# CSCI 1430 Final Project Report: Your project title

CV Addicting Bot: Jiaxin Liu, Wandong Yan, Yihao Zhou.
Brown University

## Abstract

Style transfer refers to keeping the content of the original image and transforming the style of the original image into the target style. The CycleGAN we implemented in this project proposes a combination of cyclic consistency loss and GAN's adversarial loss, which can learn to transform the image from the source domain through unsupervised training without the need for paired image data sets. Is the mapping relationship of the target domain. We implemented it on kaggle's public data set and the data set is available at https://www.kaggle.com/suyashdamle/cyclegan.

## 1. Proposal Notes

We reproduced the CycleGAN paper in this project, and applied the model to the public data set obtained from kaggle, realizing the style transfer of the painting styles of Monet, Van Gogh, Ukiyoe and Cezanne. After training for multiple epochs, our model has achieved good results. In addition, in the process of completing this project, we also conducted related research on the origin and development of image style transfer, and gained a deeper understanding of the field of image style transfer.

## 2. Introduction

The so-called image style transfer is to provide a picture, convert any photo into this style, and try to preserve the original content. The CycleGAN model we implemented in this project understands the problem of image style transfer as a broader image-to-image translation problem, and completes the conversion of an image from a given scene to another scene. CycleGAN learns the mapping relationship from the input image to the output image from the data set. Due to the difficulty of obtaining the paired data set, CycleGAN adds a cyclic consistency loss function to the basic GAN loss function: the adversarial loss function to ensure the cyclic stability of the translation problem.

## 3. Related Work

### 3.1. Style Transfer

Style transfer refers to providing two pictures: the original picture and the target picture, and output a new picture that retains the content of the original picture but the style is converted to the target picture style. Before neural networks, the image style transfer problem solving program had a common idea: analyze a certain style of image, build a mathematical or statistical model for the target style, and then change the distribution of the original image to be transferred to make it more It conforms well to the established style model. This traditional method can achieve good results, but it also has a big disadvantage: the problem-solving program designed by the traditional method can basically only transfer the style of a certain style or a certain scene. Therefore, the actual application scenarios of style transfer research based on traditional methods are very limited. Style transfer based on deep learning was first proposed by Gatys in 2015. Gatys found that texture can describe the style of an image. And in the paper [2], he used the gram matrix to calculate the statistical model to extract the image style. Later, in the paper [3], Gatys directly regarded the local features extracted by the neural network as approximate image content, and used the correlation directly pressed by the feature map to represent the texture model. By separating and recombining the image content and style, CNN It is possible to create works that conform to the aesthetics of art, which proves the power of convolutional neural networks. The core principle of the pioneering CNN-based style transfer proposed by Gatys et al. is to perform feature fitting on image feature data, and use a pre-trained VGG model as an image feature extractor to explicitly separate the abstract features of image content and style. Representation, by independently processing these feature representations to generate stylized images with original image content

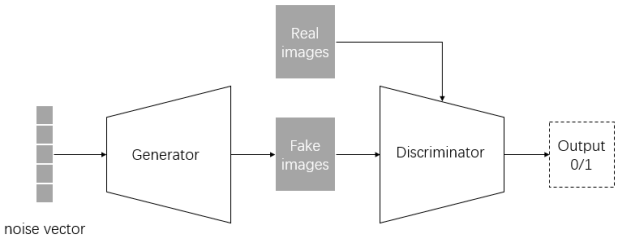



Figure 1. The simple structure of GAN.

and new styles. Neural style transfer exhibits a very good visual effect.

### 3.2. Adversarial Generative Network

Adversarial Generative Network has excellent effects in image generation, representation learning and other fields. As shown in Figure 1, the confrontation generation network includes a discriminator component $D$ and a generator component $G$. The generator $G$ receives the noise input $z$ and outputs the generated image $\hat{x}$. The processor discriminates the generated image $\hat{x}$. GAN encourages the discriminator and the generator to continuously play games during the training process by fighting the loss, and finally guides the generator to generate a generated image that cannot be distinguished from the real image in principle. The adversarial loss function is defined by the equation 1. The first part of the adversarial loss function is $\max_D$, because the generator $G$ is generally kept unchanged during training, and then the discriminator $D$ is trained. The goal of training the discriminator $D$ is to allow it to correctly distinguish the input real image or the generated image. Assuming that 1/0 represents the discrimination label true/false, then for the first item $\mathbb{E}$, since we expect $D(x)$ to approach 1, that is, $D_G^* = \arg\max_D V(G, D)$. In the same way, the input of the second item $\mathbb{E}$ comes from the generated samples of the generator $G$, so we hope that $D(G(z))$ will be close to 0, that is, we hope that the second item will be more big. The second part of the loss function is the fixed network $D$ training the network $G$. At this time, only the second term of the loss function plays a role. Since we hope that the generator $G$ can successfully confuse the discriminator $D$, it is better to expect the output of $D(G(z))$ to be close to 1, that is, $G^* = \arg\min_G V(G, D^*)$.

