
Stable Diffusion web UI

A web interface for Stable Diffusion, implemented using Gradio library.

The screenshot displays the Stable Diffusion web UI interface. At the top, a dropdown menu shows the selected checkpoint: "protogenX34OfficialR_1.ckpt [60fe2f34]". Below this is a navigation bar with tabs: "txt2img", "img2img", "Extras", "PNG Info", "Checkpoint Merger", "Train", "Tokenizer", "Settings", and "Extensions". The "txt2img" tab is active, showing a text input field with the prompt "green sapling rowing out of ground, mud, dirt, grass, high quality, photorealistic, sharp focus, depth of field". Below the prompt is a "Negative prompt" field. To the right of the prompt is a "Generate" button. Below the prompt field are two style dropdowns, both set to "None". Below the style dropdowns is a "26/75" indicator. The main settings area includes a "Sampling method" dropdown set to "Euler a", a "Sampling steps" slider set to 20, and checkboxes for "Restore faces", "Tiling", and "Hires. fix". Below these are sliders for "Width" (512) and "Height" (512), and a "Batch count" dropdown set to 4. There are also sliders for "Batch size" (1) and "CFG Scale" (12). A "Seed" field shows the value "1441787169". Below the seed field is a "Script" dropdown set to "None". To the right of the settings area is a large image preview showing a generated image of a green sapling growing out of the ground. Below the image preview are five small thumbnail images. At the bottom of the interface are buttons for "Save", "Zip", "Send to img2img", "Send to inpaint", and "Send to extras". Below these buttons is a text box containing the generated image's details: "green sapling rowing out of ground, mud, dirt, grass, high quality, photorealistic, sharp focus, depth of field", "Steps: 20, Sampler: Euler a, CFG scale: 12, Seed: 1441787169, Size: 512x512, Model hash: 60fe2f34, Model: protogenX34OfficialR_1", and "Time taken: 8.62s Torch active/reserved: 3699/4702 MiB, Sys VRAM: 7020/24576 MiB (28.56%)". At the very bottom of the interface is a footer with links: "API", "Github", "Gradio", "Reload UI", and a list of dependencies: "python: 3.10.6", "torch: 1.13.1+cu117", "xformers: N/A", "gradio: 3.15.0", "commit: 983167e6".

Features

Detailed feature showcase with images: - Original txt2img and img2img modes - One click install and run script (but you still must install python and git) - Outpainting - Inpainting - Color Sketch - Prompt Matrix - Stable Diffusion Upscale - Attention, specify parts of text that the model should pay more attention to - a man in a ((**tuxedo**)) - will pay more attention to tuxedo - a man in a (**tuxedo** : 1.21) - alternative syntax - select text and press **Ctrl+Up** or **Ctrl+Down** (or **Command+Up** or **Command+Down** if you're on a MacOS) to automatically adjust attention to selected text (code contributed by anonymous user) - Loopback, run img2img processing multiple times - X/Y/Z plot, a way to draw a 3 dimensional plot of images with different parameters - Textual Inversion - have as many embeddings as you want and use any names you like for them - use multiple embeddings with different numbers of

