Research on the Expanded Night Road Condition Dataset Based on the Improved CycleGAN

Lei Cao

School of Computer Science and Engineering

Xi'an Technological University

Xi'an, 710021, China

E-mail: 2443816614@qq.com

Li Zhao

School of Computer Science and Engineering

Xi'an Technological University

Xi'an, 710021, China

E-mail: 332099732@qq.com

Abstract—**Image style transfer is a major area of study in image processing and has applications in creative production, special effects for film and television, and other areas. Image style transfer is the process of using style transfer technology to change a common image into one with a different style without changing the content. Image style transfer methods are mainly divided into traditional image style transfer methods and deep learning image style transfer methods. The two primary classifications of picture style transfer techniques are deep learning technologies and conventional methods. Traditional image style transfer methods have poor results and are difficult to apply in people's lives. With the quick advancements in machine learning, digital image processing, and computer vision, deep learning image style transfer methods have received widespread attention from researchers. Most of these methods use convolutional neural networks to achieve image style transfer on the premise of paired data sets, but obtaining paired data sets is difficult and costly. Accordingly, it is of great significance to study unpaired images to implement style transfer algorithms. The primary focus of this study is the CycleGAN network-based picture style transfer technique, and improves this algorithm in content compiler, style compiler. It is applied to the generation of night road conditions during autonomous driving training.**

Keywords-Generative; Adversarial Networks; Style Migration; Deep Learning

I INTRODUCTION

In recent years, generative adversarial network-based image style migration methods have advanced significantly, but many image style migration algorithms can only complete data that match each other, but it's not possible that all the training sets are complete pairs, which increases the difficulty of obtaining them, and to a certain extent limits the application of the model. Naturally there are inconveniences and struggling in some topics. CycleGAN model is a general framework to solve different types of image migration without matching data, and its task is to firstly transform different image domains by learning the correspondence between the real image domain and the art style domain. [1] The implication of this correspondence is that the generator can transform an image into an image in the artistic style domain. In a similar vein, CycleGAN [2] must translate images from the Y domain to the X domain during training, and the process is similar to the process described above, only exchanging X and Y letters. Such a cycle forms a cyclic network. It is on the basis of the Cycle GAN model that the research content of the text is built, and through the understanding and study of the CycleGAN model, it is improved upon and extended in the application of migration of artistic image styles.

In generative adversarial networks, most of the generative networks are encoded and then decoded, whereas encoding uses encoders to convert the low dimensional spatial properties of the input into high dimensional spatial properties and decoders are used to decode and output the high dimensional spatial properties. [3] While the picture change is carried out in the decoding phase, the content features of the initial picture are preserved to the greatest extent possible throughout the migration process.

The proposal of Cyc1eGAN introduces a new loss, which is similar to what we call content loss in the style transfer process. To the extent that it limits the generator, it's the loss of cycle consistency. When the cycle consistency loss does

not match the actual image of the target area, the algorithm will restrict the content of the model and find the corresponding relationship in the two unmatched data sets.

II   NETWORK ARCHITECTURE MODEL

The image style transfer algorithm based on Cyc1eGAN proposed in this article. In the generative adversarial network, most of the generating networks are encoded first and then decoded. The encoding uses the encoder to convert the input low-dimensional space characteristics into high-dimensional ones. Spatial characteristics, and the decoder uses the decoder to decode and output high-dimensional spatial characteristics. During the migration process, try to keep up the writing characteristics of the initial picture, and perform image conversion in the decoding part. To make things easier to understand, we call the network that is used for encoding the encoder and the network that is used for decoding the generator. It uses residual networks to fuse images and achieve image style conversion. On this basis, a multi-scale discrimination method is adopted to increase the image style transfer's correctness.

The training of GAN is optimized alternately, and the discriminative and generative models are interleaved. The first is to keep the generated model unchanged and improve the accuracy of the discriminant model. After completion, we leave the discriminative model unchanged to increase the probability that the generative model generates real data. After training to a certain extent, the generative model produces samples that are quite similar to the original samples.

At this time, relying on the discriminant model to determine will cause difficulties and it will not be so accurate. It will enter an equilibrium state and the training will end.

