Design Studies and Intelligence Engineering
L.C. Jain et al. (Eds.)

doi:10.3233/FAIA220727

A Method of Coloring Sketch Images Based on Improving Generative Adversarial Networks

Xiaoyu Ma[a], Wei Li[a, 1], Zhiqiang Zeng[a], Shaoyong Yu[b], Shunyi Chen[c], Youmeng Luo[a], and Kaiqiang Zhang[a]

[a] *School of Computer and Information Engineering, Xiamen University of Technology, Xiamen, China;*

[b] *School of Mathematical and Information Engineering, Longyan University, Longyan, China;*

[c] *Center of Modern Engineering Training, Xiamen University of Technology, Xiamen, China;*

Abstract. Based on the generative adversarial network, this paper proposes a method of colorizing gray images without supervision and applies it to the field of the automatic coloring of gray comic sketches to solve the problem of the high production cost of color comics and the large consumption of human resources. The method proposed in this paper mainly improves the classical generative adversarial network model in the following aspects: first, the idea of the residual network is added to the builder model and the discriminator model to ensure that the training of the model can develop in the right direction; secondly, add a gradient penalty term to the loss function of the discriminator to accelerate the convergence of the model and speed up the iteration; at the same time, the activation function in the original model is changed to the Mish activation function, so that the information flowing through the network model has higher accuracy and better generalization. This paper trained the model of the Anime Sketch Colorization Pair dataset [1] from Kaggle, and the final experimental results show that the method is practical and feasible.

Keywords. Generative Adversarial Networks, Unsupervised, Residual Network, Gradient Punishment, Mish

1. Introduction

Grayscale image coloring is the process of adding color to a monochrome image. The original image shading techniques were mainly divided into two methods: color-based markers and color-based transfer. Among them, the colorization algorithm of grayscale image based on color marker was first proposed by Levin[2] et al, in 2004, and the implementation of the algorithm is mainly based on the theory that "space-time adjacent pixels of similar intensity should have similar colors" as the premise, and human participation is added, which is a semi-automatic colorization method. This requires a lot of manual color labeling, and when the edge area of the image is accidentally marked

[1] Corresponding Author: Wei Li, Email: drweili@hotmail.com

with the wrong color, the false spread of color will occur. Later, Chen[3]et al, proposed a colorization method based on segmentation technology, which first divides large-sized images into small-size images based on segmentation points, and then uses the relationship between the brightness information of small-size images and color space to calculate the linear interpolation of each adjacent small-size image, and finally uses the diffusion principle to obtain the chromaticity values of all pixels on the entire image. This colorization method can ensure the visual effect of the image, and also solves the disadvantage that the diffusion algorithm cannot handle large-size images.

The idea of colorization algorithm based on color transfer was first proposed by Reinhard [4] and improved by Welsh[5]et al, mainly by comparing the mean squared error of luminance and texture information between the reference image and the grayscale image to be colored, to continuously adjust the parameters to achieve the colorization of single-channel images. However, because the color change is not only the brightness and texture information, even if the brightness and texture information are consistent, the color image after coloring is still not very natural. Later, Irony[6]combined the method of color diffusion, assigning color to each pixel by segmenting the reference image and using neighborhood matching metrics, and finally using color diffusion to color the entire image to be painted. Liu[7]et al, grayscale multiple reference images, calculate their inherent reflectivity and illumination components, and then transfer the intrinsic reflectivity of the reference images to the image to be colored, and achieve colorization of the grayscale images by redistributing the illumination components. This algorithm needs to use computer retrieval technology to search for multiple images similar to the image to be colored as reference images, and to a certain extent, solve the color transfer error caused by the difference between the reference image and the target image is too large. However, when multiple reference images similar to the target image cannot be searched in the network, the final colorization results will not be satisfactory.