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(X)))] \quad (1)$$

In GANs, image style transfer is an image-to-image translation problem. Through adversarial training to determine the spatial distribution of image data, images in one domain can be converted to another domain. The pix2pix model proposed by Isola et al.[6], as a representative work of image-to-image translation, uses a large number of paired images for supervised training to obtain a one-to-one image translation network, which can perform the task of image style transfer excellently. Although pix2pix can achieve realistic image conversion, the training of this model requires a large amount of paired image data, which greatly limits its promotion and application. In order to better learn artistic styles, Sanakoyeu et al.[10] introduced a style-aware content loss in GAN, which can learn the same type of artistic style and is not limited to one instance in one style. Ma et al.[9] observed the feature vectors of content images and style images in the projection space, and found that the features of content images and style images in the initial state are basically separable. The double consistency loss they proposed can maintain semantics and style. In the case of consistency, learn the relationship between the content image and the style image. Also focusing on image content and style perception stylization, Kotovenko et al.[7] designed a content conversion module in the adversarial network to learn how to change the style transfer process between content and style images with similar content information. The details of the content. Choi et al.[1] proposed the StarGAN model, which can realize one-to-many image style conversion in a single GAN. In the article, the face image is taken as an example to realize the conversion of multiple facial expressions. The GAN-based image style transfer method brings a new approach to style description through the mechanism of adversarial learning. In GAN, there is no need for any pre-designed description calculation style, the discriminator can implicitly calculate the style by fitting the image data distribution, and realize the style transfer of the image. Fitting the distribution of image data through adversarial training can make the style transfer effect of the image more realistic, which reflects the GAN's ability to understand and perceive image data. Compared with the CNN-based style transfer method, GAN has better quality on the generated images, but the controllability of the style transfer process is not high, and the training of the adversarial network is prone to gradient disappearance and model collapse, which has the disadvantage of difficulty in training.

## 4. Method

### 4.1. CycleGAN

CyclenGAN[11] transforms the style transfer problem into an image-to-image translation problem, and learns the mapping between the input image and the

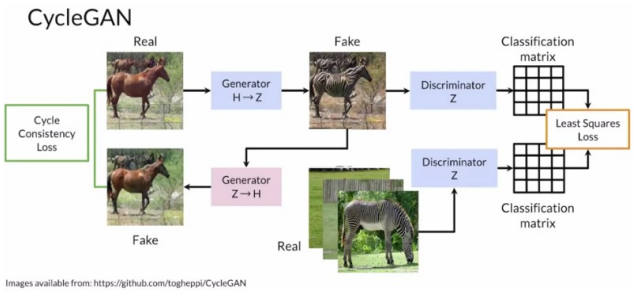



Figure 2. The structure of CycleGAN.

