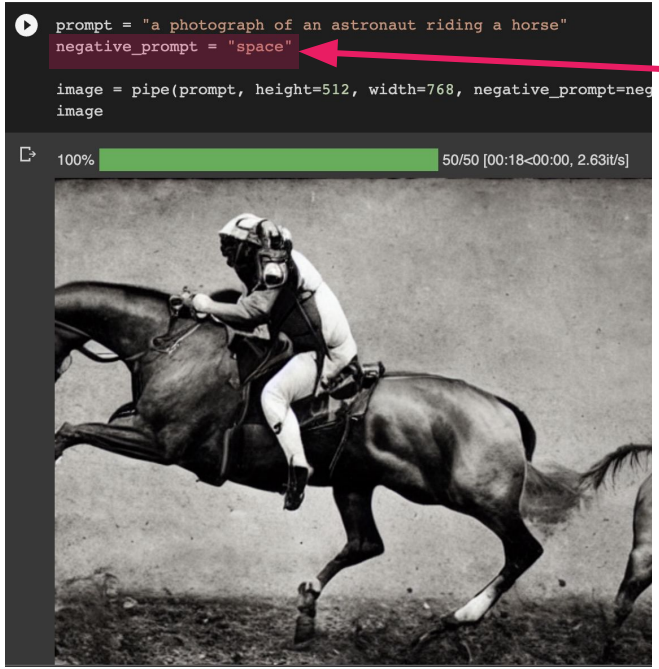
Open Source

```
prompt = "a photograph of an astronaut riding a horse"
image = pipe(prompt).images[0]  # Image here is in [PIL format](https://pi

# Now to display an image you can either save it such as:
image.save(f"astronaut_rides_horse.png")

# or if you're in a google colab you can directly display it with
image
```

100% 50/50 [00:12<00:00, 5.32it/s]

Run the code locally on your computer (if you have a GPU) or in the cloud, without any limitations or lack of flexibility.

Negative Prompts



Negate concepts from the prompt to give you creative control over the output.

Classifier Free Guidance

Choose how much the image matches the prompt, or how ‘creative’ the AI can be.

Dreambooth

Train the AI on a person, object or style that’s not in the dataset, and use that in your prompts.

Stable Diffusion Limitations

Technical Ability

- Offers lots of flexibility in the way you use it
- Needs a GPU and coding ability to run locally
- Possible to build a business around it

```
[ ] #@markdown Run to generate a grid of preview images from the last saved weigh
    import os
    import matplotlib.pyplot as plt
    import matplotlib.image as mpimg

    weights_folder = OUTPUT_DIR
    folders = sorted([f for f in os.listdir(weights_folder) if f != "0"], key=lan

    row = len(folders)
    col = len(os.listdir(os.path.join(weights_folder, folders[0], "samples")))
    scale = 4
    fig, axes = plt.subplots(row, col, figsize=(col*scale, row*scale), gridspec_k

    for i, folder in enumerate(folders):
        folder_path = os.path.join(weights_folder, folder)
        image_folder = os.path.join(folder_path, "samples")
        images = [f for f in os.listdir(image_folder)]
        for j, image in enumerate(images):
            if row == 1:
                currAxes = axes[j]
            else:
                currAxes = axes[i, j]
            if i == 0:
                currAxes.set_title(f"Image {j}")
            if j == 0:
                currAxes.text(-0.1, 0.5, folder, rotation=0, va='center', ha='cen
            image_path = os.path.join(image_folder, image)
            img = mpimg.imread(image_path)
            currAxes.imshow(img, cmap='gray')
            currAxes.axis('off')
```

Faces / Hands

Questionable Provenance

cvalenzuela Oct 20, 2022 · edited Oct 20, 2022

Hi all,

Cris here - the CEO and Co-founder of Runway. Since our founding in 2018, we've been on a mission to empower anyone to create the impossible. So, we're excited to share this newest version of Stable Diffusion so that we can continue delivering on our mission.

This version of Stable Diffusion is a continuation of the original High-Resolution Image Synthesis with Latent Diffusion Models work that we created and published (now more commonly referred to as Stable Diffusion). Stable Diffusion is an AI model developed by Patrick Esser from Runway and Robin Rombach from LMU Munich. The research and code behind Stable Diffusion was open-sourced last year. The model was released under the CreativeML Open RAIL M License.

We confirm there has been no breach of IP as flagged and we thank Stability AI for the compute donation to retrain the original model.

132 48 47 23 +

Source: Hugging Face