In recent years, with the continuous development of artificial intelligence technology, colorization methods based on machine learning have begun to be applied to the field of image shading. Cheng[8] et al, first used neural networks to color grayscale images, but due to the simple model structure and few training datasets, a large number of artifacts appeared in the generated images, and in order to solve this problem, the authors also used the post-processing method of bilateral filtering. Zhang[9]et al, proposed an image coloring method based on VGG neural network, which transforms the coloring problem into a classification task to solve the color information uncertainty of the image to be colored, and uses class equilibrium in the training of the network model to enhance the diversity of colors and improve the generalization ability of the network model. This method can achieve a reasonable colorization result for most images, but due to the lack of reference, the color of the image is too averaged and may not achieve the ideal effect of the user. The Neural Style Transfer[10] algorithm later proposed based on VGG19 is to mix content images and style images together. Designed to produce images that are similar to content images but follow the style of reference images, and show good results. Cao [11] et al, employ generative adversarial networks to color images directly, and try to optimize by introducing noise, introducing random noise variables multiple times to each layer of the convolutional layer to encourage the generator to produce random and diverse outputs. To a certain extent, this practice can achieve the diversification of the generated images, but at the same time, with the addition of strong noise in each layer, it will inevitably have an impact on the quality of the final image generation. Later, many good colorization algorithms emerged based on generative adversarial networks, such as DCGAN, cycleGAN, etc, which is still a hot topic in the field of computer vision for improving the research of generative adversarial networks and colorization algorithms.

The main contribution of this article is as follows:

- On the basis of generating the confrontation network model, by adding the idea of deep residual networks, the training of generating networks and judging networks are optimized.
- By adding gradient punishment items to the loss function, the training model is accelerated.
- By using the new Mish activation function, replace the previous RELU and other activation functions, realize the engineering application of the Mish function, and achieve the ideal effect.

2. Question

Given $X \in G^m$, X is the input image vector, G^m is a channel gray image vector set, Y is the corresponding output after the mapping, is a three-channel color vector, $Y \in C^n$, where C^n is a three-channel color image vector set, F For the mapping relationship of X to Y, there are:

$$Y_{out} = F(X_{in}) \tag{1}$$

In this way, every input X passes through F, there will be an output Y, which corresponds to it. The plan of this article is to design a mapping relationship such as F to achieve the output from X to Y.

3. Scheme

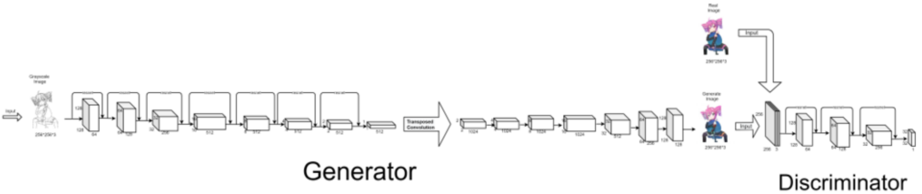

Figure 1. GAN model.

$$\min_G \max_D L(D,G) = E_{\hat{x}\sim P_r(x)}[logD(x)] + E_{\hat{z}\sim P_z(z)}\left[log\left(1 - D\big(G(z)\big)\right)\right]$$

$$= E_{\hat{x}\sim P_r(x)}[logD(x)] + E_{\hat{x}\sim p_g(x)}[log(1 - D(x))] \tag{2}$$

In this paper, using a GAN network model based on EM distance, the automatic coloring of gray sketch images is finally realized after training pairs of gray-color sketch images. The generative adversarial network contains a generator and a discriminator[12], in which the generator is mainly composed of an 8-layer convolutional neural network plus a 7-layer deconvolutional neural network, which ultimately generates a three-channel color image by feature extraction and deconvolution of the original single-channel grayscale image, while using the resnet idea to ensure the positive direction of training. The discriminator uses a three-layer convolutional neural network to feature the image generated by the incoming generator and the real color image, and makes a judgment, and continuously optimizes its own network according to the judgment result to improve the ability of the model to identify "true and false"[12], and the result is also fed back to the generator network to iteratively optimize the generator. It is through the continuous confrontational iteration between the generator and the discriminator that the purpose of unsupervised image coloring is achieved, and from the current experimental results, the resulting color image has a high degree of recognition.

3.1. Implementation Process

3.1.1. Generative Adversarial Networks

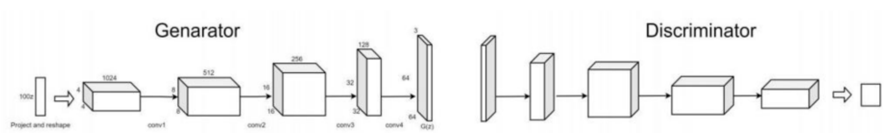

Figure 2. GAN network structure.

