Hindawi
Scientific Programming
Volume 2022, Article ID 4188217, 8 pages
https://doi.org/10.1155/2022/4188217

Research Article

Research on Automatic Intelligent Coloring of Animation Sketch Based on Enhanced Deep Learning

Zhe Wang [1,2]

[1]*Faculty of Design and Architecture, Universiti Putra Malaysia, 43400 UPM Serdang Selangor, Darul Ehsan, Malaysia*
[2]*Art & Design Department, Taiyuan Institute of Technology, Taiyuan, Shanxi 030008, China*

Correspondence should be addressed to Zhe Wang; gs61854@student.upm.edu.my

Received 11 March 2022; Revised 29 March 2022; Accepted 11 April 2022; Published 28 April 2022

Academic Editor: Jie Liu

An automatic intelligent coloring model of animation sketch based on enhanced deep learning is proposed. In the proposed model, generative adversarial networks (GANS) are adopted. The U-net network based on the Swish function residual enhancement is used in the generative model, and the ResNet network is used in the discriminant model. The U-net embedded with the Swish Gate module is adopted to transmit feature map information. The perceptual network on the discriminator is used to perceive the perceptual features of the generated image and the actual image and calculate the perceptual loss. Experiment results show that perceptual loss can better capture the difference between black-and-white images and color images, so as to better train the network end-to-end. After comparative analysis, it can be concluded that compared with the existing methods, the proposed model has greater advantages in processing animation sketches. The color images it generates have higher visual quality and richer color diversity and matching.

1. Introduction

At present, the coloring of animation sketches mostly depends on the hand-painted coloring of professional animation painters, which will spend a lot of time and energy. At the same time, the coloring effect is also affected by individuals [1, 2]. The emergence of convolutional neural networks [3] provides a new perspective for the coloring of gray images. Its emergence makes it possible to complete many tasks in computer vision at the same time. It is necessary for the computer to automatically color the animation sketch. At the same time, for some ordinary people, they can use this method to color the line sketch and create their favorite color pictures. These methods can automatically colorize animation line art to generate rich color pictures; in addition to manual selection of specific colors color to color, the coloring time is much faster than hand-painted coloring. However, generative adversarial networks (GANS) have always had a long training time, unstable generation effect, and non-convergence of the network [4, 5]. These problems can lead to poor quality of color pictures generated by the GANS-based animation line art coloring model. For example, the color filling is unreasonable, the filling color exceeds the filling area, and the color brightness is inconsistent.

As far as GANS-based coloring models are concerned, it is challenging to meet the actual needs, and the coloring results also need to be screened. Some color pictures of poor quality are inevitable, time-consuming, and laborious. GANS consists of a generator and a discriminator [5]. When the animation line art is colored, the generator inputs the animation line art and outputs the colored image. Moreover, the choice of the generator network structure and loss function will directly affect the quality of the final output color picture. Therefore, designing a suitable stable network and a suitable loss function can improve the quality of generated color images. The role of the discriminator is to discriminate whether the generated color image is close to the effect of artificial coloring avoidance. The final output is a color image with poor quality, and the discriminator will affect the training stability of GANS. Training GANS needs to achieve the Nash

equilibrium; the discriminator network needs to be further optimized to ensure training stability.

2. Related Works

In recent years, GANS has received increasing attention in deep learning. A generative adversarial model usually consists of a generator and a discriminator. The generator captures the underlying distribution of actual samples and generates new data samples. The discriminator is often a binary classifier that distinguishes real examples from generated samples as accurately as possible. The discriminator guides the training of the generator, and the alternating movement between the two models is used for continuous confrontation. Finally, the generative model can better complete the generation task. With the emergence of more and more GANS variants, GANS has achieved significant results in various fields of images. In image coloring, GANS also occupies an important position in mainstream algorithms. At present, the deep learning-based automatic coloring model mainly adopts the architecture of GANS.

Pix2Pix [6] is also a significant variant of GANS, using conditional generative adversarial networks (CGANS) to achieve image-to-image conversion; it can do many things, such as drawing sketches, convert outlines to pictures, converting night scenes to day scenes, auto colorize, and more.