vectors per token - works with half precision floating point numbers - train embeddings on 8GB (also reports of 6GB working) - Extras tab with: - GFPGAN, neural network that fixes faces - CodeFormer, face restoration tool as an alternative to GFPGAN - RealESRGAN, neural network upscaler - ESRGAN, neural network upscaler with a lot of third party models - SwinIR and Swin2SR (see here), neural network upscalers - LDSR, Latent diffusion super resolution upscaling - Resizing aspect ratio options - Sampling method selection - Adjust sampler eta values (noise multiplier) - More advanced noise setting options - Interrupt processing at any time - 4GB video card support (also reports of 2GB working) - Correct seeds for batches - Live prompt token length validation - Generation parameters - parameters you used to generate images are saved with that image - in PNG chunks for PNG, in EXIF for JPEG - can drag the image to PNG info tab to restore generation parameters and automatically copy them into UI - can be disabled in settings - drag and drop an image/text-parameters to promptbox - Read Generation Parameters Button, loads parameters in promptbox to UI - Settings page - Running arbitrary python code from UI (must run with `--allow-code` to enable) - Mouseover hints for most UI elements - Possible to change defaults/mix/max/step values for UI elements via text config - Tiling support, a checkbox to create images that can be tiled like textures - Progress bar and live image generation preview - Can use a separate neural network to produce previews with almost none VRAM or compute requirement - Negative prompt, an extra text field that allows you to list what you don't want to see in generated image - Styles, a way to save part of prompt and easily apply them via dropdown later - Variations, a way to generate same image but with tiny differences - Seed resizing, a way to generate same image but at slightly different resolution - CLIP interrogator, a button that tries to guess prompt from an image - Prompt Editing, a way to change prompt mid-generation, say to start making a watermelon and switch to anime girl midway - Batch Processing, process a group of files using img2img - Img2img Alternative, reverse Euler method of cross attention control - Highres Fix, a convenience option to produce high resolution pictures in one click without usual distortions - Reloading checkpoints on the fly - Checkpoint Merger, a tab that allows you to merge up to 3 checkpoints into one - Custom scripts with many extensions from community - Composable-Diffusion, a way to use multiple prompts at once - separate prompts using uppercase **AND** - also supports weights for prompts: `a cat :1.2 AND a dog AND a penguin :2.2` - No token limit for prompts (original stable diffusion lets you use up to 75 tokens) - DeepDanbooru integration, creates danbooru style tags for anime prompts - xformers, major speed increase for select cards: (add `--xformers` to commandline args) - via extension: History tab: view, direct and delete images conveniently within the UI - Generate forever option - Training tab - hypernetworks and embeddings options - Preprocessing images: cropping, mirroring, autotagging using BLIP or deepdanbooru (for anime) - Clip skip - Hypernetworks - Loras (same as Hypernetworks but more pretty) - A separate UI where you can choose, with preview, which embeddings, hypernetworks or Loras to add to your prompt - Can select to load a different VAE from settings screen - Estimated completion time in progress bar - API - Support for dedicated inpainting model by RunwayML - via extension: Aesthetic Gradients, a way to generate images with a specific aesthetic by using clip images embeds (implementation of <https://github.com/vicgalle/stable-diffusion-aesthetic-gradients>) -

Stable Diffusion 2.0 support - see wiki for instructions - Alt-Diffusion support - see wiki for instructions
- Now without any bad letters! - Load checkpoints in safetensors format - Eased resolution restriction:
generated image's dimensions must be a multiple of 8 rather than 64 - Now with a license! - Reorder
elements in the UI from settings screen - Segmind Stable Diffusion support

Installation and Running

Make sure the required dependencies are met and follow the instructions available for: - NVidia (recommended) - AMD GPUs. - Intel CPUs, Intel GPUs (both integrated and discrete) (external wiki page) - Ascend NPU (external wiki page)

Alternatively, use online services (like Google Colab):

- List of Online Services

Installation on Windows 10/11 with NVidia-GPUs using release package

1. Download [sd.webui.zip](#) from v1.0.0-pre and extract its contents.
2. Run [update.bat](#).
3. Run [run.bat](#). > For more details see Install-and-Run-on-NVidia-GPUs

Automatic Installation on Windows

1. Install Python 3.10.6 (Newer version of Python does not support torch), checking "Add Python to PATH".
2. Install git.
3. Download the stable-diffusion-webui repository, for example by running `git clone https://github.com/AUTOMATIC1111/stable-diffusion-webui.git`.
4. Run [webui-user.bat](#) from Windows Explorer as normal, non-administrator, user.