In recent years, with the continuous deepening of research in image colorization, from the traditional image rendering method based on manual labeling, to the use of neural networks for supervised learning based on reference images, and then to the use of unsupervised learning methods that are attracting the majority of scholars to conduct in-depth research, many methods of image colorization have emerged, and GAN networks, that is, generative adversarial networks, are a method of unsupervised learning, the essence of which is through the mutual game between two neural networks. Conduct the study of feature distributions.

The GAN network consists mainly of a generator network and a discriminator network. Among them, the generator network can be built using a multilayer convolutional neural network, mainly used to generate an expected result, and the discriminator network determines the result generated by the generator, and defines the result as true when it meets expectations, and defines it as false when it does not meet, and the result determined by the discriminator will be re-fed back to the generator, guiding the learning of the generator, and regenerating new results. The GAN network is mainly through such a continuous game confrontation, to gradually reduce the loss of the generator, in order to generate enough for the discriminator to determine the true result, and the discriminator through the correct or false judgment of each judgment result, and constantly improve and enhance its own judgment ability, in order to make the correct judgment as much as possible. The ultimate ideal state is to reach a Nash equilibrium point, that is, the result generated by the generator is judged by the discriminator, showing that the match and the non-compliance are half of the situation. At this point, it shows that the discriminator has not been able to distinguish between the results generated by the generator and the real results which is "true" and which is "false", in other words, the results generated by the generator have reached a degree of almost "false and true", which is the most ideal state expected to be achieved by model training.

3.1.2. ResNet

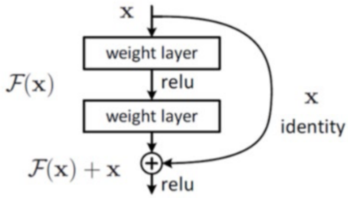

Figure 3. ResNet unit.

ResNet (deep residual network) is mainly proposed to solve the problem of degradation of training results caused by gradient disappearance and gradient explosion when training deeper and more complex models, and its essence is mainly reflected in the idea that when building a network structure, the network model can be appropriately "short-circuited". As shown in Figure 3 is a residual network unit in a residual structure, such a structure ensures that even after the learning of the unit after the gradient has dropped to zero, it is only through the identity map after the previous unit, will not lead to the gradient disappearance of the problem, this article in the construction of the generator, the use of such ideas, to ensure that the learning of the generated network is always in a positive and controllable direction.

3.1.3. Wasserstein Distance

Wasserstein distance is also called Earth-Mover distance, defined as:

$$W(P1, P2) = inf_{r\sim\prod(P1,P2)\}}|E_{(x,y)\sim y}[||x-y||] \tag{3}$$

$\prod(P1, P2)$ is a collection of all possible combined distribution combined by P_1 and P_2 distribution. For each possible combined distribution R, you can sample (x,y)~r from the medium sample X and Y, calculate this pair of samples $||\ x-y\ ||$ distance, and then calculate the calculation In the combined distribution R, the expected value of the sample to the distance $E(x,y) \sim r\big[||x-y||\big]$. In all possible combined distribution, the lower boundary $inf_{r\sim\prod(P1,P2)\}}|E_{(x,y)\sim y}[||x-y||]$ is the distance of Wassesterstein.

Wasserstein distance has the following advantages over KL divergence, as well as JS divergence: First, it is a distance with positive qualitative, symmetrical, and trigonometric inequalities in the strict sense. Second, KL divergence, as well as JS divergence, when measuring a distribution, requires that both have the same support, i.e, a distribution without a cross cannot be measured, whereas Wasserstein distance can measure the distance between any two distributions. This is why Wasserstein distance was chosen as a measure for this experiment[18]

3.1.4. Mish

The formula and image of the function are as follows:

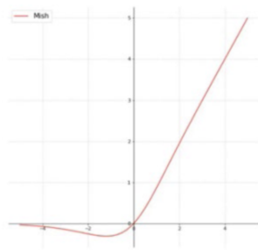

Figure 4. Mish.

$$f(x) = xtanh(softplus(x)) = xtanh(ln(1 + e^x)) \tag{4}$$

It can be seen from the figure that the function is not completely truncated when the value is negative, but allows a relatively small negative gradient to flow in, thus ensuring the flow of information. This makes this activation function borderless, allowing it to avoid the gradient saturation problems that usually exist with activation functions such as sigmoid and tanh. That is, in the case of the two sides taking the limit, the gradient approaches 1.The Mish activation function cleverly avoids this while also ensuring the smoothness of each point, which makes the gradient descent effect better than ReLU. The Mish function is a self-regular, non-monotonic neural activation function, and the smooth activation function allows better information to penetrate deep into the neural network, resulting in better accuracy and generalization. According to the paper experiments related to the mish activation function, the final accuracy of the function is improved compared to both Swish and ReLU[12].