Moreover, Style2paints, as a style transfer coloring model variant of GANS, needs to provide a reference image for color use in advance when converting the anime line art into color images [7]. The generator network proposed by Style2paints also uses U-net with residual enhancement. In the web, a residual module is added between each level in the right half of the network to enhance the coloring detail texture, and an auxiliary classifier is added to the generator network structure. The discriminator can distinguish the true and false of the generated image and classify its related styles to achieve style transfer.

PaintsChainer, which is now widely used, uses an unconditional discriminator and has achieved remarkable results [7]. Users only need to input an animation line art picture to get a color picture, and they can also get the effect under the color style by adding the color they want. However, because no labels make it easy to pay too much attention to the relationship between lines and feature maps, the image composition will lead to overfitting, and the line filling will be confusing.

It can be seen that to improve the performance of the GANSs network, much research has been done on its network structure. Moreover, GANSs have also achieved outstanding results in animation line art coloring. U-net has been proven to have an excellent effect on the coloring of anime line drafts. Still, the biggest problem is that the up-sampling convolutional layer and the down-sampling convolutional layer of U-net are directly spliced [8, 9]. When the first layer is discovered, it can simply jump-connect all the features directly to the last layer of the decoder, thus minimizing the loss, which results in the middle layers of the network not being able to learn anything, no matter how many times it is trained. In the network, there will be a problem of gradient disappearance in the middle layer.

A deep learning model for animation line draft coloring is proposed to solve the above problems. The overall structure of the model is an adversarial generative network model. The generator structure of the model uses the improved residual-enhanced U-net network structure, and the discriminator uses the ResNet network structure [10–12]. Inspired by the ResNet network, the original U-net network sampling up-convolutional and down-sampling convolutional layers changed directly. The method is no longer a jumper connection. The Swish activation function is used, and two connection modules are proposed. The proposed Swish module can better filter the feature information transmitted in the network and improve the network's learning ability. When the low-level convolutional layer completes the task, the high-level convolutional layer can still obtain the filtered feature information for learning. After coloring the animation line draft, the color details are confused, and the gradient disappears during the training process. In addition, the discriminative network is used as a perceptual network, and the perceptual features of the generated image and the actual image can be obtained to calculate the perceptual loss. The coloring model with perceptual loss can generate qualitatively better color images.

3. Coloring Model of Animation Sketch Based on Enhanced Deep Learning

U-net network is a U-shaped convolutional neural network structure, which is initially used in the field of image segmentation. It has two branches, the left one is the encoding network structure, and the right one is the decoding network. U-net has also been widely used in image synthesis. However, it is easy to form gradient disappearance in the middle layer during network training. The emergence of non-linear activation functions makes neural networks more expressive.

To improve the quality of the generated color images, a generative adversarial model for coloring animation sketch is proposed, as shown in Figure 1. The generator network uses the residual enhanced Swish activation function based on U-net to convey feature map information. The selection in the discriminator uses a ResNet network. It has the following advantages.

(1) Two kinds of connections are proposed based on the residual module and the Swish function. This can solve the problem of the gradient disappearance of the middle layer in the U-net network training process, better filter the feature map, better learn the feature map of each level, will not cause the gradient to disappear, and the convergence curve can also converge faster.

(2) Propose using perceptual loss to better capture reference images and the difference between the generated image, making the resulting color image more textured and color-to-color transitions are smoother.