output image without paired data. CycleGAN has an important assumption that there is a potential connection between the source domain and the target domain, and this connection can be learned. In order to use the training data set for supervised learning, CycleGAN proposed a method of learning to transform the image from the source domain to the target domain without paired images. By learning the mapping from the source domain to the target domain $G : X \rightarrow Y$ To capture the unique features on the target image collection. The overall model structure of CycleGAN is shown in Figure 2. First of all, CycleGAN is based on GAN, including a generator $G$ and a discriminator $D$, and uses the adversarial loss min max in the network training iteration process to instruct the generator $G$ to generate a discriminator that cannot be discriminated The image on the target domain that $D$ cannot distinguish. However, even if the optimal generator $G$ can transform the image $x$ from the source domain to the target domain, it cannot guarantee the difference between the input image $x$ and the output image $\hat{y} = G(x)$ The matching between the two is meaningful, that is, it is satisfied that the output image $\hat{y}$ retains the content of the input image $x$ and has the style of the target domain at the same time, because there are multiple mapping networks $G$ that can make the input as $For x$, output the same distribution of $\hat{y}$. In addition, it is well known that the training process of GAN is extremely unstable, and it is easy for the model collaspse to generate a single mode result or the model does not converge or converge slowly, which makes the network optimization unable to continue. In order to solve the problem of difficulty in GAN training, CycleGAN uses the property of translation should be “cycle consistent”, that is to say, when the original image $x$ is mapped to the image on the target domain, $\hat{y}$ , And then map $\hat{y}$ to the source domain, the image of the mapping result should be consistent with the original image $x$. Therefore, CycleGAN adds an additional mapping network $F : Y \rightarrow X$, then the mapping networks $G$ and $F$ are opposite to each other. In the paper, this pair of mappings is called ”bijections”. At the same time, in order to enable the increased mapping network $F$ to be synchronized with the original GAN, CycleGAN also uses cycle consistency loss to know the network parameter iteration during the network training process, so that the model can finally reach $F(G(X)) \approx x$ and $G(F(y)) \approx y$, which is also the source of the name CycleGAN. Cycle consistency loss is defined by the equation 2, including forward cycle loss and backward cycle loss. The forward cycle loss calculation mapping process $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$ guarantees forward cycle consistency, while the backward cycle loss calculates the mapping process $y \rightarrow F(Y) \rightarrow G(F(y)) \approx y$ to ensure backward cycle consistency.

$$\begin{aligned}\mathcal{L}_{cyc}(G, F) =& \mathbb{E}_{x \sim p_{data}(x)}[\|F(G(x)) - x\|_1] + \\ & \mathbb{E}_{y \sim p_{data}(y)}[\|G(F(y)) - y\|_1]\end{aligned} \tag{2}$$

The total loss function of CycleGAN is

$$\begin{aligned}\mathcal{L}(G, F, D_X, D_Y) =& \mathcal{L}_{GAN}(G, D_Y, X, Y) \\ & + \mathcal{L}_{GAN}(F, D_X, Y, X) + \lambda \mathcal{L}_{cyc}(G, F)\end{aligned} \tag{3}$$

($\lambda$ are hyperparameters), in other words, our goal is to find the optimal mapping The networks $G$ and $F$ make $G^*, F^* = \arg\min_{G,F} \max_{D_X, D_Y} \mathcal{L}(G, F, D_X, D_Y)$. The experiment proves that the combination of adversarial loss function and cycle consistency loss can enable CycleGAN to better learn the conversion between the source domain and the target domain.

## 4.2. Implementation of CycleGAN

### 4.2.1 The Generator of CycleGAN

CycleGAN’s generator is composed of encoder, converter and decoder, as shown in Figure 3(Left). The encoder consists of three convolutional layers. Assume the input picture size (256, 256, 3). The first step is to extract features from the image through the convolutional layer. The number of features extracted from the convolutional layer can also be seen as the number of different filters used to extract different features. The convolutional layer gradually extracts more advanced features. After passing through the encoder, the input image changes from (256,256,3) to output (64,64,256). The converter consists of 9 residual blocks. Figure 3 (right) shows the structure of the residual block. The different channels of the image output by the encoder combine different features of the image, and the feature vector of the image, that is, the encoding, is converted from the source domain to the target domain according to these features. The residual block is composed of two convolutional layers, and the input residual is

added to the output. This is to ensure that the input attributes of the previous layer can also be used for the subsequent layers, so that their output will not be different from the original input Too much deviation, otherwise the features of the original image will not be retained in the output. The function of the decoder is to reconstruct the low-level features from the feature vector, which can be completed by the deconvolution (transposed convolution) layer. Finally, the low-level functions are converted into images in the target domain.

#### 4.2.2 The Discriminator of CycleGAN

The discriminator is composed of multiple convolutional layers. After extracting features from the image, it is judged whether these features belong to a specific category. The last layer of the discriminator network is the convolutional layer used to generate one-dimensional output. The discriminator takes the image as input and predicts whether the image is the original image or the output of the generator.

