electronics

MDPI

Article

Colorization of Logo Sketch Based on Conditional Generative Adversarial Networks

Nannan Tian [1,2], Yuan Liu [1,2,*], Bo Wu [3] and Xiaofeng Li [4]

1 School of Design, Jiangnan University, Wuxi 214122, China; 7180306005@stu.jiangnan.edu.cn
2 School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214122, China
3 School of Software Engineering, Shandong University, Jinan 250101, China; BBqaz1234@hotmail.com
4 School of Computer Science and Technology, Shandong Jianzhu University, Jinan 250101, China; lixf@sdjzu.edu.cn
* Correspondence: lyuan1800@jiangnan.edu.cn

Abstract: Logo design is a complex process for designers and color plays a very important role in logo design. The automatic colorization of logo sketch is of great value and full of challenges. In this paper, we propose a new logo design method based on Conditional Generative Adversarial Networks, which can output multiple colorful logos only by providing one logo sketch. We improve the traditional U-Net structure, adding channel attention and spatial attention in the process of skip-connection. In addition, the generator consists of parallel attention-based U-Net blocks, which can output multiple logo images. During the model optimization process, a style loss function is proposed to improve the color diversity of the logos. We evaluate our method on the self-built edges2logos dataset and the public edges2shoes dataset. Experimental results show that our method can generate more colorful and realistic logo images based on simple sketches. Compared to the classic networks, the logos generated by our network are also superior in visual effects.

Keywords: generative adversarial networks; image translation; logo colorization

Citation: Tian, N.; Liu, Y.; Wu, B.; Li X. Colorization of Logo Sketch Based on Conditional Generative Adversarial Networks. *Electronics* **2021**, *10*, 497. https://doi.org/10.3390/electronics10040497

Academic Editor: Savvas A. Chatzichristofis

Received: 25 December 2020
Accepted: 17 February 2021
Published: 20 February 2021

1. Introduction

The development and application of machine learning have gradually brought many conveniences to designers. For example, the Luban system from Alibaba AI Design Lab can generate high-quality banners based on existing design materials. However, some basic materials still need to be designed manually, such as icons or logos. The automatic design of the logo is full of challenges. If machine learning can assist designers in designing, it will greatly improve design efficiency and generate higher application value.

In the design process, the colorization of the logo sketch is a skill that designers must master. A good designer usually needs to have a very good sense of color in order to coordinate the sketch outline and color. The visual communication of color is usually better than that of graphics. The first thing people see is often bright colors, which are easier to recognize and remember. Moreover, different degrees of color saturation, purity, brightness and other factors will produce different color effects. Different colors will bring people different visual feelings and psychological experience. Therefore, it is desired to create an artificial intelligence designer who can perceive colors from training data and automatically paint harmonious and unified colors on the logo sketch.

In recent years, with the development of deep learning, especially the Generative Adversarial Networks (GAN), colorization of sketches has become possible. In 2014, Ian Goodfellow [1] first proposed the concept of GAN, which can generate realistic images through Generative Network. GAN has become one of the most popular algorithms in Deep Learning, which achieved remarkable success in super-resolution tasks [2,3], semantic segmentation [4,5], text to image synthesis [6–9], image denoising [10], data augmentation [11], image fusion [12,13], image stylized [14–16] and so on.

In the meantime, GAN-based logo design has also gradually attracted the attention of scholars. Alexander Sage et al. [17] first proposed a GAN-based logo design network. They established a dataset (LLD) to tangle and stabilize GAN training through class labels obtained through clustering, which can generate a large number of reasonable icons. Ajkel et al. [18] proposed LoGAN: an improved auxiliary classifier called Wasserstein Generation Antagonistic Neural Network (with gradient penalty points), capable of generating icons conditioned on twelve different colors. Oeldorf et al. [19] propose Conditional Style-Based Logo Generation with Generative Adversarial Networks. The previous method was able to generate some high quality and high color saturation of the logo image. However, designers need to have a specific logo design sketch based on a different color requirements. Because the logo image of the previous method has greater randomness, rather than a specific logo design sketch. It does not meet the designer's requirements for a specific logo image.

To address these problems, in this paper, we propose a novel logo colorization network which has made the corresponding improvement method. It can paint logos with harmonious and unified color according to the logo sketch provided by the designer. Instead of generating data based on random noise, the generator takes the given sketch image as the input conditional information, making the output more targeted.