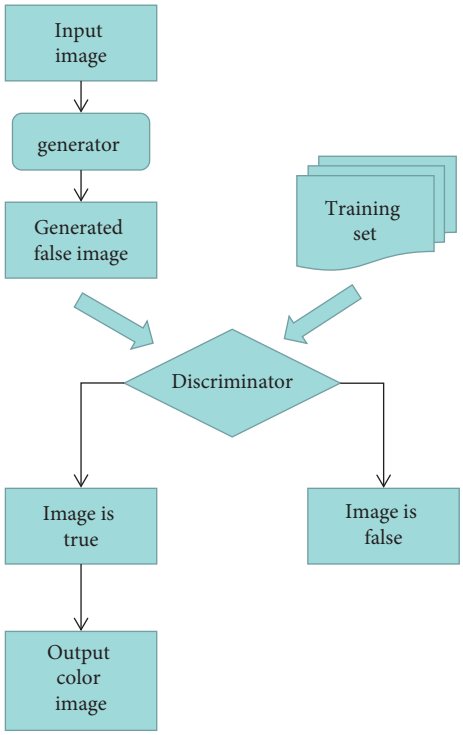

Figure 1: Overall structure of the network.

(3) The experimental results in the Anime Sketch Colorization Pair dataset show that the coloring effect of the method proposed in this paper is better than the current coloring method and is close to the impact of artificial coloring.

3.1. Whole Network Structure. The generator network structure is based on an improved version of U-net, as shown in Figure 2, which is a Swish U-net network structure enhanced by residuals. The network has six different resolution levels, and as the level increases, the resolution gradually decreases. Like U-net, Swish U-net can also be regarded as the left and right branches. Still, a Swish Mod is embedded between the left and right components of the same resolution level to filter the information transmitted from the encoding path to the decoding path; instead, of the original jumper, Swish Mod can speed up the convergence speed of the network and improve the performance of the network. Each green dotted box in Figure 2 is a Swish Gated Block, and there are 10 in total. In the left branch, the output of each Swish Gated Block consists of the feature map output by the residual part and the feature map filtered by Swish Mod; In contrast, in the right department, the output of each Swish Gated Block consists of three parts, which are the feature map output by the residual part, the feature map filtered by the input Swish Mod, and the feature map filtered by the Swish module corresponding to the left branch.

Except for the last convolutional layer of the network, all convolutional layers use normalization and LReLU functions. The input of the Swish Gated Block of the i-th layer is the output of the Swish Gated Block of the i-1 layer for 1×1 convolution obtained after the operation. In addition, the number of convolution kernels for the 1×1 convolution operation in the i-1th layer is the same as the number of convolution kernels for each convolutional layer in the i-th layer. From resolution level 1 to resolution level 6, in each resolution level, the number of convolution kernels of each convolutional layer is 96, 192, 288, 384, 480, and 512 in turn. The last convolutional layer will output the final color image, consisting of 27 1×1 convolution kernels, and no normalization and activation functions are used.

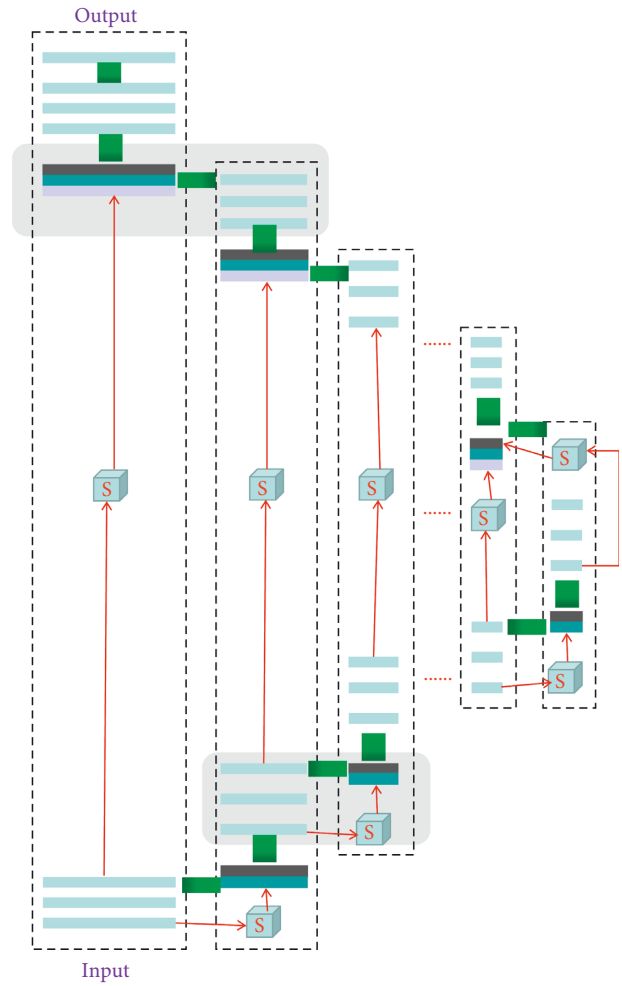

Figure 2: Network structure of Swish U-net.

