COMP4211 PA2 Report

Name: Ruiming Min; SID: 20827430; ITSC: rmin

March 17, 2024

Notice:

Since features of the LaTeX file, most of the images and tables are not shown under the discirbation. You can find them by index. Also, you can click the index to jump to the corresponding figure or table.

5 Datasets and Data Loaders

5.3 Dataloader Implementation

[Q1]

Number of images in COCO dataset: 3557. Number of images in WikiArt dataset: 7492.

[Q2]

Number of images in PACS train dataset: 1641. Number of images in PACS test dataset: 2723.

6 Style Transfer

6.2 Model

[Q3]

Since after encoding, the image is represented as a 3D tensor and its 2D shape is much smaller than the original image. Therefore, in the decoder, we need the upsampling-re-construct the image-its original size.

[Q4]

For each layer, we have the trainable parameters as follows:

- Conv2d: $(k \times k \times c_{in} + 1) \times c_{out}$
- Pooling and Upsampling: 0

Therefore, the total number of trainable parameters is 2332511.

[Q5]

Since the we need encoder-encode the image and mine the features, but if we only use the encoder, we cannot re-construct the image. Therefore, we need the decoder-re-construct the image.

[Q6]

Since in this model, we do not only need the style match our target style, but also need the content-be preserved. Therefore, we need-use the content loss and style loss-balance the style transfer.

6.4 Training the Style Transfer model

[Q7]

After 55 epochs, total loss is 13021.544, content loss is 8052.103, and style loss is 4969.441.

6.5 Inference

[Q8]

From the figure 1, we can see the style transfer examples. In these comparisons, we can see that the generated images represented the content of the content images very well expect the words in them. For the style, the generated images are very similar-the style images in color and painting techniques. However, if the style is very different from the content, the generated images may not be very good (e.g.: picture l). Moreover, the generated images have some strange textures in some areas, which may be caused by the imbalance of the style data set, i.e., there are too many paintings in the training set.

7 Classification Task

7.2 Analyzing the Dataset

[Q9]

From the figure 2, and the output of the code (table 1), we can see that the training and testing dataset distributions are very similar expect house and guitar. The ratios of training samples of house and guitar are less than the testing samples. And the number of training samples of each class is a half of the testing samples.

Label	Train Count	Test Count
Giraffe	265	425
Dog	262	407
Elephant	256	370
House	251	386
Person	247	413
Horse	210	382
Guitar	150	340

Table 1: Comparison of training and testing dataset distributions.

[Q10]

Since horse and guitar have less training samples, the model may not be able-learn the features of these two classes very well. Therefore, the model may not be able-classify the horse and guitar very well.

7.3 Model Implementation

[Q11]

For each layer, we have the trainable parameters as follows:

- Dense Layer: $(n_{in} + 1) \times n_{out}$
- Pooling: 0

Therefore, the total number of trainable parameters is 2103303.

7.4 Training the Classification model

[Q12]

After 150 epochs, the cross-entropy loss is 0.856, the detailed loss summary is shown in the table 2.

Table 2: Cross-Entropy Loss Summary Every 10 Epochs

Epoch	Cross-Entropy Loss
10	1.505
20	1.409
30	1.306
40	1.256
50	1.214
60	1.164
70	1.132
80	1.091
90	1.047
100	1.023
110	0.968
120	0.955
130	0.911
140	0.888
150	0.856

7.5 Testing Routine

[Q13]

The accuracy of training dataset is 0.662, and the accuracy of testing dataset is 0.242.
The confusion matrix is shown in the table 3 and table 4.
Also, I visualize the confusion matrix in the figure 3a and figure 3b.

Confusion Matrix for Train

	Predicted Labels						
True Labels	dog	elephant	giraffe	guitar	horse	house	person
dog	237	0	9	6	3	4	3
elephant	217	20	10	1	5	1	2
giraffe	11	1	192	31	14	8	8
guitar	17	3	13	108	5	1	3
horse	9	0	47	12	119	6	17
house	13	1	14	3	9	205	5
person	13	0	13	4	8	5	204

Table 3: Confusion matrix for the training dataset.