The CycleGAN network can transform the style between two data sets, and its network training is bidirectional. At different network levels, the retrieved photos' content and style also differ. Content characteristics are extracted using the autoencoder, while style features are extracted using the variational autoencoder.
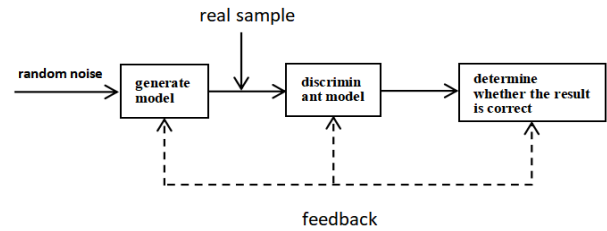

Figure 1.   Example of a ONE-COLUMN figure caption.

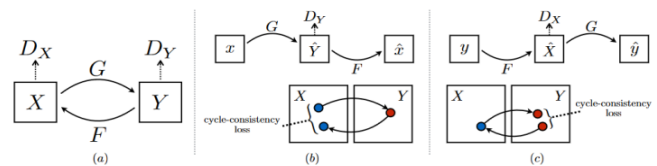Next, the image's content and style components are retrieved, and lastly, a style-transferred image is produced.

Figure 2.   CycleGAN Network structure diagram

A.   Encoder structure

1)   Content coding:

The content encoder uses the autoencoder network model. Autoencoder technique [4] is a data compression-based dimensionality reduction algorithm which is a new learning method for data compression and decompression based on data. Its function is to compress the data. In this paper, only the coding component of the autoencoder is used to extract the image. The generative adversarial network's generator will perform the function of generating the image.

The content encoders in the network structure of this document consist mainly of the residual network. Deep residual network is a very effective method in picture categorization, target location and detection.

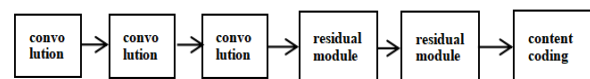The content encoder consists of 3 convolutional layers and 2 residual modules.

Figure 3.   Content encoder structure diagram

2)   How the content encoder works:

a)   The pixels of the input image are 256 x 256, and a down sampling operation is performed on it.

The first layer: 64 7 x 7 convolution kernels, sliding step size: 1, padding size: 3, and instance normalization processing, and then perform Relu activation processing.

b) The second and third layers: 128 convolution kernels of 4 x 4 size, sliding step size: 2, padding size: 1, and the subsequent steps are the same as the first step.

c) There are two convolutional layers with 256 3 x 3 convolution kernels in the residual module. Both the cushioning and the sliding step sizes are one. The subsequent steps are the same as the first step.

3) Style coding

The variational self-encoder network model is used in this paper's network structure. The variational auto-encoder needs to generate implicit vectors conforming to a Gaussian distribution during the encoding process, a constraint that enables the training data's ground rules to be understood by the encoder so that it can learn the implicit variables of the input data. With the training data variational autoencoders figure out the parameters' probability distribution. Variational autocoding techniques are used to encode image formats to give them greater randomness.

Variational autoencoder is implemented by converting the output of the encoder into two vectors corresponding to the vectors representing the standard deviation and mean. Using the mean and standard deviation vectors, an implicit variational model can be constructed to derive the coding vectors.
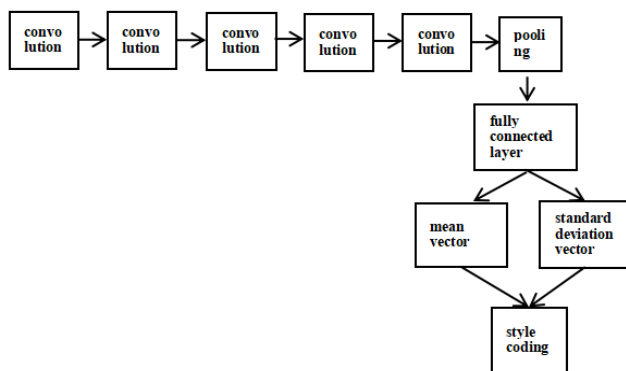

Figure 4.   Style encoder structure diagram

Five convolutional layers, one pooling layer, and one fully linked layer make up the style encoder.

