NVLab Summer School 2023

HW3: Comparison of AE and VAE

Announced Date: 2023/08/09 (Wed.) 14:00~16:00
Due Date: 2023/08/30 (Wed.) 23:59:00

TAs' Contact Information

林江宇 rchildlin@gmail.com
王紹平 kevin30205@gmail.com
盧威樑 franklu00313@gmail.com

I. Grading Policy:

1. In this homework, the code, results, and report (HW3_report.pdf) should be compressed into a ZIP file (HW3_name.zip) and uploaded to Google Drive. Also, please write a README file explaining how to run your code and describe related characteristics in your report. The report format is not limited.
2. Please properly comment on your code to let us understand your train of thought.
3. You are required to finish this homework with Python 3 and build the model using PyTorch. Moreover, built-in machine learning libraries or functions, e.g., *sklearn*, are allowed to be used.
4. If you are unable to submit your assignment on time, please contact TAs in advance.
5. Discussions are encouraged, but plagiarism is strictly prohibited (changing variable names, etc.). You can use any open source you want with a clear reference mentioned in your report. If there is any plagiarism, you will get a 0 on this homework.

II. Submission

Please follow the following format and naming rules.

```
HW3_name.zip
  | --- HW3_report.pdf (Written report)
  | --- README.md
          (Explain how to run your code, load the data/weights, the environment, libraries, etc.)
  | --- HW3.ipynb (The only code file)
  | --- weights (The folder to save your model weights)
  | --- results (The folder to save the results of your experiments, e.g., images, etc.)
```

The zip file doesn't need to contain the dataset, just clearly mention how to load the data and the weights. You need to upload **HW3_name.zip** to your Google Drive and email the share link to franklu00313@gmail.com (cc to the others). After submission, you can confirm whether the submission was successful with the TAs.

III. Project Description

A variational autoencoder is used to learn recoverable data in an unsupervised manner. In this topic, you need to design the autoencoder and variational autoencoder by PyTorch, train on eye datasets, and analyze two models by quantitative and qualitative comparisons, respectively.

Data

There are two classes in the given dataset (data.npy), containing female eyes (0) and male eyes (1). The total number of data is 1476 and the shape of each image is (50,50,3). You have to read and merge them.

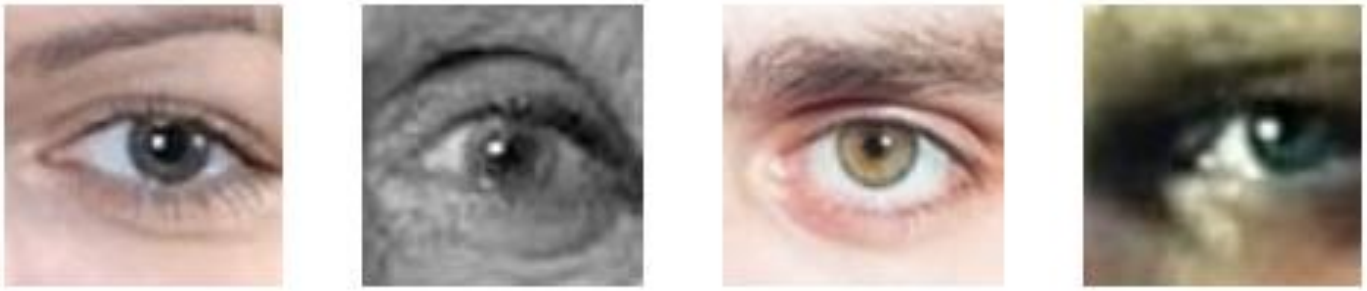

Figure 1. Examples of the given eye dataset

Structure of Dataset

eye.zip
| --- data.npy
| --- label.npy

IV. Implementation

Please employ the autoencoder and variational autoencoder to reconstruct the image.

1. Design your own PyTorch Dataset Class to load the data and labels, then merge them into a Class. You can read the images with any package, e.g., PyTorch, PIL, cv2, etc.

