Three Challenges Ahead for Stable Diffusion

Originally published at https://www.unite.ai/three-challenges-ahead-for-stable-diffusion/

The release of stability.ai's Stable Diffusion latent diffusion image synthesis model a couple of weeks ago may be one of the most significant technological disclosures since DeCSS in 1999; it's certainly the biggest event in AI-generated imagery since the 2017 deepfakes code was copied over to GitHub and forked into what would become DeepFaceLab and FaceSwap, as well as the real-time streaming deepfake software DeepFaceLive.

At a stroke, user frustration over the content restrictions in DALL-E 2's image synthesis API were swept aside, as it transpired that Stable Diffusion's NSFW filter could be disabled by changing a sole line of code. Porn-centric Stable Diffusion Reddits sprung up almost immediately, and were as quickly cut down, while the developer and user camp divided on Discord into the official and NSFW communities, and Twitter began to fill up with fantastical Stable Diffusion creations.

At the moment, each day seems to bring some amazing innovation from the developers who have adopted the system, with plugins and third-party adjuncts being hastily written for Krita, Photoshop, Cinema4D, Blender, and many other application platforms.

Stable Diffusion Krita Addon

In the meantime, promptcraft – the now- professional art of 'AI whispering', which may end up being the shortest career option since 'Filofax binder' – is already becoming commercialized, while early monetization of Stable Diffusion is taking place at the Patreon level, with the certainty of more sophisticated offerings to come, for those unwilling to navigate Conda-based installs of the source code, or the proscriptive NSFW filters of web-based implementations.

The pace of development and free sense of exploration from users is proceeding at such a dizzying speed that it's difficult to see very far ahead. Essentially, we don't know exactly what we're dealing with yet, or what all the limitation or possibilities might be.

Nonetheless, let's take a look at three of what might be the most interesting and challenging hurdles for the rapidly-formed and rapidly-growing Stable Diffusion community to face and, hopefully, overcome.

1: Optimizing Tile-Based Pipelines

Presented with limited hardware resources and hard limits on the resolution of training images, it seems likely that developers will find workarounds to improve both the quality and the resolution of Stable Diffusion output. A lot of these projects are set to involve exploiting the limitations of the system, such as its native resolution of a mere 512×512 pixels.

As is always the case with computer vision and image synthesis initiatives, Stable Diffusion was trained on square ratio images, in this case resampled to 512×512, so that the source images could be regularized and able to fit into the constraints of the GPUs that trained the model.

Therefore Stable Diffusion 'thinks' (if it thinks at all) in 512×512 terms, and certainly in square terms. Many users currently probing the limits of the system report that Stable Diffusion produces the most reliable and least glitchy results at this rather constrained aspect ratio (see 'addressing extremities' below).

Though various implementations feature upscaling via RealESRGAN (and can fix poorly rendered faces via GFPGAN) several users are currently developing methods to split up images into 512x512px sections and stitch the images together to form larger composite works.

This 1024×576 render, a resolution customarily impossible in a single Stable Diffusion render, was created by copying and pasting the attention.py Python file from DoggettX fork of Stable Diffusion (a version which implements tile-based upscaling) into another fork. Source: https://old.reddit.com/r/StableDiffusion/comments/x6yeam/1024x576_with_6gb_nice/

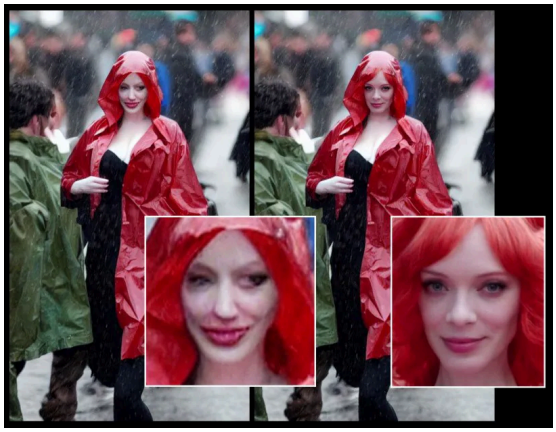
Though some initiatives of this kind are using original code or other libraries, the txt2imghd port of GOBIG (a mode in the VRAM-hungry ProgRockDiffusion) is set to provide this functionality to the main branch soon. While txt2imghd is a dedicated port of GOBIG, other efforts from community developers involves different implementations of GOBIG.