B.   Generative network structure

In generative networks, the residual structural unit normalization [5] used is an algorithm based on adaptive normalization that speeds up the stylization of an image while maintaining its style. Here, we first input the content and style data at the same time, then go through the four residual modules, and finally the style transfer is completed by the up-sampling and convolution operation.

Neural networks for different tasks often choose different normalization functions, and different normalization functions will have different effects on the final results. Compared with the original GAN model, the CycleGAN model replaces batch normalization with instance normalization in order to make the model more stable.

CycleGAN uses instance normalization to perform separate normalization processing on each image, making the content information between each image independent of each other, avoiding the mutual influence between images and resulting in blurred generated images. Some progress has been made using instance normalization, but this method only works better in image style conversion when there are small differences in the shape and texture of the image domain. In aspects such as face style conversion, which have large differences, the effect is not very ideal. Instance normalization operates on each image and normalizes the H and W dimensions on a single channel, so in IN, different channels of different features are irrelevant, which allows the content feature information to be transferred to the style feature When applied to information, good results will always be achieved, but if it is only applied to a single channel, it will interfere with the original semantic information. Layer normalization acts on all channels. Layer normalization normalizes the C, H, and W dimensions of each image. It considers more global information, so some detailed information is often ignored. During training, the parameters of adaptive layer instance normalization (AdaLIN) [11], can be learned from the data set by adaptively selecting the proportion between instance and layer

normalization. Adaptive layer instance normalization combines the two, offsets their shortcomings, and combines the advantages of both. The specific formula is as follows:

$$\hat{\partial}_I = \frac{\partial - \mu_I}{\sqrt{\sigma_I^2 + \varepsilon}}, \hat{\partial}_L = \frac{\partial - \mu_L}{\sqrt{\sigma_L^2 + \varepsilon}} \qquad (1)$$

$$AdaLIN(\partial, \gamma, \beta) = \gamma.(\rho.\hat{\partial}_I + (1-\rho).\hat{\partial}_L) + \beta \qquad (2)$$

The adaptive layer instance normalization method is introduced into the decoder of the generator network, and combined with the self-attention mechanism, important local feature information and global feature information are automatically learned during network training.
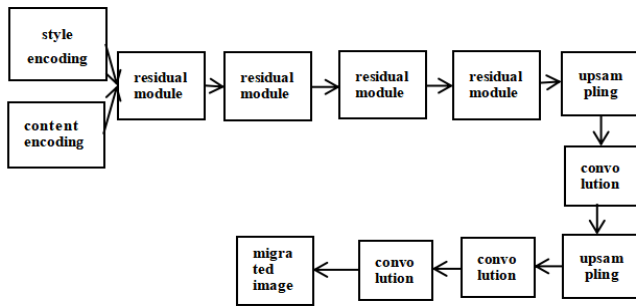

Figure 5.   Structure of the generative network

The migrating image is the result of the generative network, which takes style and content elements as inputs.

C.   Discriminative network design

Convolutional neural network [6] is still used in feature extraction of the whole image, and then recognition is performed based on the extracted features. Nevertheless, this approach frequently overlooks the image features, leading to issues like detail loss and image blurring. Therefore, we use a multi-scale discriminative network, which is an improved chunked discriminative network.
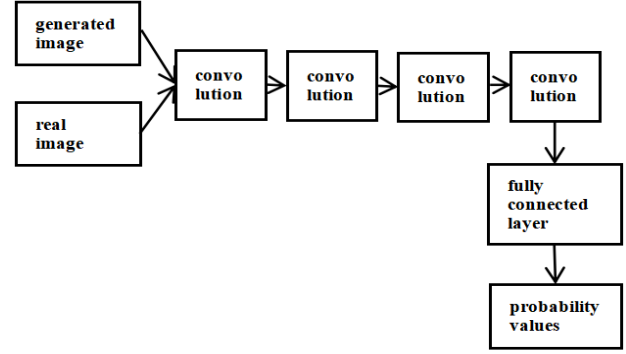

Figure 6.   Diagram of the structure of the discriminating network

The PatchGAN discriminator is essentially a multi-layer convolutional network. The input image of the network undergoes multi-layer feature extraction and finally outputs a 30*30 feature map. The network also divides the input image into 30*30 patches. The pixels in the feature map match the input image patch, and the value in the feature map is between 0 and 1, which is employed to convey how close the patch component image is to the original image.