3.2. Model

3.2.1. The composition structure of the generator network model

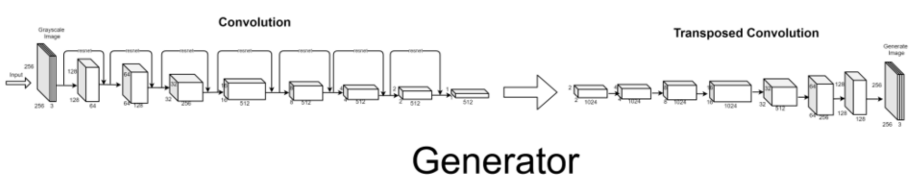

Figure 5. Generator model.

$$Loss\_G = gan\_loss + (\lambda * l1\_loss * l1\_loss) + (\lambda * tv\_loss * tv\_loss) \tag{5}$$

$$gan\_loss = BinaryCrossentropy(1, D\_gene\_out) \tag{6}$$

$$l1\_loss = tf.reduce\_mean(|target - gen\_output|) \tag{7}$$

$$tv\_loss = tf.reduce\_mean(|\Delta x|) + tf.reduce\_mean(|\Delta y|) \quad (8)$$

Here, the loss function of the generator is mainly composed of three parts, first of all, the grayscale image is the result of the distribution formed by the output of the discriminator network and the distribution of 1 by dichotomous cross-entropy, and then the L1 loss of the λ-fold true color image and the image generated by the generation network, and finally the L1 loss of the change between the adjacent pixels of λ times. The first part is the distance between the generated distribution and the distribution judged to be true by the discriminator, the second part is the distance between the generated image and the corresponding true color image, and the last part indicates that the pixels of adjacent images in the generation process should have a similar change trend. The three-part loss and the smaller the resulting image, the more realistic the resulting image.

3.2.2. The composition structure of the discriminator network model

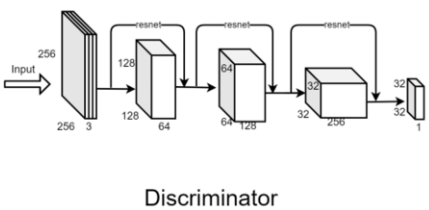

Figure 6. Discriminator model.

$$Loss\_D = L\_real + L\_gene + gp \quad (9)$$

$$L\_real = BinaryCrossentropy(1, D\_real\_out) \quad (10)$$

$$L\_gene = BinaryCrossentropy(0, D\_gene\_out) \quad (11)$$

$$gp = \lambda \underset{\hat{x} \sim P_{\hat{x}}}{E} [(||\nabla_{\hat{x}} D(\hat{x})||_2 - 1)^2] \quad (12)$$

$$L = \underset{\hat{x} \sim P_g}{E} [(D(\hat{x}) - \underset{x \sim P_r}{E} [(D(x)] + \lambda \underset{\hat{x} \sim P_{\hat{x}}}{E} [(||\nabla_{\hat{x}} D(\hat{x})||_2 - 1)^2] \quad (13)$$

Where $\hat{x}$ sample from $\tilde{x}$ and x with t uniformly sampled between 0 and 1:

$$\hat{x} = t\tilde{x} + (1 - t)0 \leq t \quad (14)$$

The loss function of the differential device here is mainly composed of three parts. After the real color image is based on the discriminator network, the result formed with the 1 distribution of the two-point cross-entropy is formed. The result obtained is adding a gradient punishment item. The first two are mainly based on the judgment of the distribution similarity, and the determination of the judgment of the generator generated and the degree of determination of real images is improved. The latter gradient punishment item is mainly to accelerate convergence and speed up iteration.