2. Design your CNN-based autoencoder and variational autoencoder model.

3. Train the designed AE and VAE in Step 2 on all 1476 images with the dataset constructed in Step 1. Save the models after finishing the training process.

4. Evaluate the average PSNR and SSIM scores of the trained AE and VAE models in Step 3 on all 1476 images. Note that you should load the trained models before starting evaluations. The average PSNR should surpass 25 and the average SSIM should surpass 0.7 for both AE and VAE models. Please use *skimage* to conduct the evaluations.

5. Save 20 reconstruction results for both AE and VAE models in Step 3 respectively. You can save them as jpg or png files. For fairness, we specified the image ID as [91-95, 481-485, 936-940, 1446-1450]. Note that the image index range is defined as 1-1476.

6. Please add the “same” Gaussian noise into the trained VAE and AE models in Step 3, and repeat Step 5. The demonstration of adding noise to VAE is shown in Fig. 2.

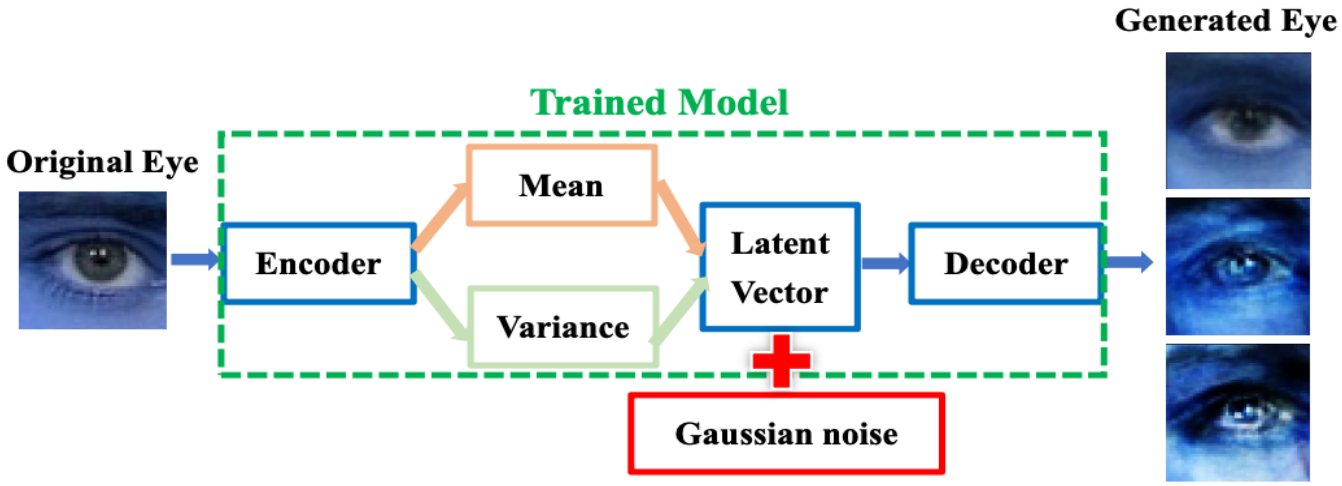

Fig. 2: The diagram of adding the Gaussian noise into a latent vector on VAE

V. Report

The format is not limited, but it should contain at least the following sections:

1. Show your model architectures of AE and VAE.

2. Show your implementation detail, such as batch size, epoch, learning rate, data preprocessing, etc. in detail. Describe how you decide on the hyperparameters to obtain the final AE and VAE models.

3. Show the quantitative and qualitative results of experiments for both AE and VAE models in Step 4, Step 5, and Step 6.

4. Compare the differences between AE and VAE, and explain the reasons for them. You may use quantitative and qualitative results you have obtained to evidence your thought.

5. What can we do to preserve the high-frequency information in the original images in your AE or VAE models? Please discuss this problem and do some experiments to prove it.

6. If we use a very deep Neural Network with a large number of neurons to construct AE and VAE, will the evaluation results increase? Why or why not? Please conduct experiments to demonstrate it.