Finally, to show how similar the complete image is to the original image, the scores of each patch are put together and averaged. The PatchGAN discriminator can identify patches that are significantly different from other patches in image information and give the corresponding patch a lower discriminant score. This way, the local information of the generated image can be consistent with the overall information, which significantly lowers the expense of producing the picture mistake. In addition, because the local image and the overall image are closely related, the patchGAN network based on local image area discrimination plays a positive role in the fusion of the overall information and local information of the image.

The discriminative network has one fully connected layer and four convolutional layers making up its network structure.

III LOSS FUNCTION DESIGN

A.   Adversarial Loss

Adversarial loss [7] is used to bring the generated image closer to the actual target style

image. Equation displays the adversarial loss computation formula.

$$L^{y_1}{}_{GAN} = E_{c_1-p(c_1),s_2-q(s_2)}[\log(1-D_1(G_1(c_1,s_2)))] \\ + E_{y_1-p(Y),s_2-q(s_2)}[\log D_1(y_1)] \quad (3)$$

$$L^{x_1}{}_{GAN} = E_{c_2-p(c_2),s_1-q(s_1)}[\log(1-D_2(G_2(c_2,s_1)))] \\ + E_{x_1-p(X),s_2-q(s_2)}[\log D_2(x_1)] \quad (4)$$

$G_1(c_1,s_2)$ denotes the output migration image, and $D_1(G_1(c_1,s_2))$, $D_1(y_1)$ are the discrimination results, respectively.

B. Image Reconstruction Loss

In the article, the generative network is fed both the style and content elements that were taken out of the style and content encoders in order to improve the capability of generative network in generative adversarial network.

The model is characterized by clearer and richer images clearly state the units for each quantity that you use in an equation.

C. Content encoding loss

Because a content encoder is included, the content coding loss is added to the loss function. The article proposes a method for style transfer using mismatched data and content feature extraction of the image using the content encoder, which ensures the invariance of the image during the style transformation. The style-shifted image [8] is re-inputted into the content encoder Ec, and as much as feasible, the content information of the picture is passed through the content information of the image that is output by the content encoder c so that the content information obtained has the same degree as the actual image described in order to train the content encoder.

D. Style coding loss

The style encoding loss is added to the loss function since the style encoder is now included. In order to ensure that the style converted image maintains the same style as the target image, the modified image is once again entered into the style encoder Es, and the extracted image style information must be as similar as possible to the style information outputted via the style encoder, so as to achieve the training of the style encoder.

E. Loss of cyclic consistency

CycleGAN network model, both to stylize the image and to recover the transformed image again, so the cyclic consistency loss of the image needs to be considered [9].

$$L^{y_1}{}_{cyc} = E_{y_1-p(Y)}[\|G_1(G_2(y_1))-y_1\|_1] \quad (5)$$

$$L^{x_1}{}_{cyc} = E_{x_1-p(X)}[\|G_2(G_1(x_1))-x_1\|_1] \quad (6)$$

IV ANALYSIS OF IMAGE STYLE MIGRATION RESULTS

A. Training and Testing

The training process of the network optimizes the parameters using Adam's algorithm [10], this uses adaptive low-order moment estimation as the basis for a one-time gradient optimization of any objective function.

It mainly contains the following features:

1) *Straightforward realization*
2) *Efficient Computing*
3) *Less memory usage*
4) *Gradient diagonal scaling's invariance*
5) *Appropriate for addressing optimization issues with a lot of data and parameters*
6) *Suitable for non-stationary objectives*
7) *Ideal for resolving issues with sparse gradients or extremely high noise*
8) *Hyper-parameters essentially just require a very tiny amount of intuition to interpret.*

We will train the content encoder, style encoder, generator, and input the content and style images to the appropriate content and style encoders for feature extraction during the project's testing phase, and then input the extracted content from the encoder to the generator to finally produce the style migration image on the generator.

