JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

HOWTO: Make Synthetic Faces with Stable Diffusion

January 2023

J.R. Parsons
Principal Professional Staff
Cyber Warfare Systems Group (AOS/QCR)

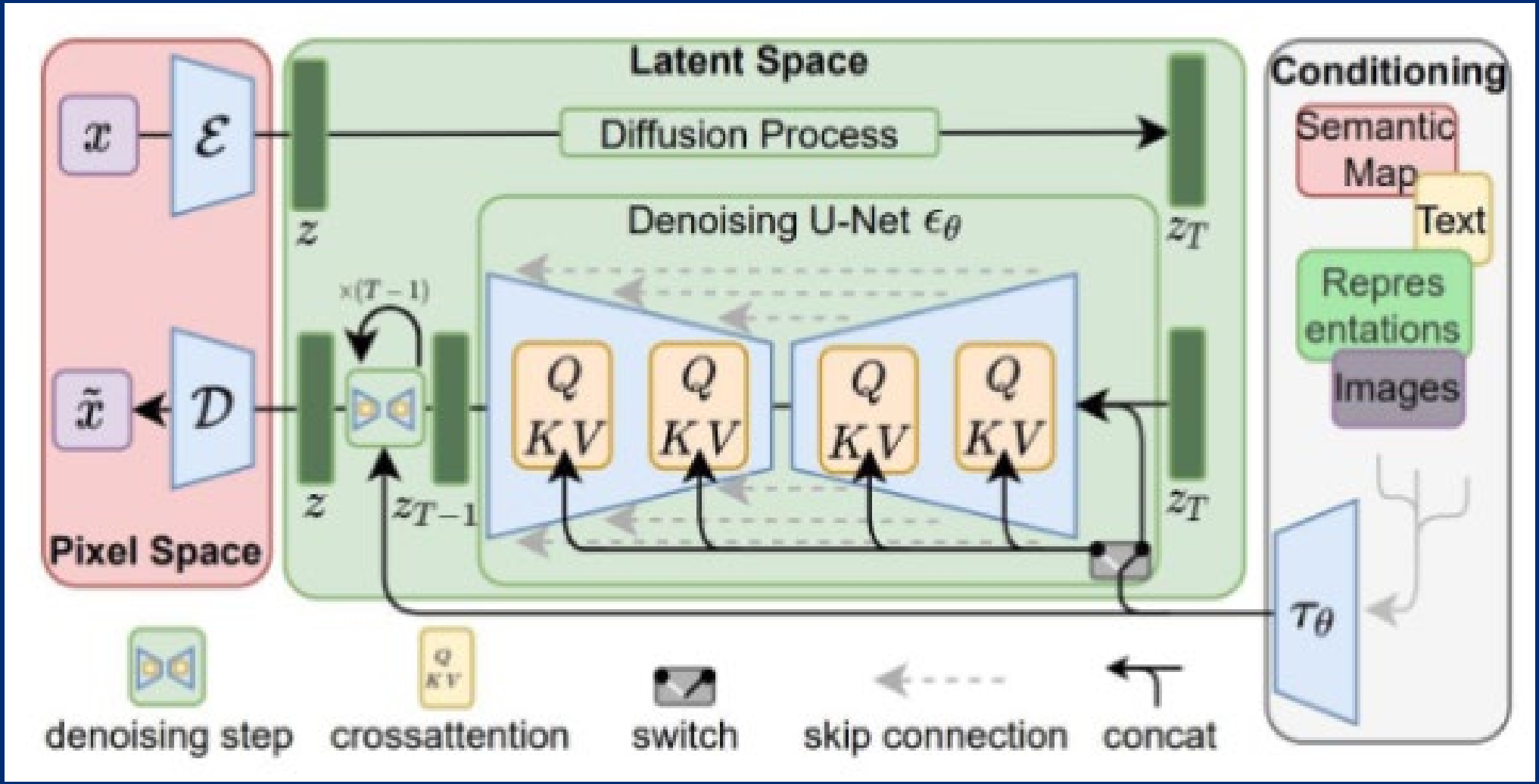
Stable Diffusion Overview

- Hardware & Software Prerequisites
- How Stable Diffusion Works
- Prompt Engineering
- Using **`txt2img`** mode
- Using **`img2img`** mode
- Other Settings
- Training Custom Embeddings
- Putting it all together: “Chimeras”

Prerequisites

- **The right computer** – Windows, Mac, or Linux, with a powerful Graphics Processing Unit (GPU) that has at least 10+ GB of VRAM
  - Any modern server-class NVIDIA card is probably adequate
  - NVIDIA RTX3060 12GB model is available for ~$400, compatible with many desktops, but may require a higher-wattage power supply
  - Non-NVIDIA cards can be challenging because of hardware support
- **Install Stable Diffusion** – this walkthrough uses the Automatic1111 Web GUI (https://github.com/automatic1111/stable-diffusion-webgui)
  - The **Image Browser** extension is optional but will let you more easily look at your results
  - The **xformers** library is optional (and sometimes painful to install) but provides a 2x-4x speedup
- **Get models -** at least one “checkpoint” model like (https://huggingface.co/runwayml/stable-diffusion-v1-5)
  - Many prompting guides assume v1.5
  - v2.0 and v2.1 require slightly different approaches
- **(Optional) images -** for reproducing an existing person, 25+ high-quality images of that person (more is better)

How It Works


Latent Space
Diffusion Process
Denoising U-Net ϵ_θ
Conditioning
Semantic Map
Text
Repres entations
Images
Pixel Space
x
$\mathcal{E}$
$\tilde{x}$
$\mathcal{D}$
z
z_T
z_{T-1}
$\times(T-1)$
Q
K V
τ_θ
denoising step
crossattention
switch
skip connection
concat

How It Works

Training

- **Variational Autoencoder** (VAE) compresses an image (or image/text pair) to a very dense representation of the content of the image (“latent representation”)
- The latent representation has noise added to it, and a **U-Net** is trained to invert the noise-addition step, conditioned on the text inputs.
- Periodically in its training, a copy of the U-Net is saved and its performance measured. U-Net files in the Models directory are called “checkpoints” for this reason.

Inference

- Once trained, the back half of the system is run without benefit of a starting image.
- Starting from pure random noise, the U-Net is tasked with identifying a mathematical operation best suited for “subtracting the noise”, using the text as its only guidance.
- That operation is run against the noise for 10+ computation steps, until only a denoised latent vector is present.
- The VAE decodes the latent representation back to an image for viewing by humans.

