Deep Learning for Computer Vision HW2 Report

B10901091 李冠儀

P1.

(5%) Describe your implementation details and the difficulties you encountered.
Implementation details: I follow the DDPM in paper, as following picture.

Algorithm 1 Training

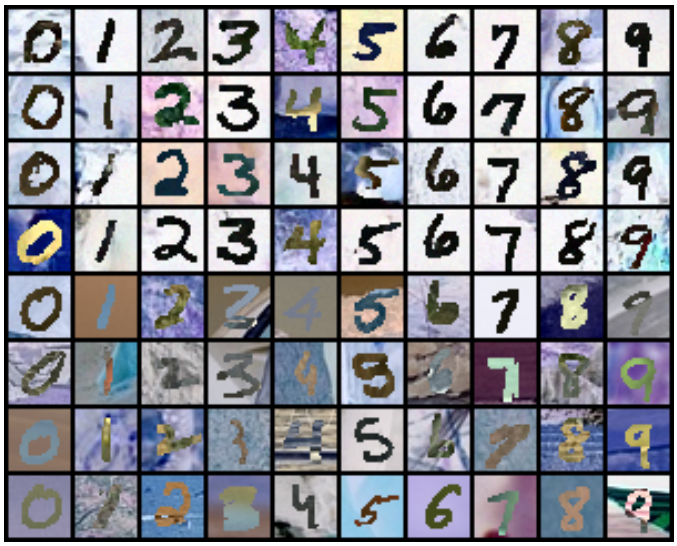
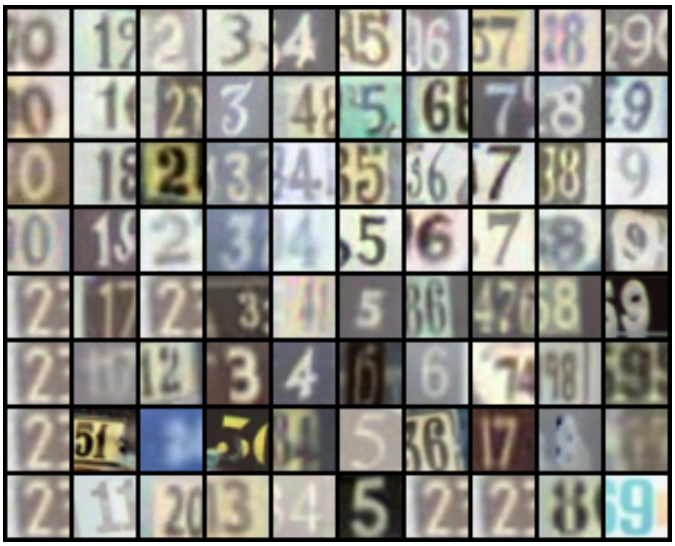
1: **repeat**
2: $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $t \sim \text{Uniform}(\{1, \dots, T\})$
4: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: Take gradient descent step on
$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$
6: **until** converged

Algorithm 2 Sampling

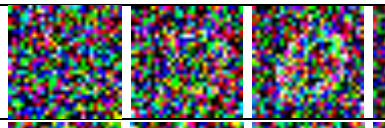
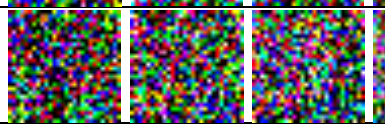
1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \dots, 1$ **do**
3: $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$

Difficulties: DDPM has many implementations detail, so I should implement carefully. And I encounter many difficulties when debugging, I think I should test from basis module instead of implement the whole code then debug.

(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 in puts and rows indicating different digits.

MNIST-M	SVHN
	

(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**.

MNIST-M	
SVHN	

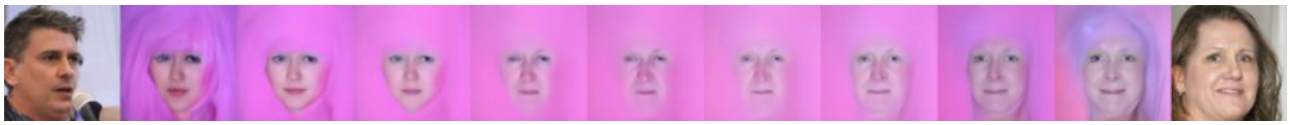
P 2.

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.
   (from up to bottom are 0.0, 0.25, 0.5, 0.75, 1.0, from left to right are 00.pt ~ 03.pt)

Observation: eta represents randomness, and the larger the eta, the greater the randomness. Therefore, observing the first raw, when eta = 0, the images will look almost the same every time. As eta increases, the images become less similar to the original one, and by the time eta = 1, it's essentially a completely different person.

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.
   Slerp:

   Linear:



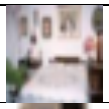
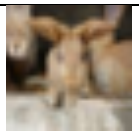
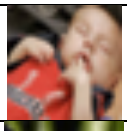
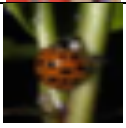
- Observation: **SLERP** ensures smooth, natural transitions between images, maintaining facial integrity throughout the interpolation; **LERP** can lead to distorted images with unnatural transitions, especially at intermediate points.

P 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.

   CLIP encodes both images and text prompts into embeddings using separate encoders. Then, compute the cosine similarity between the image embedding and each text prompt embedding. The text prompt with the highest similarity score is selected as the prediction.

   Accuracy: 71.20%

successful	failed	Failed reason
		True: bed, Predicted: house.
		True: wardrobe, Predicted: skyscraper.
		True: raccoon, Predicted: sweet_pepper.
		True: mouse, Predicted: shrew.
		True: television, Predicted: dolphin.

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 provided cat images in the dataset. Share your findings and survey a related paper that works on multiple concepts personalization, and share their method.

My prompt: <new1> and <new2> shepherd posing proudly on a hilltop with Mount Fuji in the background.

My result:(new2=cat, new1=dog)

Findings:
When testing with simple objects like cats or dogs, the results may show mixed features or unclear separation between concepts. It is worse them simple concept.

Related Paper:
In DreamBooth ("Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation"), the authors handle multiple concepts by:

- Sequential Fine-Tuning: Learning each concept individually and then incorporating multiple concepts.
- Preserving Prior Knowledge: Combining new concepts with the pre-trained model's existing knowledge to avoid overfitting.
- Spatial Layout Control: Using prompts with spatial relationships (e.g., "a cat next to a dog") to manage the interaction between concepts.
- DreamBooth helps mitigate concept confusion by refining how multiple personalized concepts are integrated into generated images.