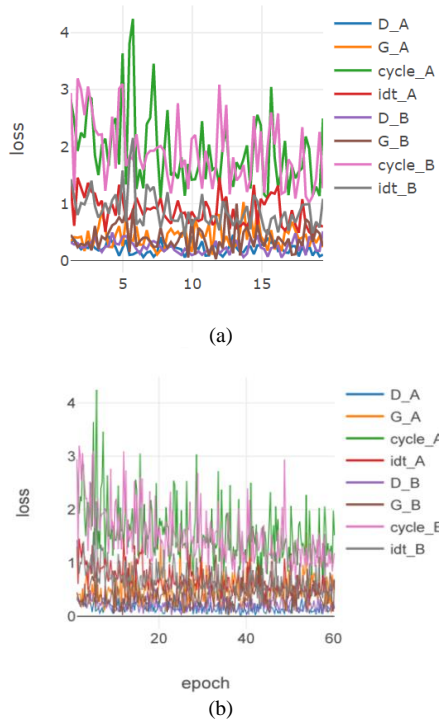Below are some loss function images from the training:

(a)

(b)

Figure 7.   Image of the loss function

Specific values for some of the losses are listed in Table I:

TABLE I.   TABLE OF INDIVIDUAL LOSS VALUES

epoch	D_A	G_A	cycle _A	idt_ A	D_B	G_B	cycle _B	idt_ B
epoch:1	0.31	0.34	2.83	1.31	0.32	0.45	2.94	1.43
epoch:1	0.37	0.41	2.46	0.61	0.28	0.36	1.23	1.09
epoch:1	0.28	0.40	1.93	1.46	0.31	0.23	3.20	0.80
epoch:1	0.20	0.42	2.24	1.30	0.23	0.29	3.00	1.01
epoch:2	0.24	0.17	2.16	1.10	0.23	0.27	2.55	0.99
epoch:2	0.22	0.52	2.89	1.37	0.27	0.32	3.06	1.16
epoch:2	0.29	0.20	2.85	1.21	0.32	0.18	2.72	1.40
epoch:2	0.22	0.41	1.85	1.18	0.42	0.61	2.30	0.96
epoch:3	0.18	0.57	1.50	0.96	0.23	0.20	2.21	0.76
epoch:3	0.19	0.88	2.03	1.05	0.15	0.42	2.22	0.85
epoch:3	0.24	0.52	2.14	0.50	0.25	0.37	1.22	1.02
epoch:3	0.43	0.24	1.66	0.90	0.25	0.49	1.76	0.82
epoch:4	0.33	0.23	2.37	0.99	0.49	0.62	2.53	1.06
epoch:4	0.23	0.45	2.50	0.79	0.30	0.66	1.61	1.16
epoch:4	0.08	0.60	1.69	1.33	0.28	0.76	3.02	0.80
epoch:4	0.24	0.25	3.64	1.09	0.42	0.66	3.10	1.58
epoch:5	0.36	0.72	1.34	0.72	0.43	0.87	1.66	0.68
epoch:5	0.10	0.35	3.81	0.73	0.34	0.15	1.82	1.75
epoch:5	0.11	0.35	4.24	0.88	0.30	0.91	1.78	2.15
epoch:5	0.14	0.60	1.26	1.10	0.20	0.24	2.22	0.63
epoch:6	0.16	0.28	1.60	0.93	0.12	0.34	1.79	0.71
epoch:6	0.06	0.28	1.27	0.80	0.25	0.40	1.59	0.61
epoch:6	0.14	0.53	2.40	0.91	0.20	0.40	1.86	1.34
epoch:6	0.16	0.28	2.51	0.90	0.32	0.34	1.93	1.23
epoch:7	0.07	0.22	3.46	0.93	0.34	0.24	1.93	1.30
epoch:7	0.14	0.11	1.99	0.91	0.31	0.25	1.98	0.94
epoch:7	0.41	0.18	1.10	1.10	0.23	0.57	2.01	0.58
epoch:7	0.16	0.18	1.84	0.80	0.16	0.32	1.93	0.83
epoch:8	0.23	0.47	2.64	0.72	0.21	0.23	1.48	1.20
epoch:8	0.25	0.56	1.50	0.65	0.28	0.32	1.18	0.73
epoch:8	0.24	0.46	1.55	0.83	0.18	0.24	1.61	0.70
epoch:8	0.20	0.51	1.44	1.05	0.13	0.34	2.76	0.70
epoch:9	0.20	0.40	1.81	0.83	0.10	0.48	1.63	0.84
epoch:9	0.05	0.59	1.14	0.84	0.20	0.49	1.77	0.42
epoch:9	0.20	0.29	1.70	0.64	0.14	0.05	1.25	0.86
epoch:9	0.43	0.15	1.42	0.86	0.10	0.73	2.14	0.61
epoch:10	0.23	0.73	1.43	0.75	0.12	0.11	2.20	0.74
epoch:10	0.20	0.50	1.65	0.69	0.16	0.57	1.71	0.74
epoch:10	0.34	0.41	2.45	0.73	0.10	0.51	1.26	1.17
epoch:10	0.09	0.43	1.43	1.04	0.16	0.32	1.93	0.71