This is essentially an image translation task. In recent years, GAN has achieved great success in image-to-image translation [20–22]. In particular, Phillip Isola et al. [20] proposed a universal structure, Pix2Pix, which has achieved good results in many image translation tasks. We make several improvements on the basis of Pix2Pix in order to make the output logo colors richer and more realistic. The contribution of this paper is mentioned as follows:

1. We improve the traditional U-Net structure. Channel attention and spatial attention mechanism [23] are added to the skip-connection in the U-Net module to improve the stability of the output.
2. We stack several attention-based U-Net modules (Att-UNet) in the generator, and each Att-UNet module output one image. We randomly select one image from Att-UNet modules and feed it to the discriminator training, which can prevent the problem of mode crash and make the training more stable.
3. We add a new loss function based on the Pix2Pix objective function to make the output of the generator more diversified.

The rest of this paper is organized as follows. A brief literature review is provided in Section 2. Section 3 describes our network structure and training methods. Section 4 discusses the experimental results in detail. Section 5 illustrates the conclusion and future work.

2. Related Work

Related studies for our work include image colorization, conditional generative adversarial networks, and attention mechanism. This section introduces important works in these three areas.

2.1. Image Colorization

In the last decade, several methods [24–31] have been introduced for image colorization tasks. Deshpande et al. [26] using a variational autoencoder (VAE) to learn a low dimensional embedding of color fields. They build a conditional model for the multimodal distribution to create diverse colorization. Mouzon et al. [27] designed a fully automatic image colorization framework by combining the variational method with a CNN. Recently, researchers have adapted adversarial networks for image colorization. Nazeri et al. [30] use a conditional Deep Convolutional Generative Adversarial Network (DCGAN) to generalize the process of automatic image colorization. Sharma et al. [31] proposed Self-Attention-based Progressive Generative Adversarial Network (RIC-SPGAN) to perform the denoising and colorization of the image. However, these methods mainly

focus on colorization of gray-scale images, which are mainly applicable to the restoration of aged or degraded images. The colorization of sketch image is still full of challenging.

2.2. Conditional Generative Adversarial Networks

The main idea of GAN is that the generative model and discriminant model play games against each other to produce good output. The generator takes the random noise as input and generates the fake data samples. The discriminator, which acts as a binary classifier, attempts to distinguish between the real samples from training data and the fake samples from the generator. During the training, the goal of the generator is to generate as many real images as possible to fool the discriminator network. The objective of the discriminator is to distinguish the output of the generator from the real logos, which is a dynamic game process. The most ideal state is that generator can generate an image that is enough to resemble the truth. In GAN, the discriminator is used to determine whether the sample generated by the generator is real or fake.

For discriminator, it is difficult to determine whether the image generated by generator is real. The disadvantage of the classic GAN is that the output is uncontrollable. Mirza et al. [32] proposed Conditional Generative Adversarial Nets (CGAN) to optimize the classic GAN. They designed a GAN with conditional constraints, which introduces conditional variable in both the generation model and the discrimination model. The conditional information can guide the data generation process and make the output more targeted. These condition variables can be based on a variety of information, such as category labels [32,33], text [6,7] and image [20,21,34]. Phillip et al. [20] proposed an image translation network Pix2Pix which uses image as the conditional information. The generator in Pix2Pix uses the architecture of U-Net [35]. For discriminator in Pix2Pix, PatchGAN classifier is used to achieve good results in a variety of image translation tasks. However, Pix2Pix must be trained using pairs of data. Cycle-GANs [34] and disco-GANs [21] were proposed to solve the training of unpaired data, which are widely used for tasks of style migration. Similar to Pix2Pix, we design an image translation network to generate logo based on sketch. To make the image generation more stable, we improve the UNet module adopted in Pix2Pix and integrate the attention mechanism.

2.3. Attention Mechanism

Attention mechanisms have made great progress in natural language processing tasks [36,37].In computer vision, many papers [23,38–40] have also proved that the introduction of attention mechanism into network structure can improve the feature representation of network model. Attention not only tells the network model what to pay attention to, but also enhances the representation of specific areas. The attention method [40] only focuses on which layers at the channel level have stronger feedback ability, but it does not reflect the importance in the spatial dimension. Convolutional block attention module (CBAM) [23] applies attention to both channel and spatial dimensions. CBAM can be embedded into most of the current mainstream networks to improve the feature extraction capability of the network model without significantly increasing the amount of computation and parameters. Inspired by these works, we explore how to combine the attention mechanism of CBAM into U-Net network to pay attention to the important channel and spatial information in the process of skip-connection.