A conveniently abstract image in the original 512x512px render (left and second from left); upscaled by ESGRAN, which is now more or less native across all Stab given 'special attention' via an implementation of GOBIG, producing detail that, at least within the confines of the image section, seem better-upscaled. Source: https://old.reddit.com/r/StableDiffusion/comments/x72460/stable_diffusion_gobig_txt2imghd_easy_mode_colab/

The kind of abstract example featured above has many 'little kingdoms' of detail that suit this solipsistic approach to upscaling, but which may require more challenging code-driven solutions in order to produce non-repetitive, cohesive upscaling that doesn't *look* like it was assembled from many parts. Not least, in the case of human faces, where we are unusually attuned to aberrations or 'jarring' artifacts. Therefore faces may eventually need a dedicated solution.

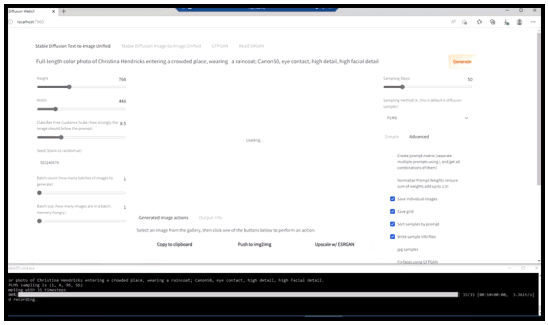
Stable Diffusion currently has no mechanism for focusing attention on the face during a render in the same way that humans prioritize facial information. Though some developers in the Discord communities are considering methods to implement this kind of 'enhanced attention', it is currently much easier to manually (and, eventually, automatically) enhance the face after the initial render has taken place.

A human face has an internal and complete semantic logic that won't be found in a 'tile' of the bottom corner of (for instance) a building, and therefore it's currently possible to very effectively 'zoom in' and re-render a 'sketchy' face in Stable Diffusion output.



Left, Stable Diffusion's initial effort with the prompt 'Full-length color photo of Christina Hendricks entering a crowded place, wearing a raincoat; Canon50, eye contact, high detail, high facial detail'. Right, an improved face obtained by feeding the blurred and sketchy face from the first render back into the full attention of Stable Diffusion using Img2Img (see animated images below).

In the absence of a dedicated Textual Inversion solution (see below), this will only work for celebrity images where the person in question is already well-represented in the LAION data subsets that trained Stable Diffusion. Therefore it will work on the likes of Tom Cruise, Brad Pitt, Jennifer Lawrence, and a limited range of genuine media luminaries that are present in great numbers of images in the source data.

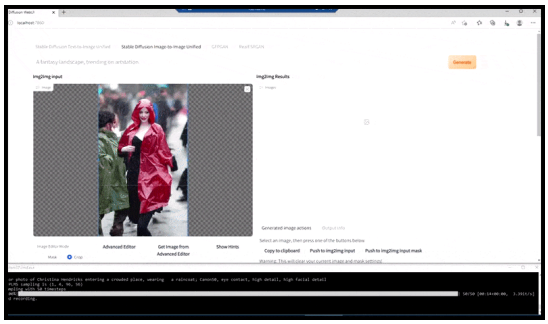


Generating a plausible press picture with the prompt 'Full-length color photo of Christina Hendricks entering a crowded place, wearing a raincoat; Canon50, eye contact, high detail, high facial detail'.

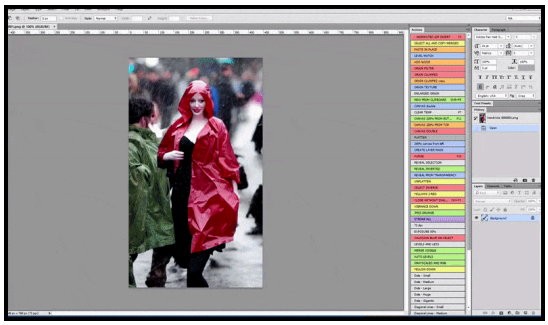
For celebrities with long and enduring careers, Stable Diffusion will usually generate an image of the person at a recent (i.e. older) age, and it will be necessary to add prompt adjuncts such as *'young'* or *'in the year [YEAR]'* in order to produce younger-looking images.