Prompt Engineering with CLIP

- **Learn how to ask for what the U-Net was trained on**
  - Dataset was a set of 5B web pages
  - Because the goal was training image/text pairs, these were picture-rich websites: Pinterest, DeviantArt, Wordpress, Tumblr, etc.
- Photography terms will recall stock photography websites
  - **`Sharp focus, Medium shot, Tilt shift macro photography`**
  - **`Nikon f2.8`** will capture the aesthetic of photos tagged with that term
- Artistic terms will recall art websites
  - **`8K, 16K, oil painting, charcoal sketch`**
  - Even the phrase "**`trending on Artsta`**tion" can drive up quality!
- Names of artists will recall that artist's specific style
- Names of websites will recall a photo style that's typical there
  - Facebook, Instagram, Behance, …

Text to Image Example

Prompt: Tilt shift macro photography, verdigris ancient copper coin on a wooden table, very detailed, intricate, sharp focus

Negative Prompt: blurry, grainy, smeared, waxy, plastic, fake

Text to Image Example

Prompt: Medieval stained glass window, photorealistic, Saint Taylor Swift, art deco inspired by Alphonse Mucha, vibrant HDR colors

Negative Prompt: blurry, grainy, smeared, cartoon, anime, painting, drawing

Text to Image Example

Prompt: Anime (porcupine) warrior [wearing steel armor], holding a broadsword, fantasy character art in the style of Studio Ghibli

Negative Prompt: human face, photograph, realistic, 3D, smeared, mutated, deformed, asymmetrical

(parentheses) emphasize a term,
[brackets] de-emphasize a term

Text to Image – Language Matters

Photography portrait, “魅力的な男性会社員”, clear and detailed face

Photography portrait, “ عامل مكتب الذكور جذابة”, clear and detailed face

Text to Image – One Subject at a Time

- The model tries to make the whole image match the prompt

- For “Chiefs vs. Cardinals”, the artist tried to get Master Chief from Halo fighting a giant Cardinals Mascot on a grassy field
  - Master Chief’s armor is red, and has pseudo-eyes and a pseudo-beak
  - The Cardinal mascot seems to have upgraded his suit to light ballistic armor

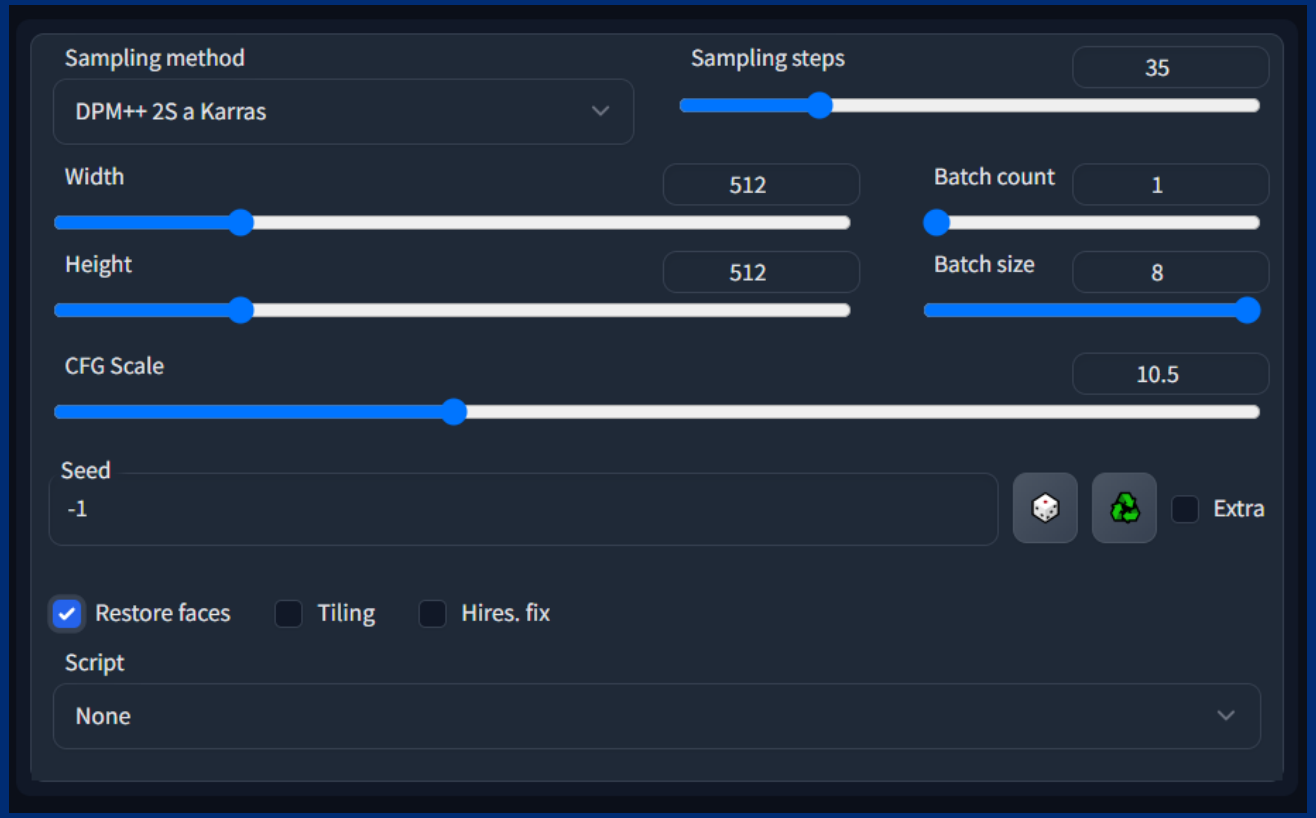
- Prompts with two distinct subjects risk the model trying to create a hybrid of the two
  - Better to use img2img to do this piecemeal
  - We will actually use this flaw to our advantage later on!

What Do the Settings Do?

- **Sampler** is which math equation(s) are used to drive the diffusion process
  - The DPM++ family are fast and efficient
- **Context-Free Guidance Scale** (CFG Scale) governs the rigidity of the prompts
  - Typically keep this between 6 and 12
- **Steps** is how many denoising steps are used
  - 15 for checking coarse composition and “vibes”
  - 25-35 is fine for most things
  - 50+ for some high-realism scenes or final draft
- **Scripts** include ways to adjust settings during a run – this grid was generated with the **“X/Y Plot”** script

Recommended txt2img Configuration

- **DPM++ 2S a Karras** moves through latent space faster than **Euler** and other samplers
- **35 Sampling steps** gets decent results
  - 20 steps “rough draft” for trying prompts
  - 50+ steps for “final draft” when your prompt is perfected
- **Batch Size = 8** for testing prompts
- **Batch Count = 1** unless you’re “seed farming” to find a perfect composition
- **CFG Scale = 10.5** (higher with long complex prompts)
- **Restore Faces** to clean up glitches
  - May have side effects!



Try playing with all of these!

Image to Image Example

Prompt: Tilt shift macro photograph of a silver coin on a wooden table, coin embossed with a woman's face, profile view, side view, realistic, intricate, highly detailed

Negative Prompt: blurry, grainy, smudged, smeared, manly, golden, beard

CFG Scale: 6.0

Steps: 40

Denoising strength: 0.7 (70%)