Generally speaking, the role of the discriminator is to distinguish between authentic images and generated images. ResNet is selected as the discriminator network on the discriminator. Here, the discriminator has two tasks: (1) It discriminates between generated images and authentic images. (2) The generated and authentic images' perceptual features are extracted by calculating the perceptual loss as a perceptual network. The discriminator network is finally normalized to improve the stability of network training. Then, the ReLU activation function is used to make network training faster while preventing gradients from disappearing.

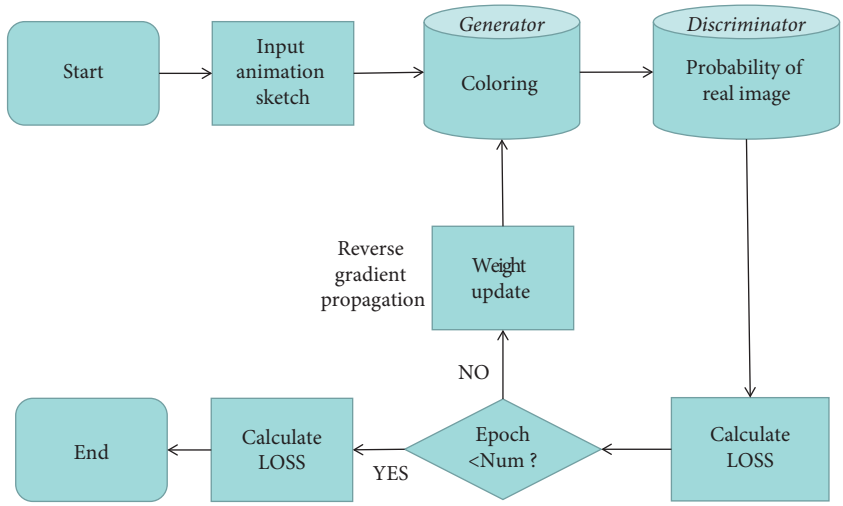

Figure 3: Training process of the network.

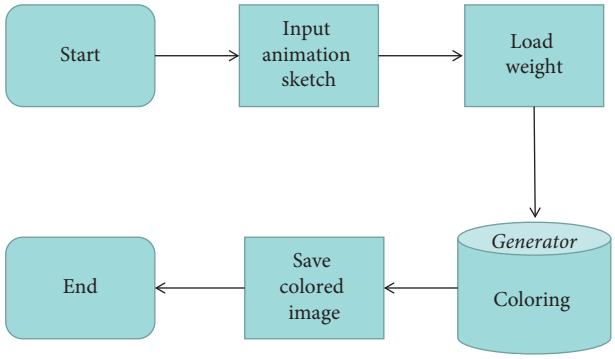

Figure 4: Prediction process of the network.

Figure 3 shows the training process. Each step includes two processes of forwarding propagation and back-propagation. The two processes are completed once as one epoch. When the number of times is less than the set training value Num, it will continue to cycle; Figure 4 shows the prediction flow chart after the model is trained, only forward propagation.

3.2. Swish Module and Loss Function. The new residual module Swish Gated Block proposed in this paper improves the residual module in ResNet [12–14]. The Swish Gated Block is composed of Swish module and residual, Swish module contains a convolutional layer and Swish activation function. In the structure of the proposed residual module, x represents the input data, $F(x)$ represents the residual, $F(x)+x$ is the output of the residual module, and "+" represents the corresponding addition of pixel points, $G(x)$ denotes the result of the convolutional layer in Swish module, "·" means the corresponding multiplication of pixels; $T(x)$ represents the output of the convolutional layer in Swish Gated Block after the non-linear LReLU function, $S(x)$ is the output of Swish module, "⊕" represents the splicing between feature maps, and "$T(x)\oplus S(x)$" is the final output of Swish Gated Blok.