With a prominent, much-photographed and consistent career spanning nearly 40 years, actress Jennifer Connelly is one of a handful of celebrities in LAION that a a range of ages. Source: prepack Stable Diffusion, local, v1.4 checkpoint; age-related prompts.

This is largely because of the proliferation of digital (rather than expensive, emulsion-based) press photography from the mid-2000s on, and the later growth in volume of image output due to increased broadband speeds.

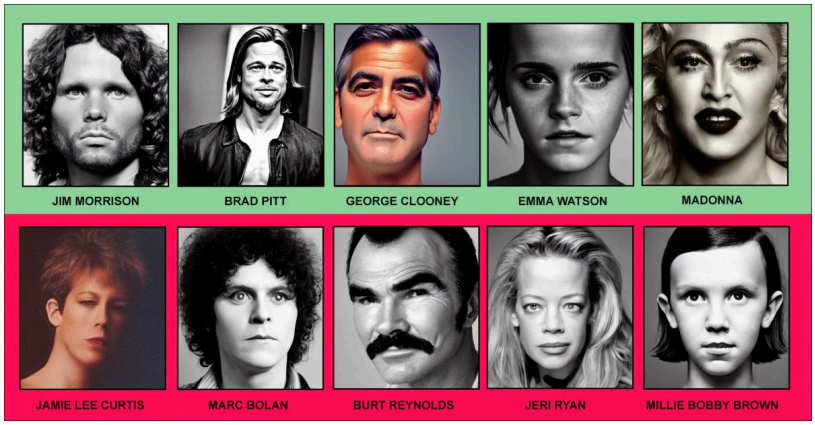


The rendered image is passed through to Img2Img in Stable Diffusion, where a 'focus area' is selected, and a new, maximum-size render is made only of that area, allowing Stable Diffusion to concentrate all available resources on recreating the face.



Compositing the 'high attention' face back into the original render. Besides faces, this process will only work with entities that have a potential known, cohesive and integral appearance, such as a portion of the original photo that has a distinct object, such as a watch or a car. Upscaling a section of – for instance – a wall is going to lead to a very strange-looking reassembled wall, because the tile renders had no wider context for this 'jigsaw piece' as they were rendering.

Some celebrities in the database come 'pre-frozen' in time, either because they died early (such as Marilyn Monroe), or rose to only fleeting mainstream prominence, producing a high volume of images in a limited period of time. Polling Stable Diffusion arguably provides a kind of 'current' popularity index for modern and older stars. For some older and current celebrities, there aren't enough images in the source data to obtain a very good likeness, while the enduring popularity of particular long-dead or otherwise faded stars ensure that their reasonable likeness can be obtained from the system.



Stable Diffusion renders quickly reveal which famous faces are well-represented in the training data. Despite her enormous popularity as an older teenager at the Brown was younger and less well-known when the LAION source datasets were scraped from the web, making a high-quality likeness with Stable Diffusion proble

Where the data is available, tile-based up-res solutions in Stable Diffusion could go further than homing in on the face: they could potentially enable even more accurate and detailed faces by breaking the facial features down and turning the entire force of local GPU resources on salient features individually, prior to reassembly – a process which is currently, again, manual.

This is not limited to faces, but it is limited to parts of objects that are at least as predictably-placed in the wider context of the host object, and which conform to high-level embeddings that one could reasonably expect to find in a hyperscale dataset.