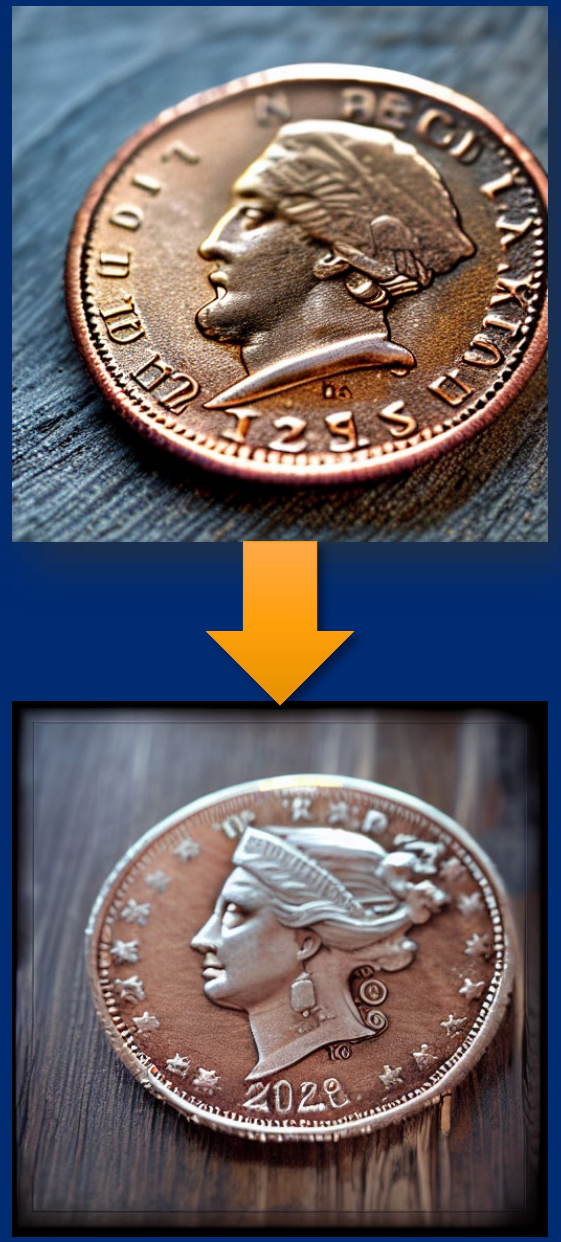
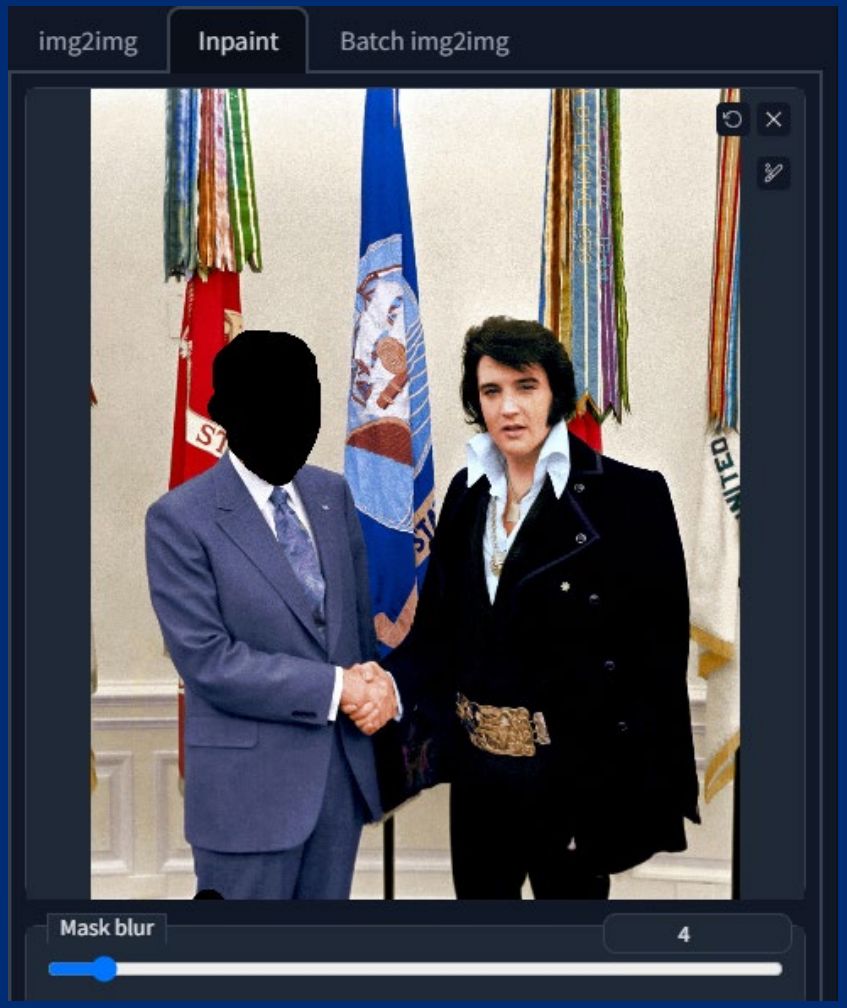


Image to Image Inpainting Example



Prompt: Grainy [[blurry]] 1970s (photograph) of George Clooney, realistic, flat matte colors, Kodachrome, colorized, flat diffuse lighting, low resolution

Negative Prompt: painting, illustration, rendered, fake, waxy, shiny, 3D, sharp, HD, 16K

CFG Scale: 10.5

Steps: 40

Denoising strength: 0.8 (80%)

Matching the Mood of the Image is Hard!

Prompt: Grainy [[blurry]] 1970s (photograph) of George Clooney, realistic, flat matte colors, Kodachrome, colorized, flat diffuse lighting, low resolution

Negative Prompt: painting, illustration, rendered, fake, waxy, shiny, 3D, sharp, HD, 16K