3. Methodology

In this section, we will first introduce some preliminary information. Then the structure of neural network and loss function are described.

3.1. Preliminaries

The classical GAN mainly consists of two neural networks, a generator and a discriminator. Generator G receives random noise z as an input sample and generates one image, which is represented as $G(z)$. Discriminator D, on the other hand, is a binary classifier

that determines whether a picture is real. Given the input information x, $D(x) = 1$ means that the image is completely real, and $D(x) = 0$ means that the image is completely fake. The two neural networks compete with each other, and the generator tries to generate real samples to trick the discriminator. The objective of the discriminator is to separate as much as possible the $G(z)$ from the real image. G and D constitute a dynamic game process. The formula is described as follows,

$$\min_G \max_D L_{GAN} \tag{1}$$

L_{GAN} is defined as,

$$L_{GAN} = E_x[\log D(x)] + E_z[\log(1 - D(G(z)))] \tag{2}$$

The Conditional Generative Adversarial Network (CGAN) is an extension of the original GAN. Both the generator and the discriminator add additional information y, which can be any information, such as category information, text information or other modal data. Its objective function is defined as,

$$L_{cGAN} = E_{x,y}[\log D(x, y)] + E_{z,y}[\log(1 - D(G(y, z), y))] \tag{3}$$

In our method, the CGAN is the base architecture in our network, which is used to generate an idea translation result.

The Pix2Pix [20] method apply GANs in the conditional setting. To enhance the effect of image translation, Pix2Pix adds the l_1 distance loss function between the real image and the generated image. Therefore, the total loss function of Pix2Pix is defined as follows,

$$G^* = \arg\min_G \max_D L_{cGAN}(G, D) + \lambda L_{l_1}(G) \tag{4}$$

3.2. Network Structure and Loss Function

Figure 1 shows the neural network designed in this paper. The input is the logo sketch, which is fed into the generator. In the generator, we adopt a stacked structure containing multiple attention-based U-Net blocks (Att-UNet), each of which has the same network structure but different network parameters. Each Att-UNet network outputs corresponding one logo image. Therefore, our generator can output multiple logo images.

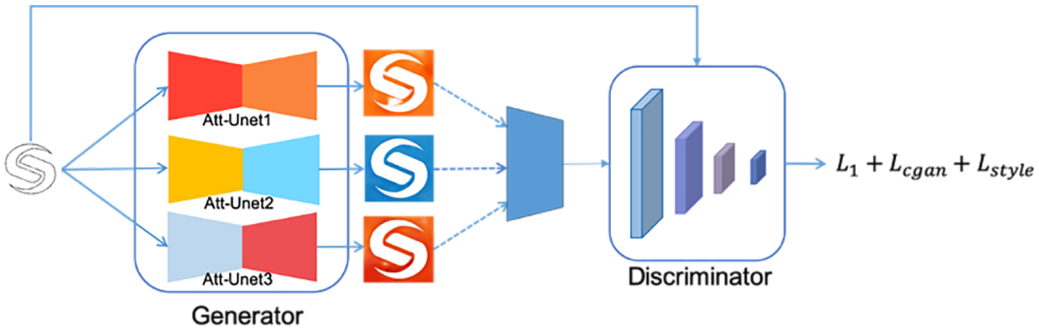

Figure 1. The structure of neural networks.

Similar to Pix2Pix [20], we use the Markovian Discriminator (PatchGAN) network as discriminator, which is one of the CGAN. The output shape of the traditional discriminator is 1×1, and the receptive field is the original image, indicating whether the image is true or false. The output shape of PatchGAN is $N \times N$, each of which is a part (Patch) in the original image. In our discriminator, the output shape is 16×16. The key features of our network and the ultimate loss function are described as below in detail.

3.2.1. Att-Unet

The classical U-Net structure is mainly composed of two parts: encoder and decoder. The encoder on the left is mainly used for feature extraction, while the decoder on the right is for upsampling. The feature map of upsampling is connected with the corresponding feature map in the encoder, which is called skip-connection. The skip-connection operation is able to maintain details at the pixel level of different resolutions. However, the importance of detailed information is not consistent. It is necessary to use attention to focus on the important channel and spatial information. Therefore, we use the channel attention and spatial attention into the process of skip-connection.