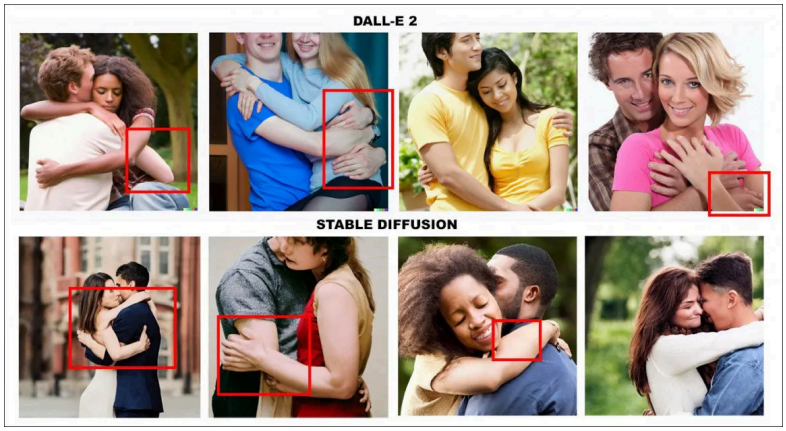
The real limit is the amount of available reference data in the dataset, because, eventually, deeply-iterated detail will become totally 'hallucinated' (i.e. fictitious) and less authentic.

Such high-level granular enlargements work in the case of Jennifer Connelly, because she is well-represented across a range of ages in LAION-aesthetics (the primary subset of LAION 5B that Stable Diffusion uses), and generally across LAION; in many other cases, accuracy would suffer from lack of data, necessitating either fine-tuning (additional training, see 'Customization' below) or Textual Inversion (see below).

Tiles are a powerful and relatively cheap way for Stable Diffusion to be enabled to produce hi-res output, but algorithmic tiled upscaling of this kind, if it lacks some kind of broader, higher-level attention mechanism, may fall short of the hoped-for standards across a range of content types.

2: Addressing Issues with Human Limbs

Stable Diffusion doesn't live up to its name when depicting the complexity of human extremities. Hands can multiply randomly, fingers coalesce, third legs appear unbidden, and existing limbs vanish without trace. In its defense, Stable Diffusion shares the problem with its stablemates, and most certainly with DALL-E 2.



Non-edited results from DALL-E 2 and Stable Diffusion (1.4) at the end of August 2022, both showing issues with limbs. Prompt is 'A woman embracing a man'

Stable Diffusion fans hoping that the forthcoming 1.5 checkpoint (a more intensely trained version of the model, with improved parameters) would solve the limb confusion are likely to be disappointed. The new model, which will be released in about two weeks' time, is currently being premiered at the commercial stability.ai portal DreamStudio, which uses 1.5 by default, and where users can compare the new output with renders from their local or other 1.4 systems:

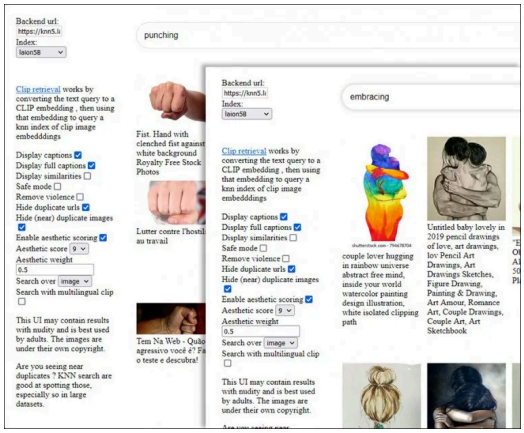
Source: Local 1.4 prepack and https://beta.dreamstudio.ai/

Source: Local 1.4 prepack and https://beta.dreamstudio.ai/

Source: Local 1.4 prepack and https://beta.dreamstudio.ai/

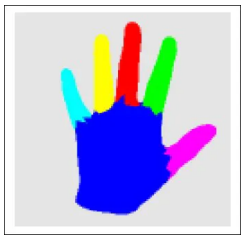
As is often the case, data quality could well be the primary contributing cause.

The open source databases that fuel image synthesis systems such as Stable Diffusion and DALL-E 2 are able to provide many labels for both individual humans and inter-human action. These labels get trained-in symbiotically with their associated images, or segments of images.



Stable Diffusion users can explore the concepts trained into the model by querying the LAION-aesthetics dataset, a subset of the larger LAION 5B dataset, which powers the system. The images are ordered not by their alphabetical labels, but by their 'aesthetic score'. Source: https://rom1504.github.io/clip-retrieval/

A *good hierarchy* of Individual labels and classes contributing to the depiction of a human arm would be something like *body>arm>hand>fingers>[sub digits + thumb]> [digit segments]>Fingernails*.