CFG Scale: 10.5

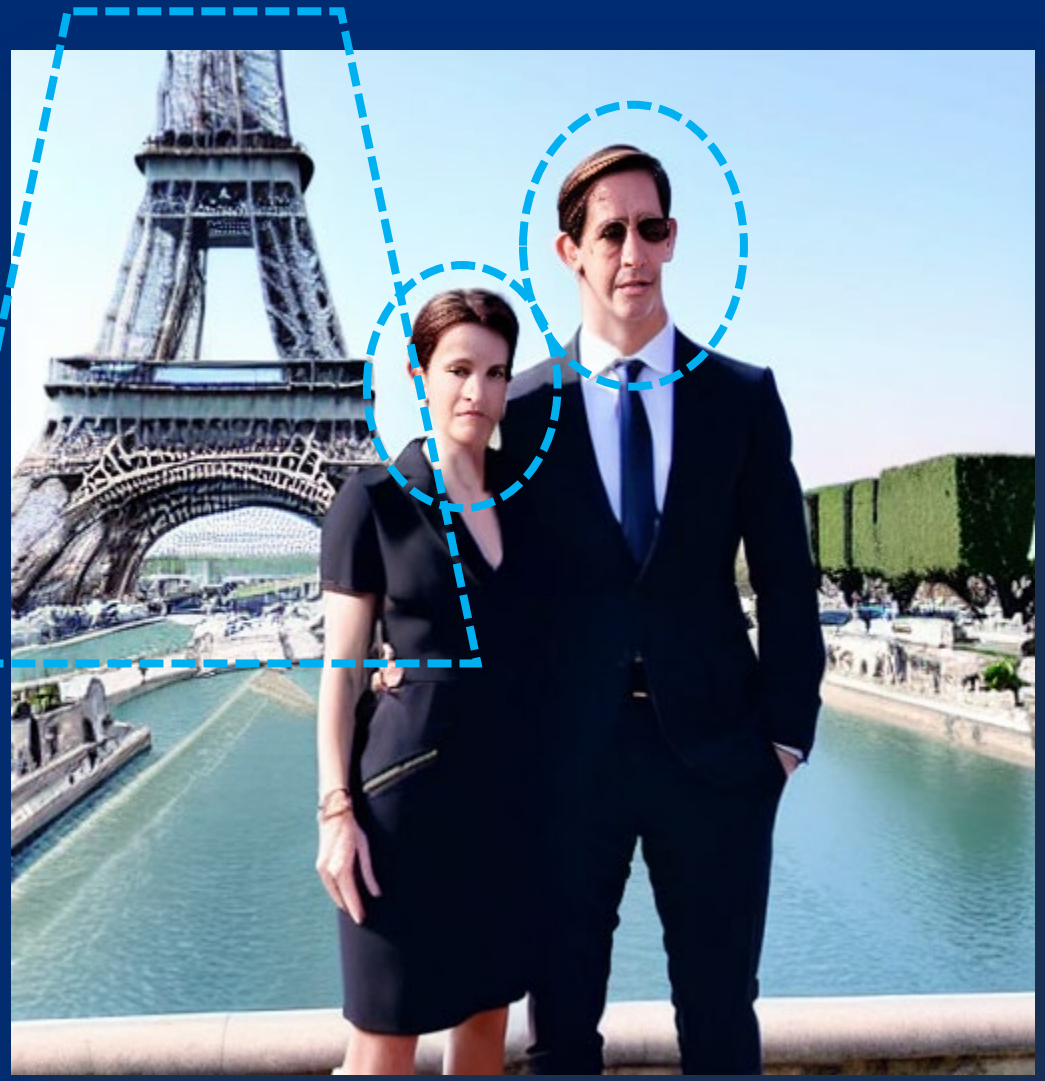
Steps: 40

Denoising strength: 0.8 (80%)

What Do the Settings Do?

- **Draw** or **Upload Mask** changes where the mask (blacked-out area) comes from
- **Inpaint Masked** or **Not Masked** changes whether masking an area means “change this” or “change everything but this”
- **Inpaint at Full Resolution:** expand the masked-out area to the selected size (usually 512x512) and apply the prompt just to that area, then patch it back in
  - Great for replacing details like faces or hands
- **Fill/Original/Latent Noise/Latent Nothing –** what does Stable Diffusion start with when it tries to fill in the masked area?
  - **Fill:** a solid black area
  - **Original**: the original image
  - **Latent Noise:** noisy static from the latent space
  - **Latent Nothing:** neutral uniform data

Image to Image Example: Fixing the Eiffel Tower



Mask blur 4

Mask mode
Inpaint masked | Inpaint not masked

Masked content
fill | original | latent noise | latent nothing

Inpaint area
Whole picture | Only masked

Only masked padding, pixels 16

Resize mode
Just resize | Crop and resize | Resize and fill | Just resize (latent upscale)

Sampling method
DPM++ 2S a Karras

Sampling steps 80

Width 512

Batch count 1

Height 512

Batch size 8

CFG Scale 11.5

Denoising strength 0.5

Seed -1

Extra

Image to Image Example: Fixing the Eiffel Tower

- Generated original with txt2img
- Three img2img inpainting passes:
  - Man's face
  - Woman's face
  - Eiffel Tower
- Saved best result after each pass and then inpainted that result
- For mild changes like this, leave masked content with **original**
- For drastic changes, try **latent nothing**
- If you're not sure where to start, use Scripts > X/Y Plot to find the best balance of denoising, steps, and CFG

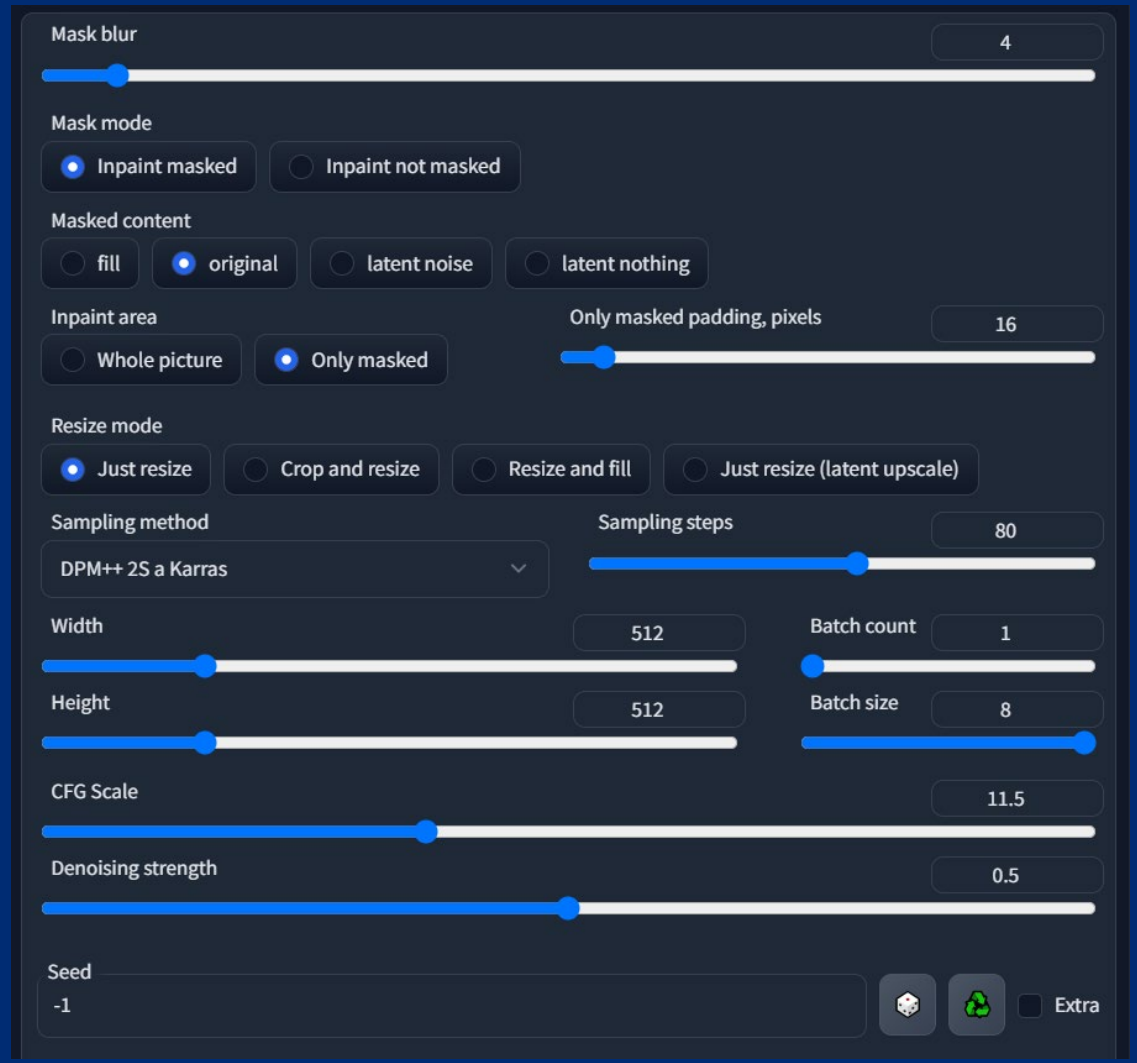
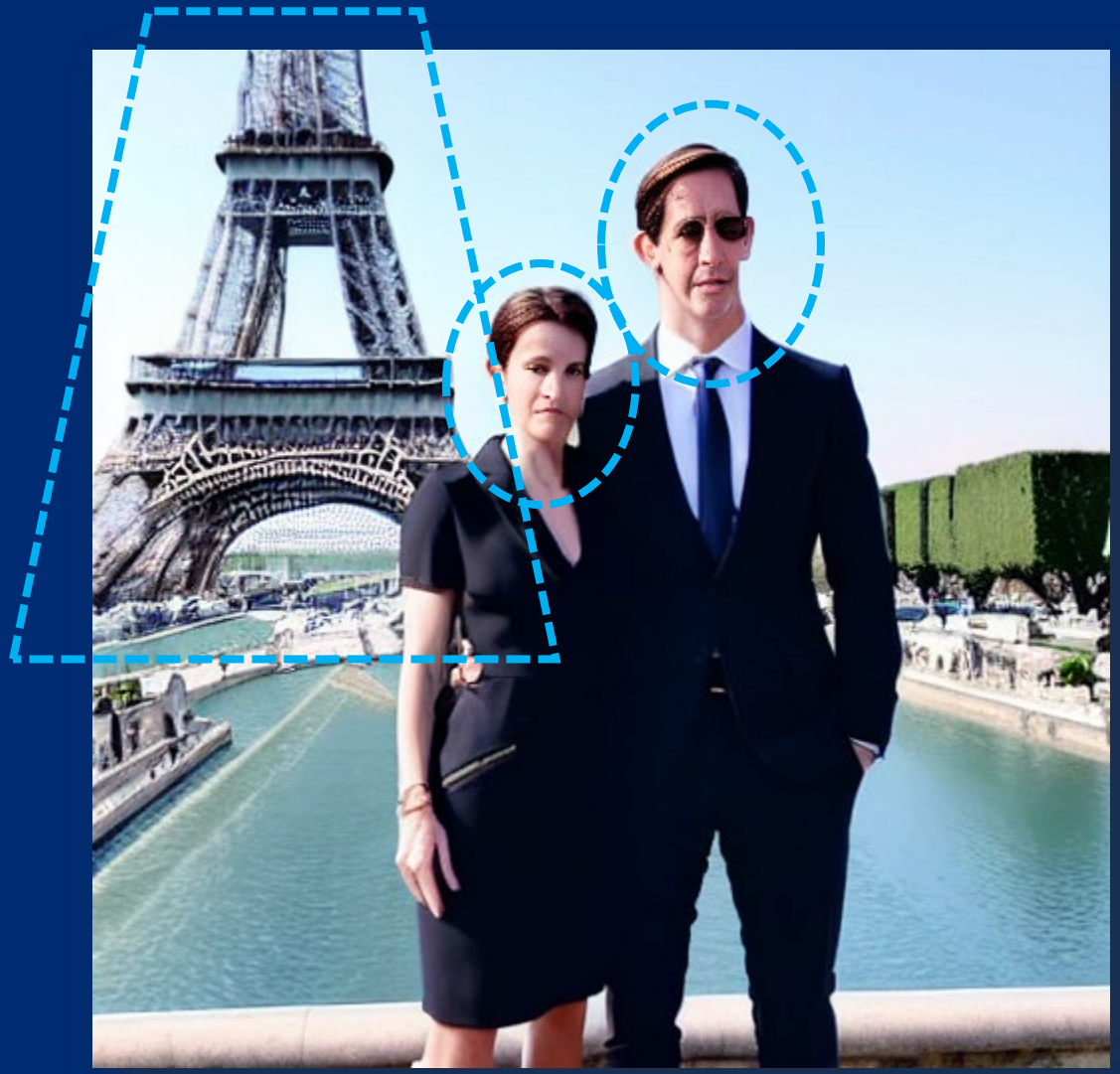