In the residual module, the input data x are directly added to the residual without processing; In Swish module, x is processed, and the Sigmoid function is used; its advantage is that it can control the magnitude of the value, and in the deep network, the importance of the data can be kept from significant changes. In addition, non-linear LReLU is used for the convolutional layer in SwishGatedBlock, which has a better effect than ReLU for generating classes. Swish module filters the input data x, like a gate that controls the transmission of the input data x from the bottom layer to the high layer through a shortcut feature map. Swish module is defined as follows:

$$S(x) = X \cdot \sigma(G(x)), \tag{1}$$

where $S(x)$ is the output of Swish module, x is the input data, $G(x)$ is the output of the convolutional layer, $\sigma(\bullet)$ represents the Sigmoid function, and "•" represents the multiplication of the corresponding pixels.

The output of Swish Gated Block is as follows:

$$y = T(x) \oplus S(x), \tag{2}$$

where $T(x)$ is the residual part in the module, $S(x)$ is the information filtered by Swish module, and finally spliced together to output the obtained feature map.

The generator and discriminator proposed in this paper are trained separately, using pairs of matching images as a data set of images. Anime line art is the input data, and paired color images are the labels. For colorization tasks, simply comparing the pixel colors of the generated image and the reference color image can seriously affect the quality of the output image. Because a black and white image is given, the hair color can be silver or black. The black and white image has a one-to-many relationship with the colored image. Still, there is only one label, so it is unreasonable only to consider the L1loss of each pixel. Perceptual missing is proposed for this purpose, which can help capture the difference between the generated color image and the reference image, and L2 regularization is added to prevent the model from overfitting

Figure 5: Colored images obtained from animation sketch.

[13]. The perceptual loss is calculated based on the feature map and expressed as follows:

$$L_g = \sum_{1} \lambda_1 \|\varphi_l(T) - \varphi_l(G)\|_1 + \alpha \sum_{l=1}^{n} \lambda_l^2. \tag{3}$$

Among them, 1 takes the value [0, 5], T represents the generated image, G represents the actual image, φ_0 (G) represents the convolution operation with the network structure of the discriminator, φ represents no convolution operation, represents the original image, φ_1 (G) denotes the output result (feature map) of the first layer of convolution, representing the perceptual feature, and so on. λ_1 $= \{0.88, 0.79, 0.63, 0.51, 0.39, 1.07\}$, indicating the weights of different layers. The regularization coefficient $\alpha = 0.009$, and the optimizer adopts Adam.

The network of the discriminator uses ResNet, and the data are normalized after the convolutional layer so that the data will not be too large and lead to unstable training [14, 15]. The loss of the discriminator here is the discriminator loss proposed by GANS.

$$L_d = -E[b\sigma(D(T)) + lb(1 - \sigma(D(T)))], \tag{4}$$

where G represents the actual image, T represents the generated image, D represents the discriminator, σ $(\cdot)$ represents the sigmoid function, and E represents the mathematical expectation.

Figure 6: Comparisons of Swish U-Net model (with perceptual loss), SwishU-Net-WPL model (without perceptual loss) and U-Net model.

4. Case Study

4.1. Datasets and Evaluation Indicators. To verify the performance of the proposed method, training is performed on large datasets, Anime Sketch Colorization Pair, which has a large number of paired anime line art images and anime colorization images. Training is conducted on 15432 anime line drawings and their corresponding color images, and all pictures of the experiments are resized to 512×512 resolution. Assessing the quality of generated images has always been a complex problem. The colors generated by different coloring models in other areas of the same coloring image are also different during the coloring process. In addition to differences in color, images generated by different shader models also vary significantly in image quality (texture, shading, brightness) and visual quality of images. Therefore,

Table 1: Comparisons of SwishU-Net model (with perceptual loss), SwishU-Net-WPL model (without perceptual loss), and U-Net model on four quantitative indicators.