Figure 2 shows the structure of Att-UNet. Given the upsampling feature maps F_α and F_β from encoder, the skip-connection operation can be represented as $F_\gamma = F_\beta \oplus F_\alpha$. Then, F_γ is processed with channel attention. The formula is defined as below,

$$F_c = M_c(F_\gamma) \otimes F_\gamma \tag{5}$$

where the operator $\otimes$ means element-wise multiplication. $M_c(F_\gamma)$ represents the channel attention map which is calculated by Equation (6),

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \tag{6}$$

where MLP denotes the Multilayer Perceptron [41], and σ denotes the sigmoid function. $AvgPool(\cdot)$ and $MaxPool(\cdot)$ indicate the average pooling operation and the maximum pooling operation, respectively.

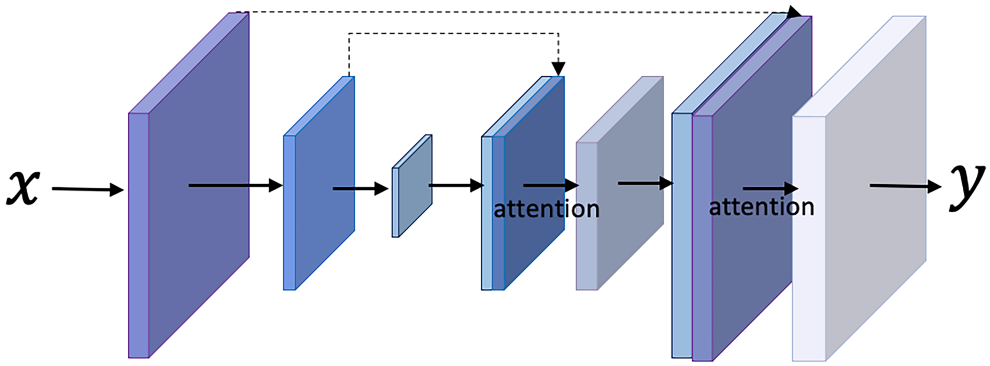

Figure 2. The structure of Att-UNet. The skip-connection operation is used to maintain details at the pixel level, the attention is used to focus on the important channel and spatial information.

In this way, the channel attention can fully consider the relationship between channels and pay attention to important channel information. Then, spatial attention further processes feature maps to generate spatial attention maps by using spatial relations between pixel.

Unlike channel attention, spatial attention focuses on the important areas in the feature map. The formula is defined as below,

$$\begin{aligned} F_s &= M_s(F_c) \otimes F_c \\ M_s(F) &= \sigma(f^{7\times7}([AvgPool(F); MaxPool(F)])) \end{aligned} \tag{7}$$

where $M_s(F_c)$ represents 2D spatial attention map. $f^{7\times7}$ denotes a convolutional layer, which the kernel size is 7×7.

3.2.2. Loss Function

Our generator consists of multiple Att-UNet blocks. To make the generator output more colorful images, based on the Pix2Pix [20] loss function, we add a new style loss function L_{style} to control the training process. This loss function maximizes the l_1 distance between the pair of output to keep the diversity. The L_{style} function is defined as follow,

$$L_{style}(G) = E_x[-\sum_{i,j=1}^{N} ||G_i(x) - G_j(x)||_1], i \neq j \quad (8)$$

where N is the total number of Att-UNet blocks in generator. G_i and G_j represent the i-th and j-th Att-UNet block, respectively.

Therefore, our final objective function is defined as,

$$G^* = \arg\min_G \max_D L_{cGAN}(G, D) + \lambda_1 L_{l_1}(G) + \lambda_2 L_{style}(G) \quad (9)$$

where the loss function L_{l_1} calculates the distance between the real image and the generated image, which was defined in Pix2Pix [20].

3.3. Training Strategy

To optimize our network, we adopt the standard training process in the paper [1]. Firstly, we optimize the discriminator using the fake image of the generator and the real image, so that the discriminator could distinguish the differences. Then, the generator and discriminator are taken as a whole network to fix the parameters of the discriminator and optimize the generator. Repeat the process above and let the generator and discriminator evolve separately and gradually.

Since our generator can output multiple different logo images for the same sketch, we randomly select one image as a fake image and feed it into the discriminator each time we optimize the discriminator. This operation is similar to a kind of online data augmentation. In this way, we can stabilize the training of the whole network. In addition, we optimize the generator three times and the discriminator one time for each input, so as to balance the ability of the generator and the discriminator to stabilize the network training and finally output high-quality images.

4. Results and Discussion

In this section, in order to evaluate the performance of the proposed network, we performed several experiments on our own and public datasets, and used common metrics to compare the differences between the models.