Image to Image Example: Fixing the Eiffel Tower



Training Custom Embeddings

- **“Embedding”** is a keyword that captures a concept
  - Person, art style, pose, or even a flaw

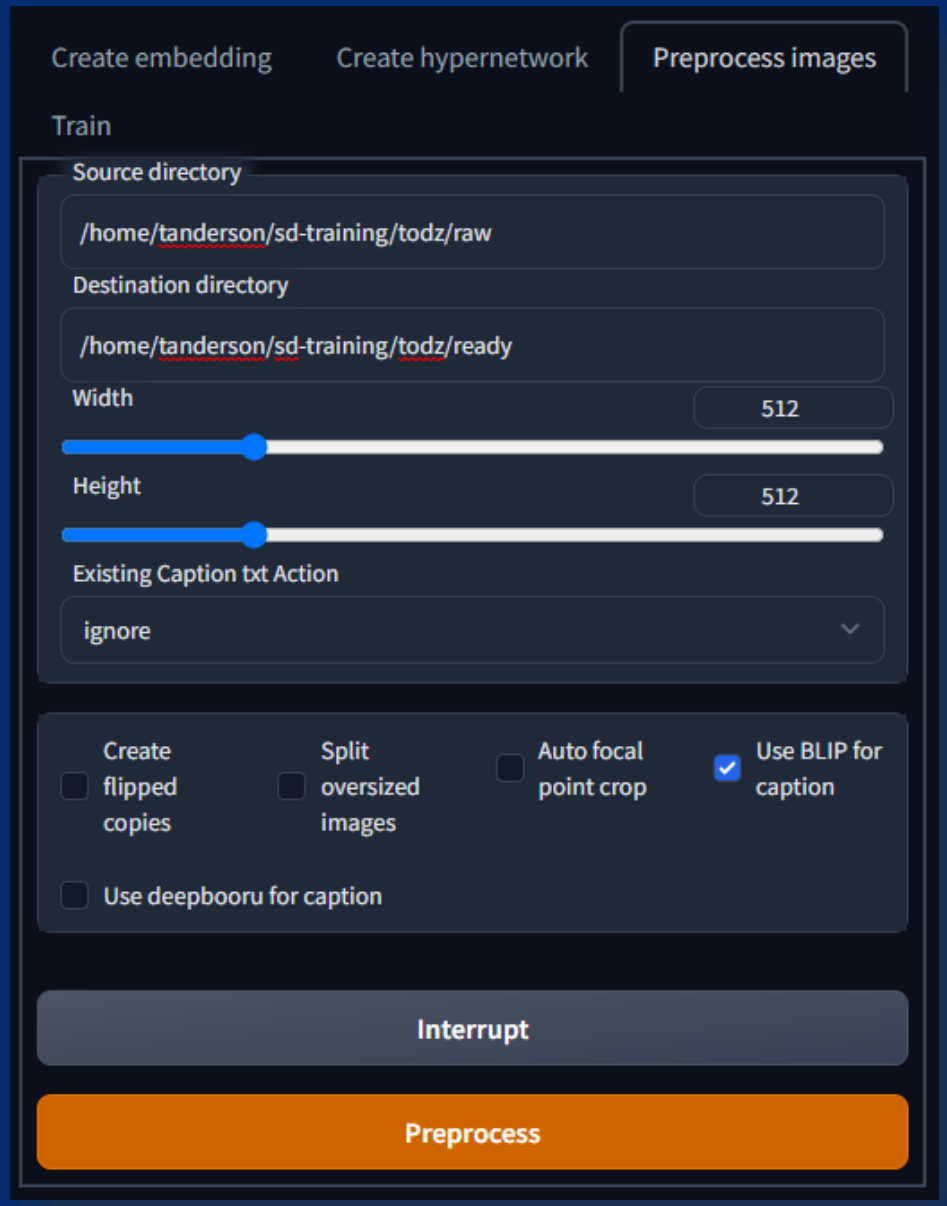
- Choose a nickname for your embedding
  - Not a plain English word
- Acquire 25+ images of the subject
  - Some people report moderate success with as few as 5
  - More diverse images = more flexible/robust embedding
- Crop to 512x512
- Describe each image with a prompt
- Set the learning parameters
- Train!
  - Go get some coffee and wait