## 5. Experiment Results

In the experiment, we first scaled the picture to size = 256, then used a horizontal flip transformation with a probability of 0.5, and regularized each output picture with a mean value of 0.5 and a variance of 0.5 to speed up the model convergence. After setting batch size=3 and learning rate 1e-5, we trained for 120 epochs, and now the model works well to transfer photos to paintings of Van Gogh and Monet. The results shown in Figure 5, from top to bottom, each row shows the experimental input images and result images in cezanne, monet, ukiyoe and vangogh styles.

## 6. Comparison with existing results

Compared with the limitation that traditional methods are only applicable to a certain style or scene, CycleGAN based on neural network can realize the transfer between different content images and styles. At the same time, CycleGAN is also different from the pix2pix model, which requires a large number of paired images for training. It uses unsupervised adversarial training and introduces cycle consistency to eliminate the pairing constraints between the source and target domains, and it can better preserve images Loss of content. However, it is precisely because of the loss of cycle consistency and the pixel-by-pixel difference of image controls as the loss of image content, that the content confidence of the images generated by CycleGAN is over-reserved, resulting in the inability to transfer the abstract artistic style well. The experimental comparison of several representative works in the image style transfer method based on deep learning is shown in Figure 6. Figure 6 shows different style transfer image pairs from top to bottom, and from left to right are the experimental results of style images, content images, and five image style transfer methods. Among them, groups A, B, C, D, and E correspond to CNN-based Gatys et al.[4], WCT[8] and AdaIN[5] methods, and GAN-based CycleGAN and Sanakoyeu et al.'s methods, respectively. From a subjective point of view, the A, B, and C groups transfer the texture and color of the style image to the generated image very well. The overall transfer quality is high and the visual effect is better. Groups D and E focus on portraying the details of style images and content images, and the details of content images are preserved, making the generated images more authentic.

## 7. Technical Discussion

The CycleGAN we implemented introduces cyclic consistency to eliminate the pairing constraints between the source domain and the target domain, so that the model can achieve different image content and styles through unsupervised adversarial training even when there is no paired image data set. Migration. When training the network, we need to weigh the proportion of the content and style in the generated image. When the selected content accounts for more, the network will learn to generate and retain more original image content, and the gap between the image style and the target style may be relatively large. When choosing a style to account for more, the network may change the content of the original image and the style of the generated image is closer to the target style. We can adjust the proportion of the two by adjusting the hyperparameters in the network loss function.

## 8. Societal Discussion

Please respond to the following questions. Different projects will have different scales and qualities of impact; we ask you to think creatively and consider the broader implications of your project rather than just the more narrow current technical capability. Responses should together take up roughly one page in your final report.

1. Describe the socio-historical context of your project to identify three broad societal factors that could affect your data, goal, and/or hypothesis. These factors might include current or historical policies, events, social conditions, and larger societal systems. Cite at least one outside source.

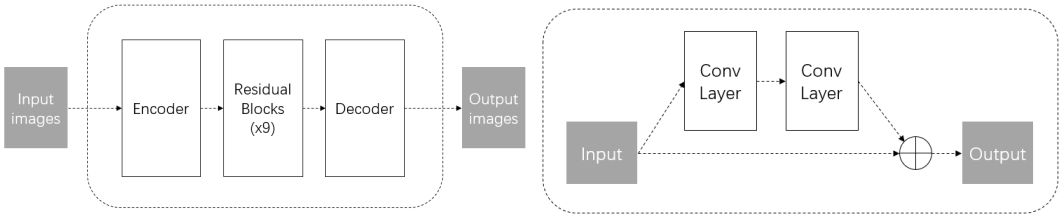



Figure 3. Left: The structure of generator. Right: The structure of residul block.