4. Experiment



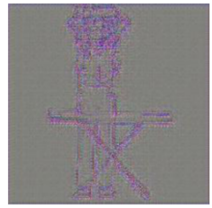

Figure 7. epoch0: sketch image. **Figure 8.** epoch0: real image. **Figure 9.** epoch0: generated image.



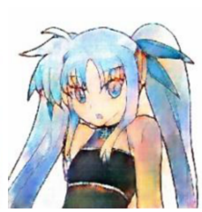

Figure 10. epoch100: sketch image. **Figure 11.** epoch100: real image. **Figure 12.** epoch100: generated image.



Figure 13. epoch200: sketch image **Figure 14.** epoch200: real image **Figure 15.** epoch200: generated image



Figure 16. epoch300: sketch image **Figure 17.** epoch300: real image **Figure 18.** epoch300: generated image

4.1. Dataset

This paper experiments use an animated sketch coloring pair dataset from Kaggle[1]. The dataset contains 17769 pairs of sketch color animated character images, of which the training set contains 14224 images and the test set includes 3545 images[19]. Each image is an RGB image with a size of 512*1024. Before the model training, the input image is first simply preprocessed, each picture is segmented and resized into two 256*256 size pictures to form a gray-color corresponding image pair, and then input into the model separately for feature extraction. The experimental selection of bitchsize is 100, the set learning rate is 0.0002, and the model is trained for 300 epochs, and the color effect of part of the training process is shown in the figure above. Figure 7-Figure 9 shows the effect of the training and test sets during initial training, Figure 10-Figure12 shows the effect of the training set and test set after 100 epochs, Figure 13-Figure 15Figure shows the generation of the test set and training set after 200 epochs, and Figure 16-Figure 18Figure shows the generation of the test set and training set after 300 epochs. At this point, some of the generated images have reached the criteria that are more acceptable for artificial discrimination. This experiment is based on ubuntu20.04 operating system, CPU 2.5GHz 6-core Intel core i5, GPU NVIDIA GeForce RTX 3080, memory 12GB, the programming language used is Python3.6, and the framework based on deep learning is tensorflow.

4.2. Evaluation

4.2.1. Objective evaluation

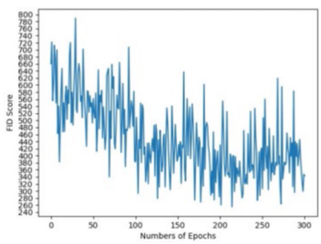

Figure 19. FID value of the model.

FID: It measures the similarity of two sets of images from the statistical aspect of the computer vision features of the original image, and is a measure of the distance between the real image and the feature vector that generated the image. This visual feature is extracted and calculated using the Inception v3 image classification model. FID has a score of 0.0 in the best case, indicating that both sets of images are identical. A lower score indicates that the two sets of images are more similar, or that the statistics of the two are more similar [17]. FID scores are often used to assess the quality of images generated by generative adversarial networks (GANs), and lower scores have a high correlation with higher quality images[14].

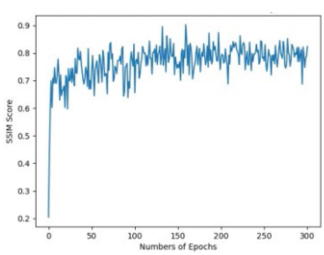

Figure 20. SSIM value of the model.

SSIM: structural similarity indicator, proposed in 2004, is a measure of the degree of distortion of the picture, can also measure the similarity of the two pictures of the indicator, SSIM is the perception model, more in line with the intuitive feeling of the human eye, mainly in measuring the correlation of adjacent pixels of the image, to reflect the structural information it has, but because SSIM should be applied to the local, so in fact, we use mean-SSIM, that is, first divide an image into some areas, Find an SSIM value for each region, and then the average of the SSIMs of all regions above an image as the SSIM value of the entire image, the higher the value, the higher the similarity.

4.2.2. Subjective evaluation

Although the quantitative evaluation index can more objectively reflect the performance of different image shading models, for the subject of colorization, the subjective evaluation of the color results artificially is also in line with the needs of production and life. Therefore, this paper selects the internationally recognized, 5-point quality evaluation index and scores it based on the user's intuitive visual experience, and the results are equally effective for evaluating the quality of the model's colorization ability. Of course, the results are affected by the interests, emotions, visual characteristics, psychological factors, etc. of the candidates, and will have certain one-sidedness and limitations.