Creating the Empty Embedding

- Within the WebUI, select **Train** and within the Train menu, **Create embedding**.
  - **Name:** pick a unique token (nickname) for this concept that isn't common in the training data
    - BAD: "dave" – there is a lot of "Dave" on the internet
    - BETTER: "dave10"
    - BEST: "67f0b63f" – good for the model, not very memorable for humans
  - **Initialization text:** what sort of thing are you training? Pick a very short term for what it is
    - GOOD: person, dog
    - BETTER: man, face, greyhound
  - **Number of Vectors per token** should be 4-10 for a specific human face
    - Use a larger number if you anticipate using big complex prompts
  - …once you're sure you have what you need select **Create Embedding**

Preprocessing Your Images

- Set up **source** and **destination** directories for the images
  - These images should be stored on the same filesystem that Stable Diffusion is running on, not a network share
- Leave **width** and **height** at 512x512
- **Create flipped copies** if left/right differences don't matter – this can double the size of your dataset, but doesn't add much diversity
- **Use BLIP for caption** will try to auto-caption your images
  - Useful if you have 10+ images you need to caption
  - But for now (early 2023) BLIP is not as good at captioning as humans are
  - Makes mistakes like "with a cell phone in her hand" or "wearing a coat and a coat and a coat"
  - Manually correct your BLIP captions!
- **Auto focal point crop** can make square crops that emphasizes faces
- When you're ready, click **Preprocess**



Example Captions for Training

- A good template is very descriptive!
  - **Kind of photo:** selfie, medium shot, close up face, head and shoulders portrait, side view
  - **Subject:** man, person, face, professional man, tall man, older man, woman, etc.
  - **Feature:** gray hair, glasses
  - **Clothing:** a patterned shirt, business casual clothing
  - **Light:** diffuse light, sunset light, very bright artificial light, fluorescent light, dark shadows, etc.

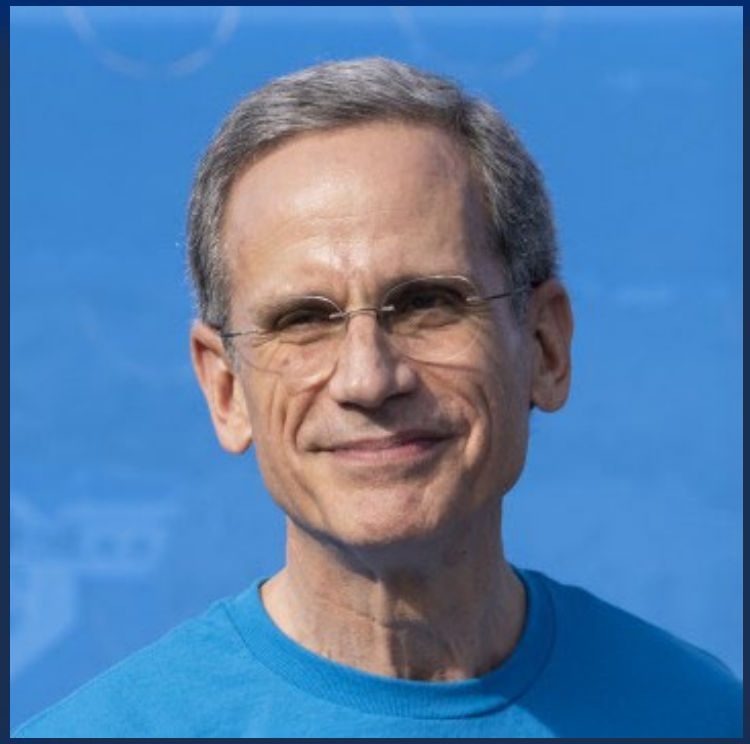
"[KIND OF PHOTO] of a [SUBJECT] with [FEATURE] wearing [CLOTHING] and [CLOTHING] and [CLOTHING] [DOING WHAT] [WHERE] in [KIND OF LIGHT]"

- Describe every detail in the image *except* what you want it to learn
  - You're teaching the model to subtract all the things it already knows so it can learn the thing it doesn't know

- If you're trying to learn a person's face…
  - DO describe their outfits, the background, the kind of lighting, their pose
  - DO NOT describe their natural hair color or their natural skin tone

Example Captions for Training (cont'd.)

medium shot of a man wearing a checkered shirt and blue blazer in bright indoor lighting



face portrait of man wearing a blue t shirt standing in front of a blue wall in very bright sunlight

a side view profile of a man gesturing with his hands standing outdoors with natural sunlight

Training

- Select the **embedding** you created above
- For **learning rate**, the default is fine
  - You can also enter a “learning plan” like

    `0.005:100, 1e-3:1000, 1e-5`

    …which means “Learn at 0.005 for 100 steps, then 0.0001 until the 1000th step, then 0.000001 for the rest of the run.” This will take longer but avoids overfitting.
- **Batch size** makes training more effective but requires more VRAM
  - Without xformers installed your batch size will be in the low single digits
  - With xformers batch size can be in the 20s or even 40s (Settings > Training > Enable Cross-Attention Optimizations During Training)
- **Gradient Accumulation Steps:**
  - If batch size is smaller than your total number of images, set Gradient Accumulation Steps so that

    Batch Size x Gradient Accumulation Steps = Total Images

    This substantially slows down training, so pick the largest batch size you can
- **Save images with embedding in PNG chunks**: leave this checked for now. It stores the new embedding within the images it generates.

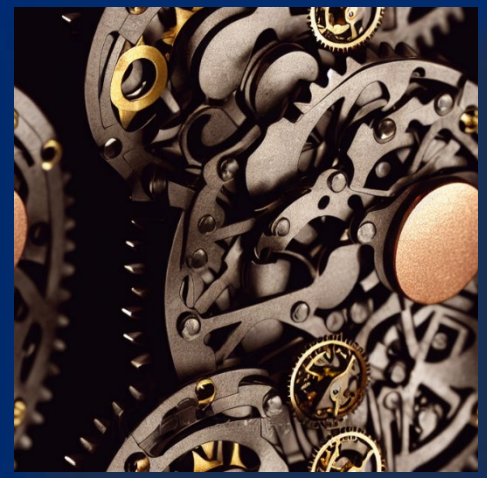
Training (cont'd.)

- **Prompt template** is a recipe for generating prompts during training. For every image, the training system will use a randomly selected row from the prompt template file, and make two replacements:
  - **[name]** will be replaced with the token that you're trying to train the system to recognize, e.g. "$dave004"
  - **[filewords]** will be replaced with the prompt in the filename
- Create a text file with several rows that include phrases like
  - `A photograph of [name], [filewords]`
  - `A picture of [name], [filewords]`
  - `A photo of [filewords], (([name]))`
- These will be randomly assigned to your images, so don't try to tailor them or match them up to individual files

Training (cont'd.)