During the actual training process, the training results can be judged according to the loss value.

The smaller the value, the more successful the training is. Generally, the training results improve with a lesser decrease of D. As can be seen from Figure 7, as training continues, each loss value has a decreasing trend. It can also be observed during the training process that the effect of style transfer gradually becomes stronger as the epoch increases.

There are also some demonstrations of images during training, as in Figure 8:
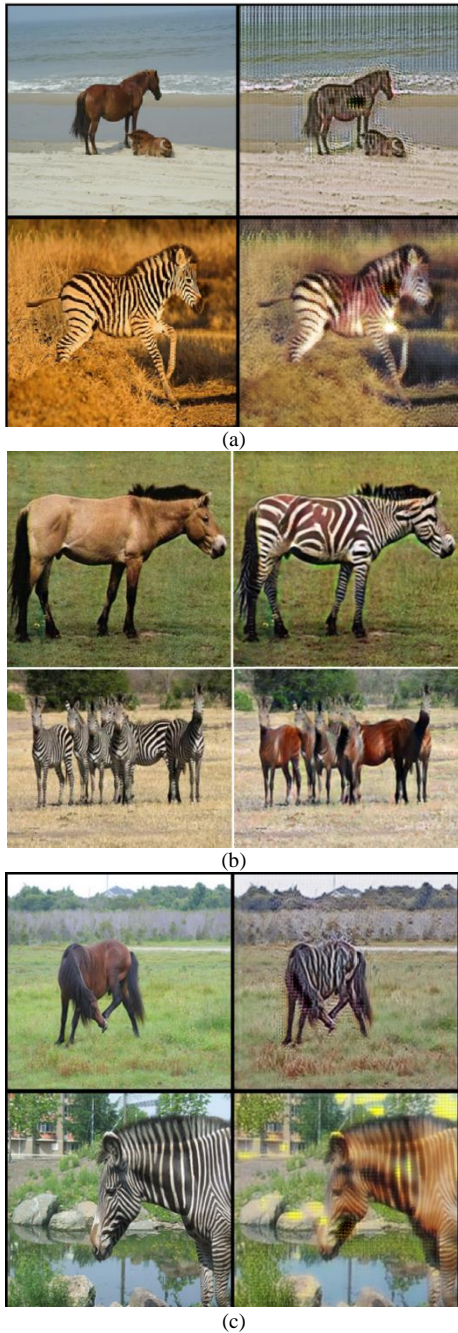

(a)

(b)

(c)

Figure 8.   Training presentation diagram

It can be seen that at the beginning of the training, the effect is not very obvious. As the number of iterations and training increases, the style migration can slowly be seen, and eventually the transformation is completely clear.

B. Experimental application and analysis

The topic of autonomous driving has advanced quickly in recent years, and the quality and quantity of datasets are crucial for the research in the domain of self-driving cars, and excellent datasets can help the autonomous.

Diving system better adapt to different lighting and environmental conditions, to enhance the self-driving system's functionality and safety at night and in low light conditions. At night or in low-light conditions, objects in images are often more difficult to identify and locate than during the day, which can lead to degraded performance of autonomous driving systems, increasing the risk of traffic accidents. Therefore, the dataset can be enriched and enhanced by using the method of daytime and nighttime road map style transfer, to enhance the capacity for generalization and the training effect of the autonomous driving system.