	FID	PSNR	SSIM	FSIM
Swish U-net model	108.23	18.55	0.85	0.86
U-net model	115.96	17.62	0.87	0.85
Swish U-net-WPL model	119.25	15.97	0.84	0.84

we use several standard image quality quantitative indicators to evaluate and compare SwishU-net and other existing colorization methods. The quantitative evaluation indicators used in the experiments include peak signal-to-noise ratio (PSNR), structural similarity (SSIM), and feature similarity (FSIM) [15–17]. To confirm the role of perceptual loss in the colorization model, we use Fréchet inception distance (FID) [15–17] as an evaluation criterion to quantify the quality of color images. Figure 5 shows the coloring results of the proposed coloring model. To get a color image of eight different colors, only an animation line draft is needed to input.

4.2. Experimental Results. Figure 6 shows a comparison of two colorization models with and without perceptual loss. It can be seen that the color image generated by the shading model with perceptual loss is more vivid and complete, especially the color gradient is smooth, the shadow distribution is reasonable, and there is no sense of violation; the color image generated by the shading model without perceptual loss. The colors are not rich enough, and there are fewer gradients of color changes. In addition, the color image generated by the colorization model without perceptual loss is also low in color saturation, and there is no apparent boundary between the characters and the background in the picture. Therefore, the perceptual loss significantly influences the shading effect, and the image texture generated by the shading model with the perceptual loss is more detailed. The transition between colors is also smoother.

To further investigate our approach, quantitative analysis was used to evaluate the quality of the generated images, as shown in Table 1. We used FID as a quantitative indicator to assess the generated color images' quality (sharpness) and color diversity. The automatic colorization model is a one-to-many transformation, where FID is used to determine the quality of the generated color images; SwishU-net without perceptual loss is abbreviated as SwishU-net-WPL. In addition, PSNR, SSIM, FSIM are used here to evaluate the performance of the three algorithms, and the best results are shown in bold. SwishU-net achieves the best performance on all metrics. SwishU-net without perceptual loss has the worst performance on all metrics, indicating that perceptual loss plays a vital role in the colorization model. The quality of the color image generated by the proposed Swish module residual enhanced network is better than the image generated by the U-net network, indicating that the Swish module residual enhanced generative model has a better coloring effect.

Table 2: Comparisons of the Swish U-Net model, Style2paints model, and PaintsChainer model on average running time and model complexity.

	Average running time	Model complexity
Swish U-net model	≈30	≈58000
Style2paints model	≈41	≈70000
PaintsChainer model	≈46	≈77000

4.3. Algorithm Complexity Calculation. Table 2 compares the algorithmic complexity of SwishU-net and the current mainstream algorithms. All algorithms are based on python language and implemented on GPU; only 512×512 images are tested here. 16 parameter layers are selected for the above experiments to balance performance and computational efficiency.

It can be seen that Style2paints and PaintsChainer consume much time due to the complex optimization process. At the same time, the generative network of SwishU-net reduces the running time by not using normalization layers. Despite using a lightweight framework, the average running time and the number of parameters results show that SwishU-net performs better after quantitative analysis.

5. Conclusion

It takes time and energy to color the animation sketch by manpower. As a popular color generation technology of network animation sketch, GANS has not achieved certain results in the past few years. There are some basic problems, such as color confusion, poor color gradient, unreasonable color sketch, and so on. Therefore, the GANS network based on a Swish function module proposed in this paper can better learn the details of sketch during color filling and avoid color confusion when the color exceeds the filled area. At the same time, it can directly carry out end-to-end training from animation sketch to color picture. The module can automatically color the sketch and generate color pictures with rich colors and clear textures. The experimental results show that this method has better coloring ability than the existing methods and can obtain more realistic and better visual effect color images. In the future, if there are better conditions, it will make up for the shortcomings of this paper, use the GPU with stronger performance to increase the network parameters, and expand the network scale, in order to generate higher resolution images and solve the problems of complexity and diversity. Although the U-net network structure has better generation effect than the self-encoder, it also increases the parameter quantity and complexity of the model. In the future, we will consider looking for a more suitable network structure as the discriminator model to reconstruct the image, so as to achieve better generation effects while simplifying the model.