Confusion Matrix for Test

	Predicted Labels						
True Labels	dog	elephant	giraffe	guitar	horse	house	person
dog	119	4	96	48	51	50	38
elephant	124	11	63	47	52	40	33
giraffe	106	4	129	90	30	45	20
guitar	102	5	52	79	45	19	38
horse	110	2	73	50	83	38	26
house	83	4	88	28	33	131	19
person	119	5	79	39	40	25	105

Table 4: Confusion matrix for the testing dataset.

[Q14]

The figure 4 shows the misclassified images. These figures show that the model may not be able to classify those images with complex backgrounds and multiple objects very well. Also, they may be too vague, concise and unconventional.

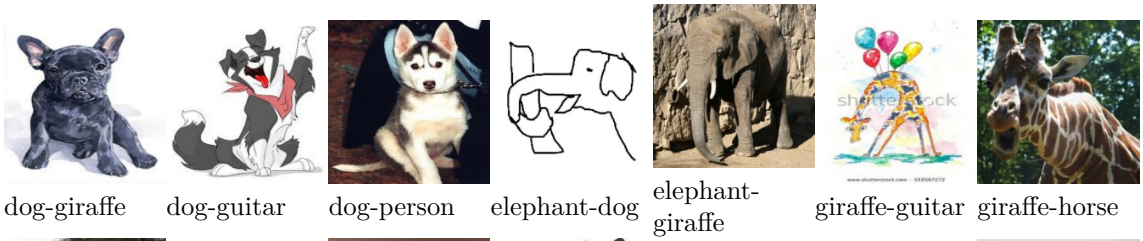


Figure 4: Misclassified Images (the label below is [true label]-[wrong label])

7.6 Data Augmentation by Style Transfer

[Q15]

In figure 5, we can see the example of images with labels and styles.

7.7 Retraining with Augmented Dataset

[Q16]

After 300 epochs, the cross-entropy loss is 0.884.

[Q17]

The accuracy of training dataset is 0.660, and the accuracy of testing dataset is 0.262.

The confusion matrix is shown in the table 5 and table 6.

Also, I visualize the confusion matrix in the figure 6a and figure 6b.

Confusion Matrix for Train

	Predicted Labels						
True Labels	dog	elephant	giraffe	guitar	horse	house	person
dog	361	12	24	22	16	7	13
elephant	261	82	23	28	14	22	20
giraffe	17	6	338	28	19	23	32
guitar	33	17	63	204	9	14	9
horse	18	4	36	18	274	26	26
house	22	3	15	4	11	377	17
person	22	0	31	14	8	17	348

Table 5: Confusion matrix for the training dataset.

Confusion Matrix for Test

	Predicted Labels						
True Labels	dog	elephant	giraffe	guitar	horse	house	person
dog	132	7	75	56	43	51	43
elephant	142	23	52	46	41	31	35
giraffe	111	12	153	51	31	42	25
guitar	107	16	49	77	39	27	24
horse	107	17	54	38	100	42	24
house	100	10	54	33	42	123	23
person	124	10	51	42	48	33	104

Table 6: Confusion matrix for the testing dataset.

Comparing the confusion matrix of the original model and the model with augmented dataset, we can see that the model with augmented dataset has a little bit better performance.

[Q18]

This result may be caused by the fact that the augmented dataset has more samples and more styles, which can help the model learn more features and be more robust.

