TLDR (courtesy of Chat-GPT): A text to image AI model has been developed using 23,088 samples from Yande.re. The captions for the training data were done through the use of 3 iterations of WD1.4 tagger and a reduced threshold. The model was trained using Adam optimizers and the training was done at different resolutions (512pix and 768pix) with different learning rates over the course of 90 epochs. A block merge with BasilMix was attempted at the 20th epoch, but it was evident that the merged weights were quickly being trained out, so the final release was not block merged. The model was trained on a single RTX 4090 GPU using EveryDream 2 Trainer. Checkpoints were saved every 5 epochs and verification testing was done throughout the training process.

On to the process!

The model was trained using 23,088 samples from Yande.re. File captions were done through the use of 3 iterations of WD1.4 tagger to ensure maximum identification of objects within the training data. A second captioning run was done using one tagger with a reduced threshold to produce shorter captions for later use. These tasks were undertaken by 金Goldkoron and I may be inaccurate in reporting this process.

Theory: Adam optimizers use adaptive LR to adjust LR per batch size and over the course of the run to optimize training to better reach global optima. However, all previous training runs have shown that with a lower LR little progress is made, and with a higher LR it quickly gets stuck in a local minima. I utilized chaining to “kickstart,” or rest, Adam to its highest lambda value to rapidly improve learning, while manually setting the max lambda value for LR via stepped decay per chain. I also used a cosine LR, without warmup steps, while estimating decay rate, to best accomplish a cosine LR across the entire length of training. Essentially, each next chain picks up where the previous left off, restarting the cosine. I used proportionality when estimating LR for each additional batch size.

Base model for training: The base model chosen to train on top of was ultimately NAI. While we had some moral concerns of using leaked intellectual property, a large (and unknown) number of existing anime models already incorporate NAI in some form or another and many don’t disclose this. Many other anime models also already feature high stylization and we wanted as “vanilla” of an option as possible while still allowing an achievable result on a single GPU. This was so we could account for all the intellectual property being used, and have as minimal influence from various extraneous weights as possible. As such, we would rather use the early initial leaked version of NAI knowingly, rather than unknowingly incorporate it into our model via someone else’s merge or fine tune. We also wanted to see the best representation of our training data on top of a known model used by others as a base. This would give us the most accurate comparison between our model and models such as AnythingV3, AOM, etc. Full credits go to the NAI team for making this possible!

Training details: Training was done throughout its length at a batch size 4 on an RTX 4090. All training was done on EveryDream 2 Trainer (https://github.com/victorchall/EveryDream2trainer), utilizing DDIM sample scheduler and DDPM noise scheduler (using linear beta schedule). Adam8bit was the optimizer used, mix precision was used (amp setting in ED2, essentially all at fp32 except VAE which is at fp16), and xformers enabled.

Conditional dropout of tags was set to 0.125.

Due to the length of the captions generated by the auto-taggers, tags are to be shuffled every 10ep via a python script to circumvent captions being truncated by CLIP.

The first 30 epochs (ep) were done at a base resolution of 512pix using aspect bucketing of ED2. The LR manually stepped every 10ep using the following method: 3e-06, 2.6e-06, 2.2e-06. All LR cosine decay

The next 40ep were done at a base resolution of 768pix using aspect bucketing. LR manually stepped every 10ep using the following method: 3e-06, 2.6e-06, 2.2e-06, 1.6e-06. All LR using cosine decay

The next 20ep were done at a base resolution of 768pix using aspect bucketing and reduced length tags at a LR of 2.2e-06 constant for additional fine tuning.