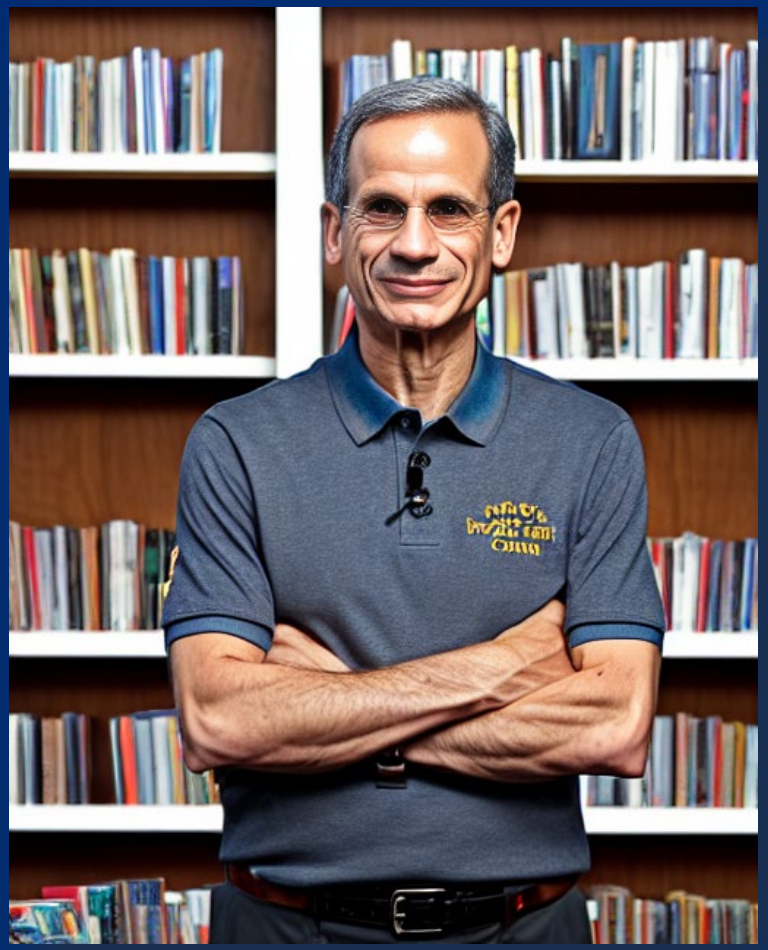
- Once you are certain you've specified the right prompt template file, set your learning rate and batch size correctly, click **Train Embedding**
- Progress images will be generated every ~500 steps
  - Set progress outputs to 50 steps if you want to closely monitor the training
  - These will look freaky and horrible for a long time!
  - They are generated with only 20 steps
- You will see the "training loss" drop to about 0.1 and then stabilize there, and that's okay
- You don't have to watch it train… it will run, unattended, for hours



The Horror, The Horror

...Success!



Prompt: A portrait of ralph006, business casual clothing, **((red))** (polo shirt) , exceptional detail, intricate, library bookshelves, sharp focus, indoor photography, natural sunlight

Negative prompt: blurry, grainy, smeared, ugly, monstrous, deformed, mutant, waxy, plastic, fake, rubber mask

Steps: 100

CFG scale: 11.5

Size: 512x640

Real image for comparison

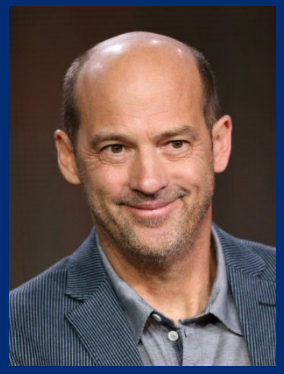
Putting it all Together - Prelude

- We're still learning to make **repeatable** faces
  - …using the Cardinal/Chief approach!
- Blending well-known people works, but with some caveats:
  - Superstars are "too powerful". 10,000+ pictures in the dataset means their facial features will always dominate. Taylor Swift, George Clooney, Barack Obama, Demi Moore… forget it!
  - People with very distinctive or quirky facial features can be hit or miss. Anya Taylor-Joy, Steve Buscemi, Jim Carrey, Bjork – using **one** of these can be a useful anchor, but using several can be disastrous
- Current best bet is mix and match B- and C-list celebrities until you find a recipe that produces the same face reliably
- If you generate a batch and one of the faces resembles a different celebrity, lean into it – try adding their name to the mix

Example: Making a Chimera

- Model, 2 actors, British politician
- Only 2 useful results in first batch
  - And not consistent!






Detailed (head and shoulders) portrait photograph of an economics professor, realistic and domestic, candid photograph, casual clothing, dramatic diffuse lighting, sunrise, [Rory Stewart, David Gandy, Jason Dohring, Anthony Edwards], رجل وسيم

CGI, rendered, group photo, several men, monochrome, black and white, grayscale, blurry, grainy, mutant, weird, deformed, cartoon, ((fake)), wax, ugly,


Making a Chimera, Part 2

- Consider using **X/Y Prompt** feature once you've got the concept correct
- Tune for "best execution" trying different values for CFG and Steps
- Trial and error, unfortunately

Making a Chimera, Part 3

- **Lots** of trial and error! Keep making small batches until you get a batch that's internally consistent
- If you keep seeing a celeb that it resembles, consider adding them in
  - Tried adding Ethan Hawke and Justin Tucker after this run…
  - Rory Stewart is either too distinctive or too famous (or both!)
  - Keep iterating…!

Making a Chimera – Final Prompt

These prompts are the recipe I used to make “todz” training data:

- **Prompt:** Detailed (head and shoulders) portrait photograph of a [chubby] economics professor, realistic and domestic, candid photograph, casual clothing, buzz cut, dramatic diffuse lighting, sunrise, Jason Dohring, Justin Tucker, [[Rory Stewart]], Josh Brolin, ruggedly handsome
- **Negative:** ((goatee)), beard, CGI, rendered, group photo, several men, monochrome, black and white, grayscale, blurry, grainy, mutant, weird, deformed, cartoon, ((fake)), wax, ugly, emaciated, obese

Making a Chimera – “Quick” Results

Making a Chimera - Training

- Once you can reliably generate 20+ images of a **similar-enough** person, you can train that person as a token
  - This means you can generate that person repeatedly on-demand in a variety of settings
  - To support training, keep the names and change up the style
    - Instagram selfie
    - Passport photo / mugshot
    - Lots of distinctive outfits: coat and tie, Hawaiian shirt, sports jersey, fuzzy sweater
  - Push hard for diverse images here, or your training will get stuck on the similarities
  - Still investigating if we benefit from filtering this through 68-landmark FR tooling

- We chose **todz** for this chimera (those four letters are “rare” in the training data)

- Getting a “generic white guy” is pretty easy
  - Young white women are also very easy to generate
  - Non-white people are more challenging but not impossible
  - May take more effort to get to a reliable generation prompt