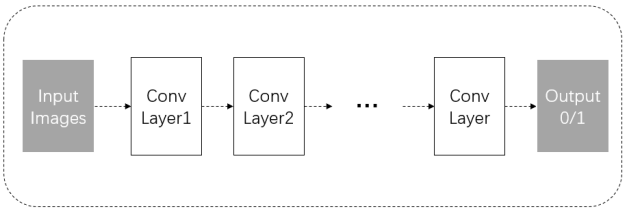
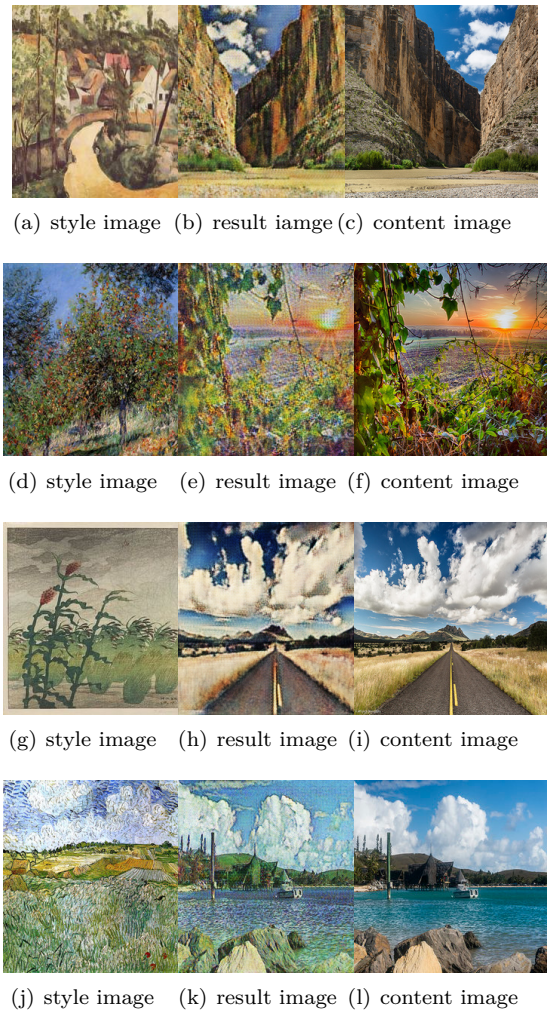



Figure 4. The structure of discriminator.

Answer: Style transfer is one of the more interesting applications of convolutional neural networks. It is a cross-collision between AI and traditional art. The use of neural networks for style transfer was first proposed by Gatys et al. in the paper Image Style Transfer Using Convolutional Neural Networks in 2015, and the name image stylized transfer has only been used since then. From 2015 to 2017, the novel field of image stylized migration has attracted many people's practical exploration. After years of rocket-like development, the deep learning model of style transfer has evolved from a single content single style through any content and multiple styles to any content and any style model. At present, style transfer is mainly used for entertainment. Developers have also developed many mature style transfer applications, such as Prisma. Prisma is very popular. It was rated as one of the best apps of the Apple Store and Google Store in 2016.

2. Who are the major stakeholders in this project? What is your relationship to these stakeholders?
Stakeholders are those who may be affected by or have an effect on your project topic. Some examples of stakeholders are a particular demographic group, residents of a particular geographic area, and people experiencing or at risk for a particular problem.
Consider the following questions to help identify stakeholders:

   - Who does this project topic currently affect?
   - Who might be harmed by your research findings?



(a) style image (b) result iamge (c) content image

(d) style image (e) result image (f) content image

(g) style image (h) result image (i) content image

(j) style image (k) result image (l) content image

Figure 5. Experiment Results Images

   - Who might benefit from your research findings?

Answer: Because of the amazing results of neural style transfer, it has also brought many successful industry applications and has begun to realize commercial returns. Developers make profits by putting software based on style transfer on the market. Users also get fun from the style transfer.

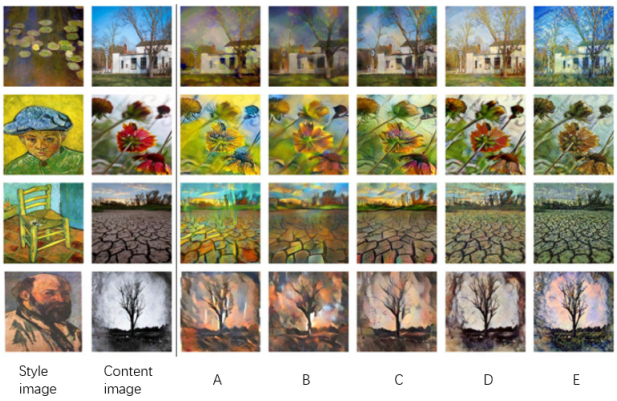



Figure 6. Experimental comparison chart of representative algorithms for image style transfer

Because of the magical artistic creativity of style transfer, artists have also begun to use style transfer to create. But this has also brought undesirable effects. Since the ownership of the results created by deep learning is unknown, when people use the resulting images generated by style transfer to obtain benefits, we cannot follow the existing laws on the legality of the benefits. Make judgments. At the same time, because the value of artwork itself is difficult to stock, the value of works created through the use of deep learning technology is also difficult to assess qualitatively, because it seems to have no soul. Of course, if it is only from the perspective of creation cost, the work is indeed more than a mention. Its cost is at best only the energy spent by the technicians when collecting the works of art, as well as the server costs spent in the model training process, and on the Internet There are a large number of ready-made open source algorithms available.