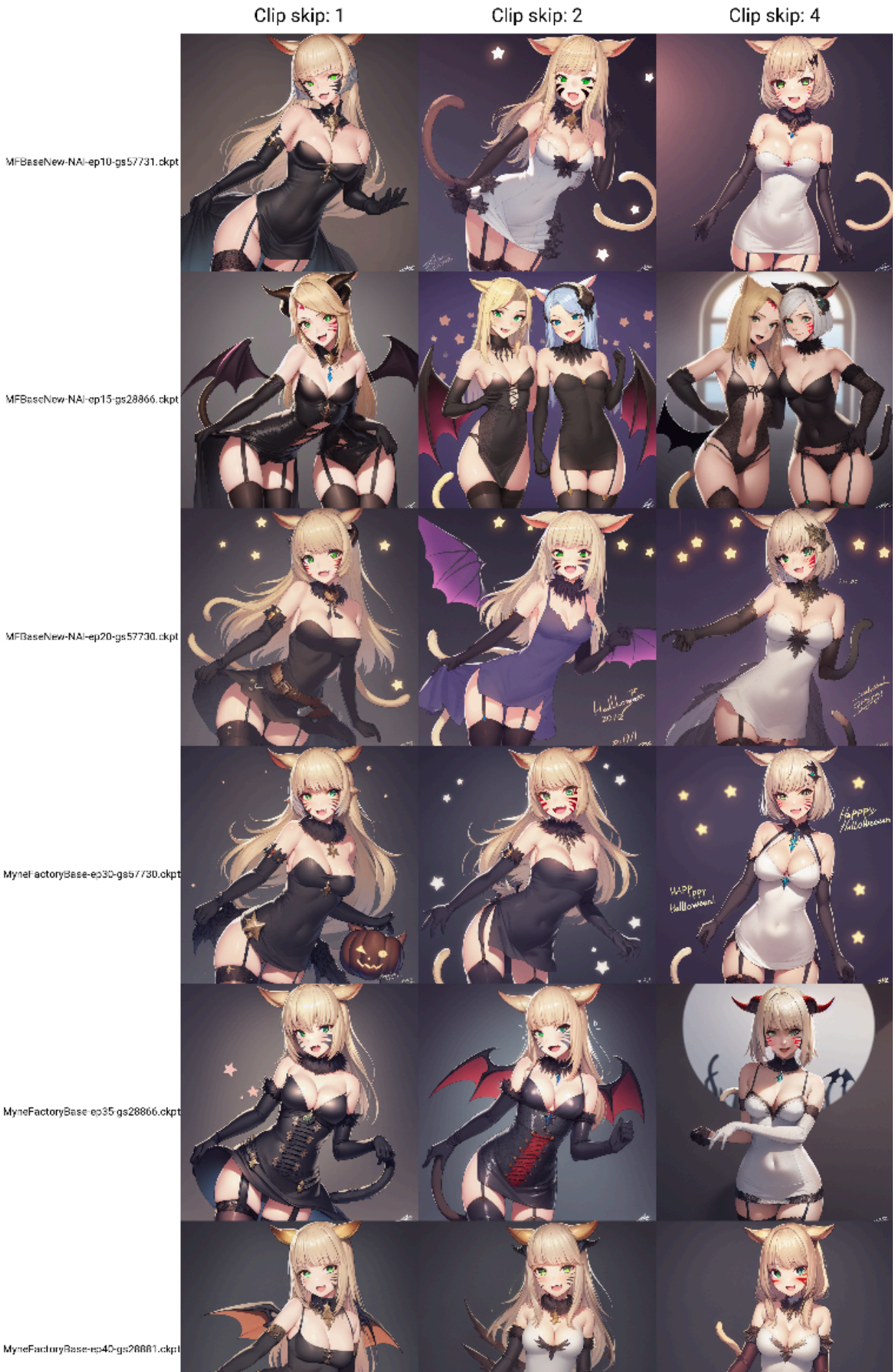
The training was doneThe last 20 epochs were trained at 768 resolution at 2.2e-06 constant learning rate (with text encoder frozen), utilizing shorter captions to improve the compositional quality of generations using shorter prompts; as previous testing had indicated a red shift when using less tags in prompts and vastly reduced background quality.

Checkpoints were saved every 5ep, and verification testing was done by 金Goldkoron on various milestones throughout the training process to ensure quality. Logs were also simultaneously monitored via Tensorboard to ensure gradient divergence did not occur.

Notes on text encoder training: Text Encoder was trained for 50% of the training durations, freezing and unfreezing every 10ep. During the final 20ep of finetuning, the TE was frozen.

Notes on Block Merge:
It should be noted that at the ep20 milestone we did block merge with BasilMix using the following weights:
1,0.9,0.7,0.5,0.3,0.1,1,1,1,1,1,1,0,0,0,0,0,0,0,0.1,0.3,0.5,0.7,0.9,1
This is the same method used to create Abyss Orange Mix 2

However, it was evident by ep25 that the merged weights were being trained out quickly. By the start of 768 res training the weights were not detectable. Upon completion of MyneFactory Base, at ep90, I ran a test comparing all iterations of Myne Factory Base; and it is evident by the end of the training the weights had entirely shifted back to our training data (see sample at end of document). While I don't believe Basil Mix is supplying any weights of any significance, it is worth noting that there may have been some influence from it in the final product. Ultimately, we decided we enjoyed the aesthetics of our model as-is, and we decided a block merge was not the best way to go. However, feel free to attempt a block merge of the final product with Basil Mix using the above weights. It does change textures and shading, and some may prefer the more realistic feel. Ultimately, we found that any merging of weights pre (or during) training is a wasted effort with unpredictable results. And being as this is a base foundation for future anime-specific models, block merging the final release for our purposes would be less than optimal due to the fact that the weights would be overwritten again in future fine tuning.


Clip skip: 1
Clip skip: 2
Clip skip: 4
MFBaseNew-NAI-ep10-gs57731.ckpt
MFBaseNew-NAI-ep15-gs28866.ckpt
MFBaseNew-NAI-ep20-gs57730.ckpt
MyneFactoryBase-ep30-gs57730.ckpt
MyneFactoryBase-ep35-gs28866.ckpt
MyneFactoryBase-ep40-gs28881.ckpt