Granular semantic segmentation of the parts of a hand. Even this unusually detailed deconstruction leaves each 'finger' as a sole entity, not accounting for the three sections of a finger and the two sections of a thumb. Source: https://athitsos.utasites.cloud/publications/rezaei_petra2021.pdf

In reality, the source images are unlikely to be so consistently annotated across the entire dataset, and unsupervised labeling algorithms will probably stop at the *higher* level of – for instance – 'hand', and leave the interior pixels (which technically contain 'finger' information) as an unlabeled mass of pixels from which features will be arbitrarily derived, and which may manifest in later renders as a jarring element.

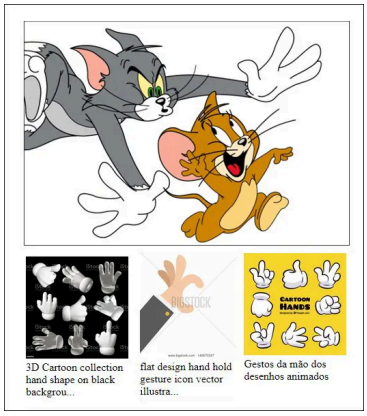
How it should be (upper right, if not upper-cut), and how it tends to be (lower right), due to limited resources for labeling, or architectural exploitation of such labels if they do exist in the dataset.

Thus, if a latent diffusion model gets as far as rendering an arm, it's almost certainly going to at least have a go at rendering a hand at the end of that arm, because *arm>hand* is the minimal requisite hierarchy, fairly high up in what the architecture knows about 'human anatomy'.

After that, 'fingers' may be the smallest grouping, even though there are 14 further finger/thumb sub-parts to consider when depicting human hands.

If this theory holds, there is no real remedy, due to the sector-wide lack of budget for manual annotation, and the lack of adequately effective algorithms that could automate labeling while producing low error rates. In effect, the model may currently be relying on human anatomical consistency to paper over the shortcomings of the dataset it was trained on.

One possible reason why it *can't* rely on this, recently proposed at the Stable Diffusion Discord, is that the model could become confused about the correct number of fingers a (realistic) human hand should have because the LAION-derived database powering it features cartoon characters that may have fewer fingers (which is in itself a labor-saving shortcut).



Two of the potential culprits in 'missing finger' syndrome in Stable Diffusion and similar models. Below, examples of cartoon hands from the LAION-aesthetics dataset powering Stable Diffusion. Source: https://www.youtube.com/watch?v=0QZFQ3gbd6I

If this is true, then the only obvious solution is to retrain the model, excluding non-realistic human-based content, ensuring that genuine cases of omission (i.e. amputees) are suitably labeled as exceptions. From a data curation point alone, this would be quite a challenge, particularly for resource-starved community efforts.

The second approach would be to apply filters which exclude such content (i.e. 'hand with three/five fingers') from manifesting at render time, in much the same way that OpenAI has, to a certain extent, filtered GPT-3 and DALL-E 2, so that their output could be regulated without needing to retrain the source models.

For Stable Diffusion, the semantic distinction between digits and even limbs can become horrifically blurred, bringing to mind the 1980s 'body horror' strand of horror movies from the likes of David Cronenberg. Source: https://old.reddit.com/r/StableDiffusion/comments/x6htf6/a_study_of_stable_diffusions_strange_relationship/

However, again, this would require labels that may not exist across all the affected images, leaving us with the same logistical and budgetary challenge.

It could be argued that there are two remaining roads forward: throwing more data at the problem, and applying third-party interpretive systems that can intervene when physical goofs of the type described here are being presented to the end user (at the very least, the latter would give OpenAI a method to provide refunds for 'body horror' renders, if the company was motivated to do so).

3: Customization

One of the most exciting possibilities for the future of Stable Diffusion is the prospect of users or organizations developing revised systems; modifications that allow content outside of the pretrained LAION sphere to be integrated into the system – ideally without the ungovernable expense of training the entire model over again, or the risk entailed when training in a large volume of novel images to an existing, mature and capable model.

By analogy: if two less-gifted students join an advanced class of thirty students, they'll either assimilate and catch up, or fail as outliers; in either case, the class average performance will probably not be affected. If 15 less-gifted students join, however, the grade curve for the entire class is likely to suffer.