In this experiment, 1000 actual traffic maps were selected as the training set using the BDD100K dataset, including 500 road maps during the day and 500 road maps during the night period. 400 actual traffic maps were used as training sets, including 200 during the day and 200 at night.

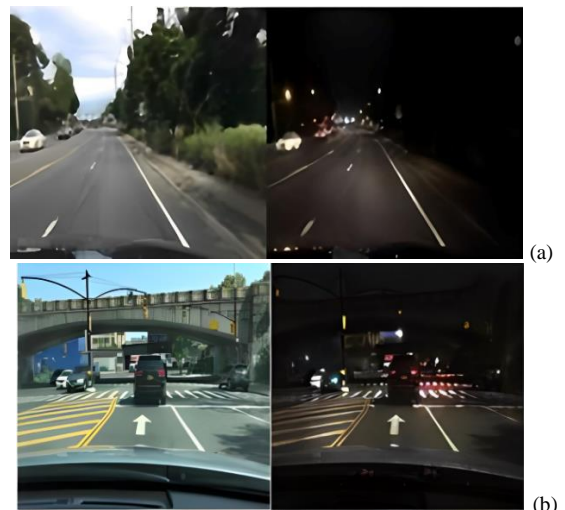The conversion effect is shown in Figure 9:

(a)

(b)

Figure 9.   Day and night conversion diagram

V   Conclusions

With deep learning technology developing continuously in recent years, there is a wave of new development trend within the domain of image processing, and image style migration is one of the research hotspots. The focus of this thesis is to implement this on the basis of generative adversarial networks. The article covers relevant techniques and effects as well as the current status of picture style migration development. The picture style migration method based on deep learning is explained.

Then it explains the corresponding theory and knowledge, convolutional neural networks, generative adversarial networks, and artificial neural networks. Emphasis is placed on the fundamental idea behind the generative adversarial network model, its training procedure, its style of derivation, and its application to the field of graphics transformation.

The generative adversarial network-based technique for image style transfer still has many issues that need further research. In network training, in order to lessen the difficulty of network training and increase the network's learning efficiency the model must be optimized and trained. The model-optimized offline picture style transfer approach has essentially achieved real-time performance in tests, but model learning still requires a significant amount of time.

Through image detail processing, the resultant style transfer images can be further enhanced. To provide a more realistic picture style transfer effect, the network's activities are refined further during the network design phase.

References

[1] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, WardeFarley D, Ozair S, Courville A, Bengio Y. Generative adversarial nets. In: Proceedings of the 2014 Conference on Advances in Neural Information Processing Systems 27. Montreal, Canada: Curran Associates, Inc., 2014. 2672−2680.

[2] Dong H, Neekhara P, Wu C, et al. Unsupervised image-to-image translation with generative adversarial networks [J].2017.

[3] Huang X, Liu M Y, Belongie S, et al. Multimodal unsupervised image-to-image translation[A].// Proceedings of the European Conference on Computer Vision [C]. 2018:172-189.

[4] Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation [J]. Springer, Cham, 2015.

[5] YU X, FATIH P. Imagining the Unimaginable Faces by Deconvolutional Networks [J]. IEEE Transactions on Image Processing, 2018:2747-2761.

[6] Odena A, Dumoulin M et al. "Deconvolution and Checkerboard Artifacts", Distill, 2016.http://doi. org/10.23915/distill.

[7] Gupta A, Zou J. Feedback GAN for DNA Optimizes Protein Functions [J]. Nature Machine Intelligence, 2019, 1(2): 105-111.

[8] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generat ive adversarial networks [J]. Communications of the AC M, 2020, 63(11): 139-144.

[9] Liang X, Chen L, Nguyen D, et al. Generating synthesized computed tomography (CT) from conebeam computed tomography (CBCT) using CycleGAN for adaptive radiation therapy [J]. Physics in Medicine & Biology, 2019, 64(12): 125002.

[10] Zhu J-Y, Park T, Isola P, et al. Unpaired image-to-image translation using cycle consistent adversarial networks [C]//IEEE international conference on computer vision, 2017: 2223-2232.

[11] Gulrajani I, Ahmed F, Arjovsky M, et al. Improved Training of Wasserstein GANs [C]// Advances in Neural Information Processing Systems, Long beach, USA,2017: 5769-5779.