Data Availability

The data set can be accessed upon request.

Disclosure

This paper is a research project of Philosophy and Social Sciences in Shanxi Province in 2020, the visualization research of traditional animation in information interaction from the perspective of UX. (2020W229).

Conflicts of Interest

The authors declare that there are no conflicts of interest.

References

[1] E. Sohn, J. Jeon, and T. J. Park, "A two layered approach for animation sketching," *Journal of Korea Multimedia Society*, vol. 12, 2009.

[2] E. Eisung Sohn and fnm Yoon-Chul Choy, "Sketch-n-Stretch: sketching animations using cutouts," *IEEE Computer Graphics and Applications*, vol. 32, no. 3, pp. 59–69, 2012.

[3] M. Matsugu, K. Mori, and Y. Mitari, "Subject independent facial expression recognition with robust face detection using a convolutional neural network," *Neural Networks*, vol. 16, no. 5-6, pp. 555–559, 2003.

[4] C. Yang, D. Eschweiler, and J. Stegmaier, "Semi- and self-supervised multi-view fusion of 3D microscopy images using generative adversarial networks," 2021, https://arxiv.org/abs/2108.02743.

[5] N. Sachdeva, M. Klopukh, and R. S. Clair, "Using conditional generative adversarial networks to reduce the effects of latency in robotic telesurgery," *Journal of Robotic Surgery*, vol. 6, 2020.

[6] P. Isola, J. Y. Zhu, and T. H. Zhou, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5967–5976, IEEE Computer Society Press, Honolulu, HI, USA, July, 2017.

[7] H. Ren, J. Li, and N. Gao, "Two-stage sketch colorization with color parsing," *IEEE Access*, vol. 8, Article ID 44599, 2020.

[8] N. Berk and P. Valentina, "Use of 2D U-net convolutional neural networks for automated cartilage and meniscus segmentation of knee MR imaging data to determine relaxometry and morphometry," *Radiology*, vol. 288, 2018.

[9] S. Hasan and C. A. Linte, "A modified U-net convolutional network featuring a nearest-neighbor Re-sampling-based elastic-transformation for brain tissue characterization and segmentation," in *Proceedings of the 2018 IEEE Western New York Image and Signal Processing Workshop (WNYISPW)*, October, 2018.

[10] C. Brito, A. Machado, and A. Sousa, "Electrocardiogram beat-classification based on a ResNet network," *Stud Health Technol Inform*, vol. 264, 2019.

[11] A. Sl, B. Shwa, and B. Ydza, "Detecting pathological brain via ResNet and randomized neural networks," *Heliyon*, vol. 6, 2020.

[12] H. Ren, M. El-Khamy, and J. Lee, "DN-ResNet: efficient deep residual network for image denoising," 2019, https://arxiv.org/abs/1810.06766.

[13] Y. Wang, X. Zhou, and H. Zhou, "Transmission network dynamic planning based on a double deep-Q network with deep ResNet," *IEEE Access*, vol. 9, no. 99, p. 1, 2021.

[14] M. Y. Gao and P. Song, "A novel deep convolutional neural network based on ResNet-18 and transfer learning for detection of wood knot defects," *Journal of Sensors*, vol. 2021, Article ID 4428964, 16 pages, 2021.

[15] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[16] L. Lin Zhang, L. Lei Zhang, X. Xuanqin Mou, and D. Zhang, "FSIM: a feature similarity index for image quality assessment," *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2378–2386, 2011.

[17] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, G. Klambauer, and S. Hochreiter, "GANS trained by a two time-scale update rule converge to a nash equilibrium," *Advances in Neural Information Processing Systems*, vol. 1, no. 2, pp. 6626–6637, 2017.