Likewise, the synergistic and fairly delicate network of relationships that are built up over sustained and expensive model training can be compromised, in some cases effectively destroyed, by excessive new data, lowering the output quality for the model across the board.

The case for doing this is primarily where your interest lies in completely hi-jacking the model's conceptual understanding of relationships and things, and appropriating it for the exclusive production of content that's similar to the additional material that you added.

Thus, training 500,000 *Simpsons* frames into an existing Stable Diffusion checkpoint is likely, eventually, to get you a better *Simpsons* simulator than the original build could have offered, presuming that enough broad semantic relationships survive the process (i.e. *Homer Simpson eating a hotdog*, which may require material about hot-dogs that was not in your additional material, but did already exist in the checkpoint), and presuming that you don't want to suddenly switch from *Simpsons* content to creating *fabulous landscape by Greg Rutkowski* – because your post-trained model has had its attention massively diverted, and won't be as good at doing that kind of thing as it used to be.

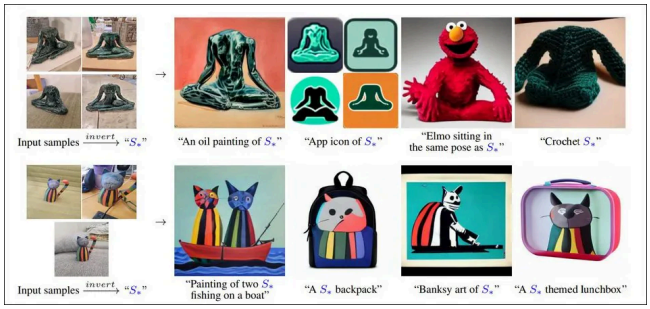
One notable example of this is waifu-diffusion, which has successfully post-trained 56,000 anime images into a completed and trained Stable Diffusion checkpoint. It's a tough prospect for a hobbyist, though, since the model requires an eye-watering minimum of 30GB of VRAM, far beyond what's likely to be available at the consumer tier in NVIDIA's forthcoming 40XX series releases.

The training of custom content into Stable Diffusion via waifu-diffusion: the model took two weeks of post-training in order to output this level of illustration. The six progress of the model, as training proceeded, in making subject-coherent output based on the new training data. Source: https://gigazine.net/gsc_news/en/202201

A great deal of effort could be expended on such 'forks' of Stable Diffusion checkpoints, only to be stymied by technical debt. Developers at the official Discord have already indicated that later checkpoint releases are not necessarily going to be backward-compatible, even with prompt logic that may have worked with a previous version, since their primary interest is in obtaining the best model possible, rather than supporting legacy applications and processes.

Therefore a company or individual that decides to branch off a checkpoint into a commercial product effectively has no way back; their version of the model is, at that point, a 'hard fork', and won't be able to draw in upstream benefits from later releases from stability.ai – which is quite a commitment.

The current, and greater hope for customization of Stable Diffusion is Textual Inversion, where the user trains in a small handful of CLIP-aligned images.



A collaboration between Tel Aviv University and NVIDIA, textual inversion allows for the training-in of discrete and novel entities, without destroying the capabilities of the source model. Source: https://textual-inversion.github.io/

The primary apparent limitation of textual inversion is that a very low number of images are recommended – as few as five. This effectively produces a limited entity that may be more useful for style transfer tasks rather than the insertion of photorealistic objects.

Nonetheless, experiments are currently taking place within the various Stable Diffusion Discords that use much higher numbers of training images, and it remains to be seen how productive the method might prove. Again, the technique requires a great deal of VRAM, time, and patience.

Due to these limiting factors, we may have to wait a while to see some of the more sophisticated textual inversion experiments from Stable Diffusion enthusiasts – and whether or not this approach can 'put you in the picture' in a manner that looks better than a Photoshop cut-and-paste, while retaining the astounding functionality of the official checkpoints.

First published 6th September 2022.

Writer on machine learning, domain specialist in human image synthesis. Former head of research content at Metaphysic.ai, until its dissolution into DNEG's Brahma.ai.
Portfolio site: martinanderson.ai
Contact: martin@martinanderson.ai

Discover More