李哲維 R12922205 hw2 2024/10/29

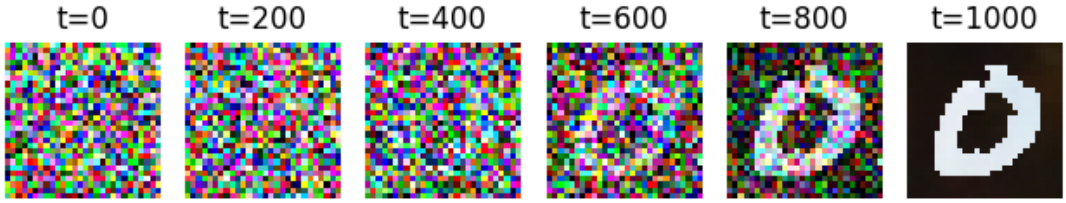
Problem 1:

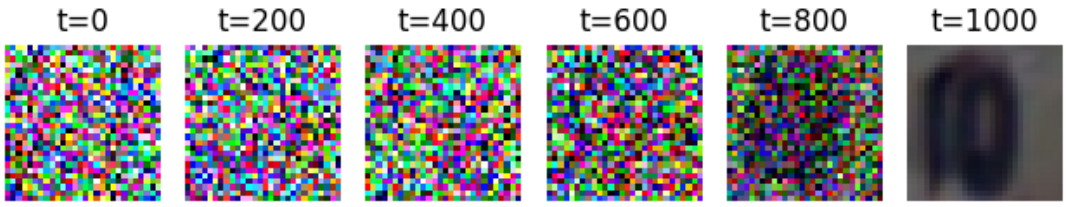
1. (5%) Describe your implementation details and the difficulties you encountered.

   There were multiple obstacles for implementing a DDPM algorithm from scratch. A big part of them is to understand the meaning of the formula and the reason to do so, and even though I had successfully implemented it, I still do not fully understand the math behind. The other part is to enhance the accuracy of the generated samples, since I initially stuck at average acc=80% and did not know what was wrong. After costing a lot of effort and hours on testing every single component mentioned in the DDPM thesis, it finally cross the baseline at least.

2. (5%) Please show 10 generated images for each digit (0-9) from both MNIST-M & SVHN dataset in your report. You can put all 100 outputs in one image with columns indicating different noise inputs and rows indicating different digits. [see the below MNIST-M example, you should visualize BOTH MNIST-M & SVHN]

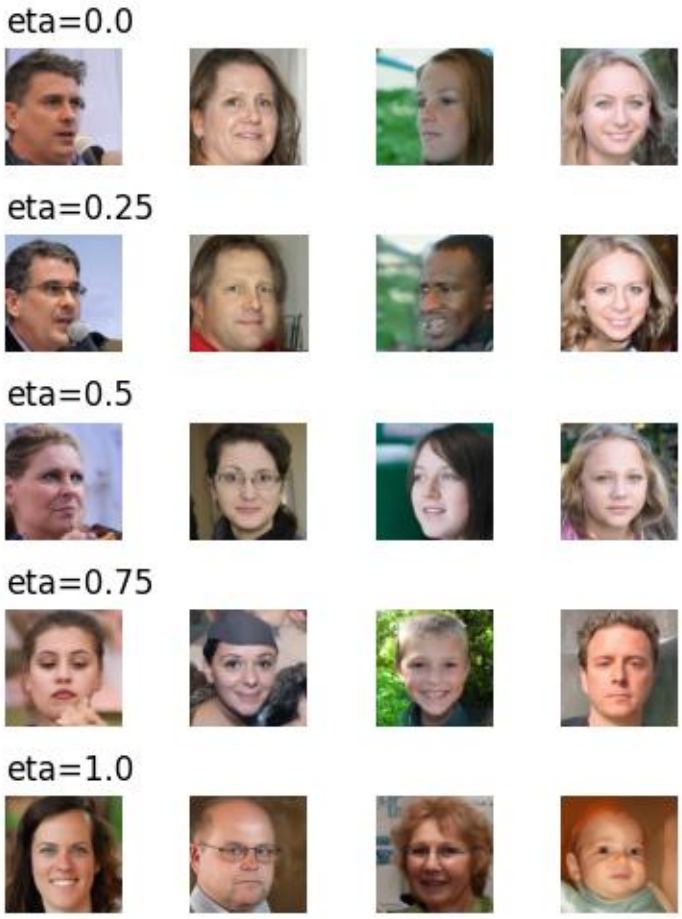
3. (5%) Visualize a total of six images from both MNIST-M & SVHN datasets in the reverse process of the first “0” in your outputs in (2) and with different time steps. [see the MNIST-M example below, but you need to visualize BOTH MNIST-M & SVHN]


t=0
t=200
t=400
t=600
t=800
t=1000


t=0
t=200
t=400
t=600
t=800
t=1000

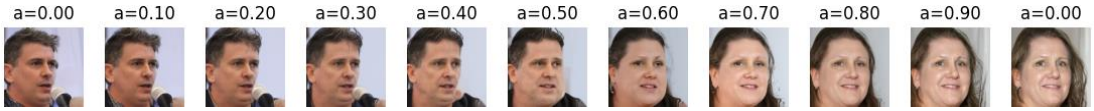
Problem 2:

1. (7.5%) Please generate face images of noise 00.pt ~ 03.pt with different eta in one grid. Report and explain your observation in this experiment.