Table 1. User evaluation standard.

Score	User visual perception quality
1	Information about the picture of the gray degree
2	Keep a small part of the gray information
3	Keep most gray information
4	Keep grayscale information and basically no noise
5	As a result, it is natural, rich, full of noise

4.3. Comparison


Figure 21. a sketch image for test **Figure 22.** a real image for test

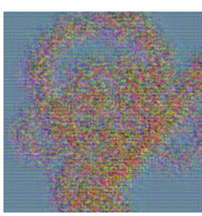
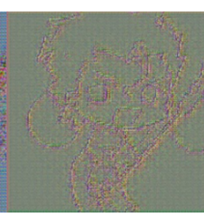
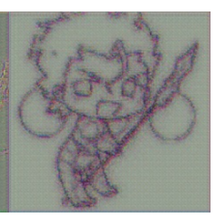

Figure 23. epoch0: generated image(a) **Figure 24.** epoch0: generated image(b) **Figure 25.** epoch0: generated image(c)



Figure 26. epoch100: generated image(a) **Figure 27.** epoch100: generated image(b) **Figure 28.** epoch100: generated image(c)



Figure 29. epoch200: generated image(a) **Figure 30.** epoch200: generated image(b) **Figure 31.** epoch200: generated image(c)



Figure 32. epoch300: generated image(a) **Figure 33.** epoch300: generated image(b) **Figure 34.** epoch300: generated image(c)

As shown in Figures 21 to 34: Three different models are colored under the same gray image as the input conditions of the model, and when different training cycles are used. Where a corresponds to the picture generated by the style transfer algorithm model, b corresponds to the picture generated by the cycleGAN algorithm model, and c corresponds to the picture generated by the algorithm model proposed in this paper. Through comparison, it can be clearly seen that the algorithm model proposed in this paper is relatively faster in training and has a relatively better coloring effect.

Table 2. Comparison from different methods.

Methods	FID	SSIM (mean)	SSIM(standard deviation)
Neural Style Transfer	346.712	0.6352416	0.09634227
CycleGAN	273.243	0.7136576	0.08305232
GAN in this paper	272.034	0.7665779	0.065

As shown in Table 2, the results of the images generated by this experimental model on the SSIM index are better than the two more cutting-edge algorithmic models based on the evolution of the GAN network, and the results of the FID indicator are also better than the other two algorithms in the same training period, and the value of the SSIM standard deviation indicator is lower than that of the other two algorithm models, which shows that the improvement trend of the image quality generated by the algorithm model proposed in this paper is also relatively stable[16].

Table 3. User evaluation results.

Methods	Color naturalness	Color saturation	Colorfulness
Neural Style Transfer	4.1	4.1	4.2
CycleGAN	4.3	4.2	4.3
GAN in this paper	4.5	4.3	4.5

This time, a total of 50 participants of different ages were randomly selected, 30 pairs of images generated by different algorithms were scored, and each participant was shown the color renderings generated by each model when the same grayscale sketch image was used as input, and it was required to select the image that was considered to be more natural and more in line with the general visual effect within 5s of time, and the participants were not informed of the algorithm model corresponding to each generated picture before making their own judgment. As shown in Table 3, the results of image quality evaluation based on user evaluation criteria can be seen as the results of coloring by the method proposed herein, which have a higher degree of recognition.

5. Discussion

This paper is mainly committed to providing a new and feasible reference scheme for the automatic coloring process of gray sketch images, so as to solve the problem of large human resources consumption and high production cost when manually coloring sketch sketches in the field of animation production. Based on the generative adversarial network, by integrating the idea of residual network, and introducing the mish activation function and gradient penalty term, this paper successfully builds an unsupervised automatic colorization model, and trains and tests on the dataset by animated sketch coloring, and finally achieves good results.

Since this experimental model is based on the generative adversarial network to build, so there are higher requirements for the computer hardware platform, training the model, the memory occupation is larger, in the future can consider from the calculation resource occupation aspects of the improvement, reduce the training cost of the model, so that the model is more convenient to build, the dependence on the computer configuration is lower.