(a) Content Image

(b) Style Image

(c) Generated Image

(d) Content Image

(e) Style Image

(f) Generated Image

(g) Content Image

(h) Style Image

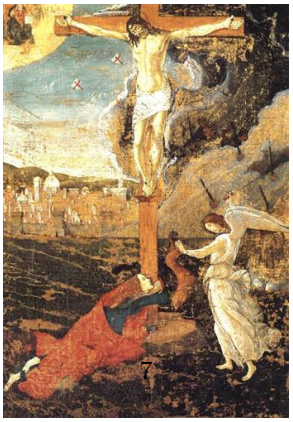
(i) Generated Image

(j) Content Image

(k) Style Image

(l) Generated Image

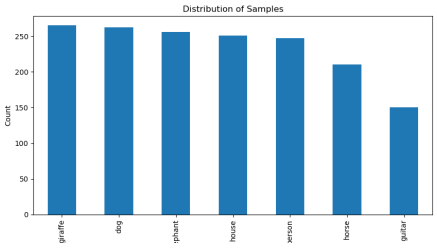
(m) Content Image



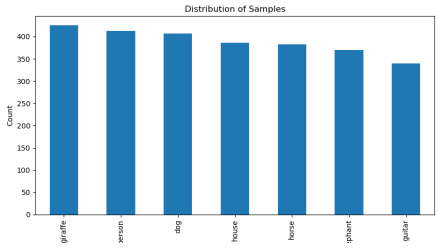
(n) Style Image

(o) Generated Image

Figure 1: Example of Style Transfer