3. Research or journalism on your broader project topic may have already been conducted. What was the societal impact of existing research? Discuss the implication of this research on your project and consider the following questions to help identify at least one implication. It may affect:

   - How you should frame your goal,
   - How you should design your algorithm,
   - How you should analyze your data,
   - How you should interpret your findings, and
   - How you should present your results.

   Answer: Style transfer based on deep learning again applies AI to the field of art, bringing AI technology closer and closer to people's lives, and providing a new perspective on how to make AI more intelligent. But behind the brilliance of these technologies, every time AI has a new development in a different field, it will inevitably cause anxiety. In order to make sure that the creation of the network not only retains the original content of the original image (in a sense, it can be regarded as the soul of the work), but also has the creative style of the target style image, CycleGAN proposes cyclic consistency loss and adversarial loss. Knowing that the network generation is different Creative style, but new works with the same spiritual core.

4. How could an individual or particular community's civil rights or civil liberties (such as privacy) be affected by your project?

   Answer: The ease of use of style transfer makes creation easier, but if you use someone else's work as the input style picture or content picture, the ownership of the resulting image cannot be determined, which may damage the rights of the original author.

5. If you are using data, what kind of biases might this data contain? Do any of these represent underlying historical or societal biases? How can this bias be mitigated?
   Consider the following questions to help you:

   - Were the systems and processes used to collect the data biased against any groups?
   - Is the data being used in a manner agreed to by the individuals who provided the data?

   Answer: The data set we use is inevitably biased. However, when using the Internet's public data sets for style transfer experiments, generally speaking, it is easier to obtain a data set containing the works of creators or owners that are popular with the public, which also makes the Internet more effective in transferring styles of these people's works. Good, and for the realization of pictures such as niche creation styles, the model may have different degrees of algorithm discrimination. The bias of this kind of data set may be due to the fact that the creators themselves produce few works, or the style is not accepted by the public aesthetics, which is caused by long-term culture. The data set we used in this project is all agreed, because it comes from the open source network. It is very important to ensure that the data used has the consent of the original owner, because the use of data without the permission of others may violate the interests of others and violate relevant

laws. The requirement to use the agreed data set is also to better maintain the network environment.

## 9. Conclusion

Nowadays, the image style transfer algorithm based on deep learning has achieved Significant development, they can obtain satisfactory performance results, and have been promoted and applied in the industry. However, there are still some problems and challenges. For example, the scale of model parameters used for image style transfer is huge. The standard for effective evaluation of effects has not been determined, etc.

## References


[1] Yunjey Choi, Min-Je Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. CoRR, abs/1711.09020, 2017. 2

[2] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 28. Curran Associates, Inc., 2015. 1

[3] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style. CoRR, abs/1508.06576, 2015. 1

[4] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2414–2423, 2016. 4

[5] Xun Huang and Serge J. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. CoRR, abs/1703.06868, 2017. 4

[6] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. CoRR, abs/1611.07004, 2016. 2

[7] Dmytro Kotovenko, Artsiom Sanakoyeu, Pingchuan Ma, Sabine Lang, and Björn Ommer. A content transformation block for image style transfer. CoRR, abs/2003.08407, 2020. 2

[8] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. CoRR, abs/1705.08086, 2017. 4

[9] Zhuoqi Ma, Jie Li, Nannan Wang, and Xinbo Gao. Semantic-related image style transfer with dual-consistency loss. Neurocomputing, 406:135–149, 2020. 2

[10] Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, and Bjorn Ommer. A style-aware content loss for real-time hd style transfer. In proceedings of the European conference on computer vision (ECCV), pages 698–714, 2018. 2

[11] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. CoRR, abs/1703.10593, 2017. 2


## Appendix

### Team contributions

Jiaxin Liu Data preprocessing, Model implementation and training.

Wandong Yan Model exploration, implementation, training.

Yihao Zhou Data collection, Model implementation and training.