Acknowledgments: This research was supported by Fujian Natural Science Foundation of China (2022J011233 and 2020J01266), Xiamen University of Technology (XPDKT20027), and National Natural Science Foundation of China (61871464).

References

[1] Anime Sketch Colorization Pair. https://kaggle.com/ktaebum/anime-sketch-colorization-pair. Accessed 22 Jan. 2020.

[2] LEVIN A, LISCHINSKI D, WEISS Y. Colorization using optimization[J]. ACM Transactions on Graphics (TOG,) 2004, 23(3):689-694.

[3] CHEN Ying, DONG Jiawei, ZONG Gaigai, CAO Guangcheng. Manifold preserving image colorization using segmentation[J]. China Sciencepaper, 2016(20).

[4] Reinhard E, Adhikhmin M, Gooch B, et al. Color transfer between images. IEEE Computer graphics and applications. 2001, 21(5): 34-41.

[5] Welsh T, Ashikhmin M , Mueller K . [ACM Press the 29th annual conference-San Antonio, Texas (2002.07.23-2002.07.26)] Proceedings of the 29th annual conference on Computer graphics and interactive techniques, SIGGRAPH 02-Transferring color to greyscale images[J]. ACM Transactions on Graphics, 2002.

[6] Revital Irony, Daniel Cohen-Or, Dani Lischinski. Colorization by Example[C]// Eurographics Symposium on Rendering Techniques. DBLP, 2005.
[7] Xiaopei Liu, Liang Wan, Yingge Qu, et al. Intrinsic Colorization[J]. ACM Transactions on Graphics, 2008, 27(5):152.
[8] Zezhou Cheng, Qingxiong Yang, Bin Sheng. "Colorization Using Neural Network Ensemble", IEEE Transactions on Image Processing, 2017.
[9] Zhang R, Isola P, Efros A A. Colorful image colorization[C]//European Conference on Computer Vision. Springer International Publishing, 2016: 649-666.
[10] Gatys, Leon A., et al. “A Neural Algorithm of Artistic Style.” ArXiv:1508.06576 [Cs, q-Bio], Sept. 2015. arXiv.org, http://arxiv.org/abs/1508.06576.
[11] CAO Y. Diversified dyeing based on generative countermeasures network[D]. Shanghai:Shanghai Jiao Tong University, 2018.
[12] Misra, Diganta."Mish: A Self Regularized Non-Monotonic Activation Function". ArXiv:1908.08681v3 [cs.LG] August 2020.
[13] CMingxuan Li, Guoxiong Zhou, Aibin Chen, Jizheng Yi, Chao Lu, Mingfang He, Yahui Hu. "FWDGAN-based data augmentation for tomato leaf disease identifification", Computers and Electronics in Agriculture, 2022
[14] Shiping Deng, Kaoru Uchida, Zhengwei Yin. "Cross-modal and Semantics-Augmented Asymmetric CycleGAN for Data-Imbalanced Anime Style Face Translation", 2021 3rd International Conference on Video, Signal and Image Processing, 2021
[15] Jia Guo, Jinghai Xie, Jingzhong Yuan, Yu Jiang, Shihua Lu. "Fault Identification of Transmission Line Shockproof Hammer Based on Improved YOLO V4", 2021 International Conference on Intelligent Computing, Automation and Applications (ICAA), 2021
[16] Qing-Hu Wang, Zhi-Li Pei, Xiu-Ping Hou, Qi Sun, Hong Zheng. "DNA Algorithm Based on Incomplete Molecule Commixed Encoding for the Shortest Path Problem", 2009 First International Conference on Information Science and Engineering, 2009
[17] Beibei Jing, Hongwei Ding, Zhijun Yang, Bo Li, Liyong Bao. "Video prediction: a step-by-step improvement of a video synthesis network", Applied Intelligence, 2021
[18] Cheng Cheng, Beitong Zhou, Guijun Ma, Dongrui Wu, Ye Yuan. "Wasserstein distance based deep adversarial transfer learning for intelligent fault diagnosis with unlabeled or insufficient labeled data", Neurocomputing, 2020
[19] Shu Wang, Dianwei Wang, Pengfei Han, Xincheng Ren, ZHIJIE XU. "Text Recognition in UAV Aerial Images", 2021 4th International Conference on Artificial Intelligence and Pattern Recognition, 2021