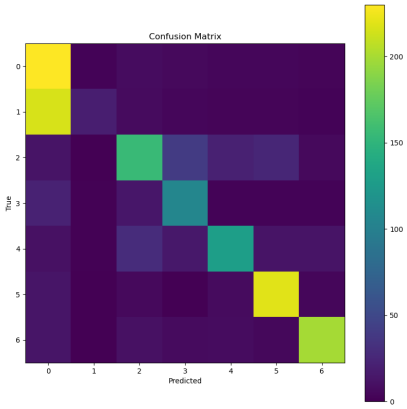


(a) Train Image

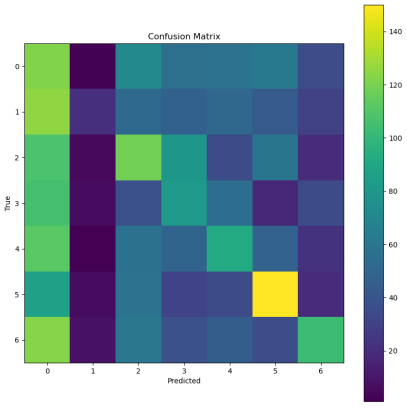


(b) Test Image

Figure 2: Class Distribution



(a) Train Confusion Matrix



(b) Test Confusion Matrix

Figure 3: Confusion Matrix

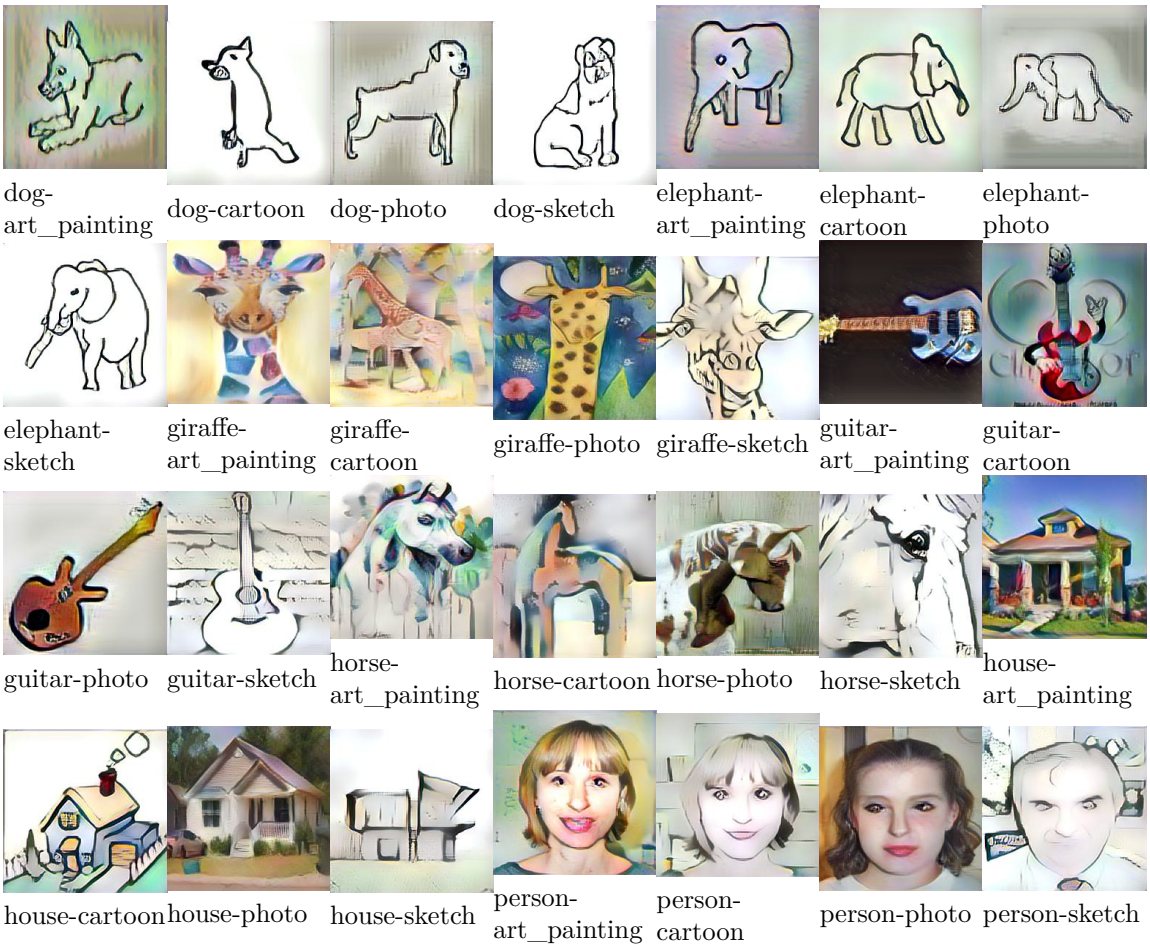
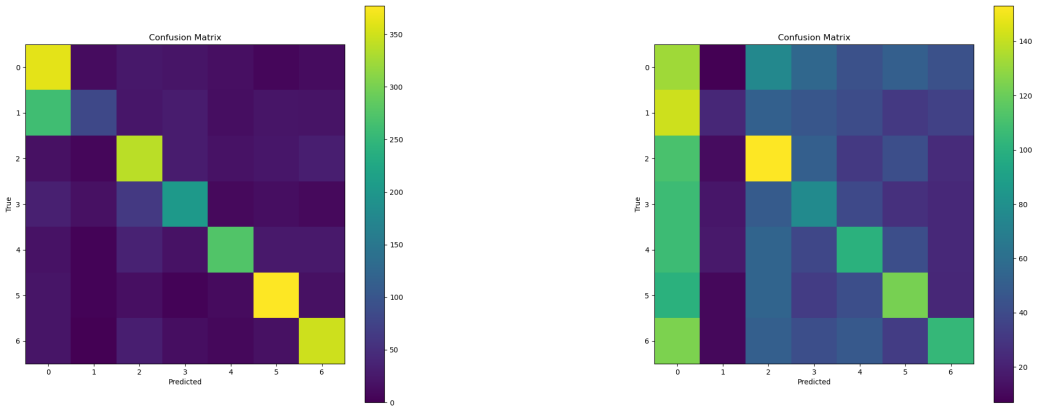


Figure 5: Example of images with labels and styles.



(a) Train Confusion Matrix

(b) Test Confusion Matrix

Figure 6: Confusion Matrix