Automatic Installation on Linux

1. Install the dependencies:

```
1 # Debian-based:
2 sudo apt install wget git python3 python3-venv libgl1 libglvnd
3 # Red Hat-based:
4 sudo dnf install wget git python3 gperftools-libs libglvnd-glx
5 # openSUSE-based:
6 sudo zypper install wget git python3 libtcmalloc4 libglvnd
```

```
7 # Arch-based:
8 sudo pacman -S wget git python3
```

2. Navigate to the directory you would like the webui to be installed and execute the following command:

```
1 wget -q https://raw.githubusercontent.com/AUTOMATIC1111/stable-
diffusion-webui/master/webui.sh
```

3. Run `webui.sh`.
4. Check `webui-user.sh` for options. ### Installation on Apple Silicon

Find the instructions here.

Contributing

Here's how to add code to this repo: Contributing

Documentation

The documentation was moved from this README over to the project's wiki.

For the purposes of getting Google and other search engines to crawl the wiki, here's a link to the (not for humans) crawlable wiki.

Credits

Licenses for borrowed code can be found in [Settings](#) -> [Licenses](#) screen, and also in [html/licenses.html](#) file.

- Stable Diffusion - <https://github.com/Stability-AI/stablediffusion>, <https://github.com/CompVis/taming-transformers>
- k-diffusion - <https://github.com/crowsonkb/k-diffusion.git>
- Spandrel - <https://github.com/chaiNNer-org/spandrel> implementing
 - GFPGAN - <https://github.com/TencentARC/GFPGAN.git>
 - CodeFormer - <https://github.com/sczhou/CodeFormer>
 - ESRGAN - <https://github.com/xinntao/ESRGAN>
 - SwinIR - <https://github.com/JingyunLiang/SwinIR>
 - Swin2SR - <https://github.com/mv-lab/swin2sr>

-
- LDSR - <https://github.com/Hafiidz/latent-diffusion>
 - MiDaS - <https://github.com/isl-org/MiDaS>
 - Ideas for optimizations - <https://github.com/basujindal/stable-diffusion>
 - Cross Attention layer optimization - Doggettx - <https://github.com/Doggettx/stable-diffusion>, original idea for prompt editing.
 - Cross Attention layer optimization - InvokeAI, lstein - <https://github.com/invoke-ai/InvokeAI> (originally <http://github.com/lstein/stable-diffusion>)
 - Sub-quadratic Cross Attention layer optimization - Alex Birch (<https://github.com/Birch-san/diffusers/pull/1>), Amin Rezaei (<https://github.com/AminRezaei0x443/memory-efficient-attention>)
 - Textual Inversion - Rinon Gal - https://github.com/rinongal/textual_inversion (we're not using his code, but we are using his ideas).
 - Idea for SD upscale - <https://github.com/jquesnelle/txt2imghd>
 - Noise generation for outpainting mk2 - <https://github.com/parlance-zz/g-diffuser-bot>
 - CLIP interrogator idea and borrowing some code - <https://github.com/pharmapsychotic/clip-interrogator>
 - Idea for Composable Diffusion - <https://github.com/energy-based-model/Compositional-Visual-Generation-with-Composable-Diffusion-Models-PyTorch>
 - xformers - <https://github.com/facebookresearch/xformers>
 - DeepDanbooru - interrogator for anime diffusers <https://github.com/KichangKim/DeepDanbooru>
 - Sampling in float32 precision from a float16 UNet - marunine for the idea, Birch-san for the example Diffusers implementation (<https://github.com/Birch-san/diffusers-play/tree/92feee6>)
 - Instruct pix2pix - Tim Brooks (star), Aleksander Holynski (star), Alexei A. Efros (no star) - <https://github.com/timothybrooks/instruct-pix2pix>
 - Security advice - RyotaK
 - UniPC sampler - Wenliang Zhao - <https://github.com/wl-zhao/UniPC>
 - TAESD - Ollin Boer Bohan - <https://github.com/madebyollin/taesd>
 - LyCORIS - KohakuBlueleaf
 - Restart sampling - lambertae - https://github.com/Newbeeer/diffusion_restart_sampling
 - Hypertile - tfernd - <https://github.com/tfernd/HyperTile>
 - Initial Gradio script - posted on 4chan by an Anonymous user. Thank you Anonymous user.
 - (You)