Explain: the result shows that when the eta grows, the variation of the image compared to the original image becomes larger.

2. (7.5%) Please generate the face images of the interpolation of noise 00.pt ~ 01.pt. The interpolation formula is spherical linear interpolation, which is also known as slerp. What will happen if we simply use linear interpolation? Explain and report your observation. (There should be **two images** in your report, one for spherical linear and the other for linear)



Explain: The first row of image fusing noise 00.pt and 01.pt by using linear interpolation, the second one is generated by using slerp. It is obvious that the second image show a more ideal and reasonable process in terms of fusing two distinct images by a ratio (alpha). The ones using linear interpolation show large part of background being blank (i.e. pink space) which might imply the inability for the model to represent/predict correctly (the combination result is outside of the learned distribution), while the other one (slerp) can generate appropriate background. The result suggests that simply using linear interpolation might not be enough for fusing two noises sample in order to fit the learned distribution.

Problem 3:

1. (7.5%) Conduct the CLIP-based zero shot classification on the hw2_data/clip_zeroshot/val, explain how CLIP do this, report the accuracy and 5 successful/failed cases.

Ans:

Top-1 accuracy: 58.72, Top-5 accuracy: 85.92

Explain:

CLIP (Contrastive Language-Image Pretraining) performs zero-shot classification by leveraging a joint understanding of images and text. It is trained on a vast dataset of image-text pairs, allowing the model to learn representations that connect visual content with linguistic concepts.

During classification, CLIP generates an embedding for the input image and creates corresponding embeddings for each potential class label. It then calculates the similarity between the image embedding and the class text embeddings, selecting the label with the highest score as the prediction.

This approach enables CLIP to effectively classify images without requiring

additional training on specific target classes.

The 5 successful/failed cases: shown on the graph below

- 5 successful cases (first row):
  - ['32_458.png', '24_485.png', '39_473.png', '7_455.png', '12_486.png']
- 5 failed cases (second row):
  - ['27_459.png', '17_462.png', '1_470.png', '37_462.png', '41_481.png']

2. (7.5%) What will happen if you simply generate an image containing multiple concepts (e.g., a <new1> next to a <new2>)? You can use your own objects or the cat images provided in the dataset. Share your findings and survey a related paper that works on multiple concepts personalization and share their method.

Ans:

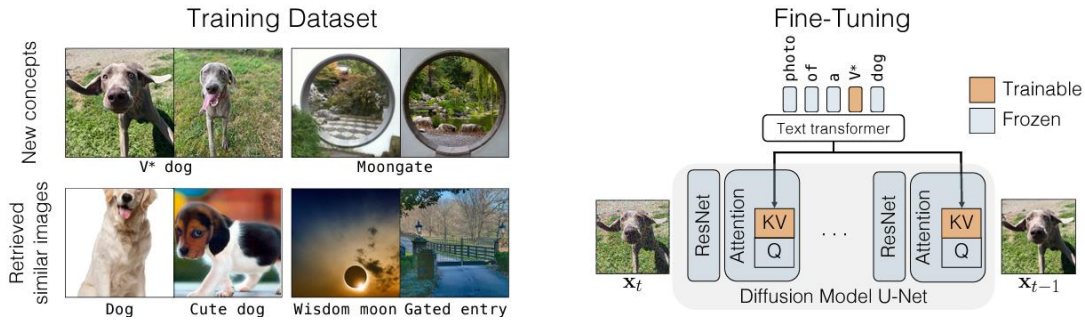
Several results might occur by simply putting the two new concepts learned by textual inversion into one input prompt.

First, the blending of concepts would happen, since the model might attempt to blend the two concepts, leading to an image that incorporates features of both but might not fully capture the distinctiveness of either concept.

Second, context confusion might occur if the relationship between the two concepts isn't clear or cannot be initialized by the existing pretrained embeddings, the model may produce an ambiguous image.

Survey: Multi-Concept Customization of Text-to-Image Diffusion

This work, *Custom Diffusion*, provides an efficient method for augmenting existing text-to-image models. The overview of their method is shown in the figure below. Since this model focuses on efficiency in computation and memory, it only fine-tunes a specific subset of model weights related to cross-attention layers to incorporate new concepts without causing the model forgetting. This is achieved by using a small set of real images with similar captions and employing augmentation during fine-tuning, resulting in faster convergence and better outcomes. Additionally, the method allows for training on multiple concepts either simultaneously or separately, followed by merging.



Given images of new concepts, this work retrieves real images with similar captions as the given concepts and creates the training dataset for fine-tuning, (left part of figure). To represent personal concepts of a general category, this work introduce a new modifier token V*, used in front of the category name. During training, we optimize key and value projection matrices in the diffusion model cross-attention layers along with the modifier token.