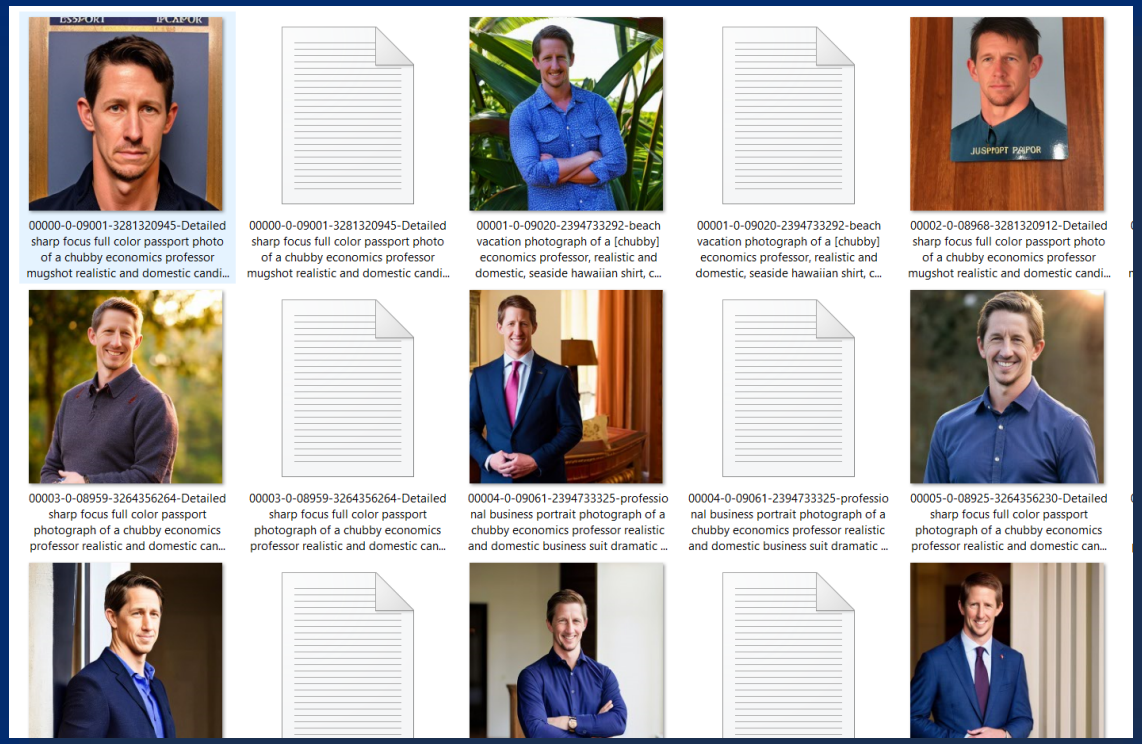
Interlude: Extensions

- The AUTOMATIC1111 GUI offers flexibility through **extensions**
  - **Image Browser** is a must-have
  - **Wildcards** is outstanding for generating diverse versions of a chimera
  - New extensions are released often
- Wildcards:
  - Put **`thing.txt`** in **`/extensions/stable-diffusion-webui-wildcards/wildcards/`**
  - Use **`__thing__`** in your prompts
  - Each image you generate will have a random **thing** in it
  - **“`wearing __outfit__`”**
  - **“`with a __emotion__ expression on his face`”**

```
 GNU nano 4.8  expression.txt
serious
happy
grieving
thoughtful
puzzled
betrayed
shocked
angry
furious
gleeful
joyous
seductive
beatific
peaceful
relaxed
```

```
 GNU nano 4.8     setting.txt
indoor library bookshelves
indoors in an English pub
indoor bedroom, mahogany furniture
indoor medieval basement, stone walls
indoor intricate grocery store shelves
indoor fireplace warm cozy room
outdoor forest clearing trees
outdoor beach seaside coast
outdoor mountains
outdoor wooden gazebo
outdoor crowded urban city streets
outdoor stadium crowds
```

“TODZ” Results



25 training images
curated from ~20+ prompts
trial and error (for now)

Generated “todz” pictures

Meet “Todd Z.”, a Stable Diffusion Chimera

Selfie outside the Louvre

((Instagram)) vacation selfie, beach photo

Selfie outside Bagram Afghanistan, Army uniform

Security

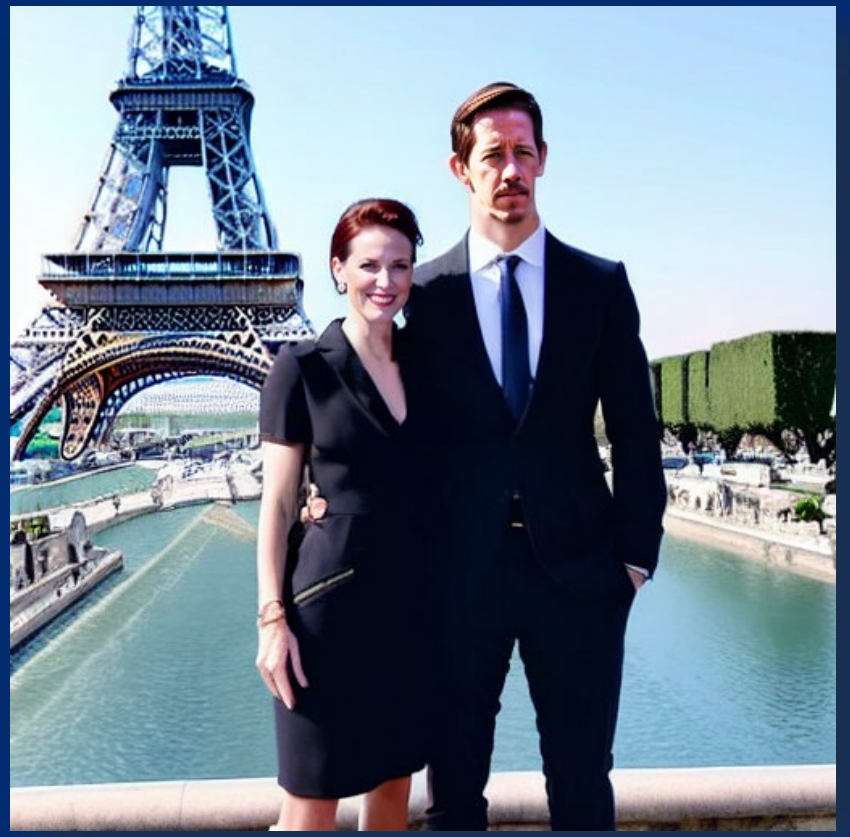
- Be careful with Extensions
  - By default the **`--listen`** mode locks down the ability to install extensions from the GUI
  - …because that would lead to arbitrary remote code execution
  - (So don't disable it unless you're confident about your boundary/sharing)

- Only use model files from trusted sources
  - Model checkpoints (CKPT) are **Python Pickle** objects
  - Vulnerable to deserialization attacks & silent arbitrary code execution
  - …unless opened safely, which AUTOMATIC1111 does!
  - …but other attacks may be possible.

- In their raw format PNGs from Stable Diffusion are very detectable
  - Settings > Saving Images/Grids > **Do not add watermark to images**
  - Settings > Saving Images/Grids > **Save text information about generation parameters as chunks…**
  - Or, leave these on, and crop/re-save/edit to obliterate original file information

Next Steps (Research Questions)

- If we use 68-landmark facial recognition to curate the training data, will our chimera retain that identity?
  - Can we generate a large volume of **todz** images and curate or “distill” him to a chosen identity?
- Can we use image-to-video techniques (FOMM, etc.) to take a ***single image*** and build a working chimera from one photo?
- What techniques and tool settings result in the “best” chimeras?
  - Learning rate, batch size, diversity of samples, etc.
  - If we move from embeddings to Dreambooth models or Hypernetworks do we get more power?
- Are these images detectable using the same techniques used against StyleGAN3 and other generative models?



JOHNS HOPKINS
APPLIED PHYSICS LABORATORY