4.1. Datasets

We built our own logo dataset Edges2logos dataset. We crawled about 10,000 images with the size of 120 × 120 pixels from the logo website. There are many low-quality logos among the original images, which will obviously affect the training effect. Therefore, we screened the original data and selected about 2600 high quality logo images according to our specific guidelines. We chose the image of the color with high lightness mostly. Such color is suitable for enterprises and products related to the Internet.

In addition, we pay attention to the contrast of cold and warm color pictures. Cold and warm color will bring people different psychological feelings. Cold color often makes people associate to reason and cold. On the contrary, warm color makes people feel sunshine and enthusiasm. The logo sketch is concise and easy to understand as far as possible, which is helpful for people to recognize and remember.

To train and validate our approach, we split the data into training set and testing set in a 9:1 ratio. We used OpenCV to obtain the general sketch of the logo images. To make our model more robust, we made offline argumentation of the training set. We appropriately reduced the local area of the sketch to make the training set three times larger.

Moreover, we choose the public dataset Edges2shoes to further verify the effect of colorization. The dataset consists of sketches and color images of the shoes, which contains a training set of 49,825 images and a testing set of 200 images.

4.2. Experimental Setup

Our logo GAN model is implemented using Keras [30]. The generator structure consists of $N = 3$ Att-UNet blocks and the discriminator uses a 16×16 Markovian discriminator (PatchGAN) network. During the training stage, the input image was scaled to a fixed resolution of 256×256. The batch size was set to 16, and all the image data was normalized to between 0 and 1. For the optimization, we adopted Adam [42] algorithm with momentum value of 0.5 and a learning rate of 0.0002. During the test phase, we feed the sketches of the testing set into the generator to generate the corresponding logo images. Our framework was trained on a single Nvidia Titan-X GPU.

4.3. Objective Function Analysis

A new style loss function L_{style} is added in this paper to ensure that multiple Att-UNets can output images with diverse colors. We designed a comparison experiment of two training modes, one with a style loss function and the other without. To make a fair comparison, all models use the same architecture, including a generator consisting of three Att-UNet blocks and a discriminator network with an output size of 16×16. We conducted training and testing based on edges2logo and edges2shoes datasets, respectively. The comparison results of the two mode on the testing set are shown in Figures 3 and 4. From the comparison of the images of the two modes, it is observed that the model with L_{style} loss produces higher color lightness. The color is purer and more uniform. In addition, the psychological feeling is sunny, happy and comfortable.

We extracted the background color of the logo image as shown in Figure 3. We compared the background color using the HSB (H, hues; S, saturation; B, brightness) color mode. It is shown that the S (saturation) and B (brightness) value is generally on the high side in L_{style} loss group, which means more lively color.

In addition, three objective metrics are chosen to evaluate the performance. we calculated the MAE (Mean Absolute Error) [43] value, MSE (Mean Square Error) [43] value and $SSIM$ (Structural Similarity) [44] value on testing dataset for further quantitative analysis.

MAE is computed by taking the mean of the absolute error of the generated images on a pixel level for each color channel. The formula is described as below,

$$MAE = \frac{1}{N}\sum_{i=1}^{N} ||G_i(x) - O(x)||_1 \tag{10}$$

where N indicates the number of the outputs of our proposed method, $G_i(x)$ denotes the i-th output with x as input image. $O(x)$ presents the corresponding original image in our testing dataset (edges2logos and edges2shoes).

For MSE, the l_2 norm instead of l_1 norm is used in Equation (10). The formula is described as below,

$$MSE = \frac{1}{N}\sum_{i=1}^{N} ||G_i(x) - O(x)||_2^2 \tag{11}$$

The $SSIM$ [44] is used to measure the structural similarity between reference image and target image. It contains three parts: loss of correlation, luminance distortion, and contrast distortion. The $SSIM$ is given as,

$$SSIM(R,F) = \sum_{r,f} \frac{2\mu_r\mu_f + C_1}{\mu_r^2 + \mu_f^2 + C_1} \times \frac{2\sigma_r\sigma_f + C_2}{\sigma_r^2 + \sigma_f^2 + C_2} \times \frac{\sigma_{rf} + C_3}{\sigma_r\sigma_f + C_3} \tag{12}$$

where R and F indicate the reference image and target image, respectively. r and f denote the image patch of R and F in a local window of size $h \times w$. μ_r and μ_f present the mean values of corresponding image patches. σ_r and σ_f indicate the standard deviation of reference and target image patch, respectively.

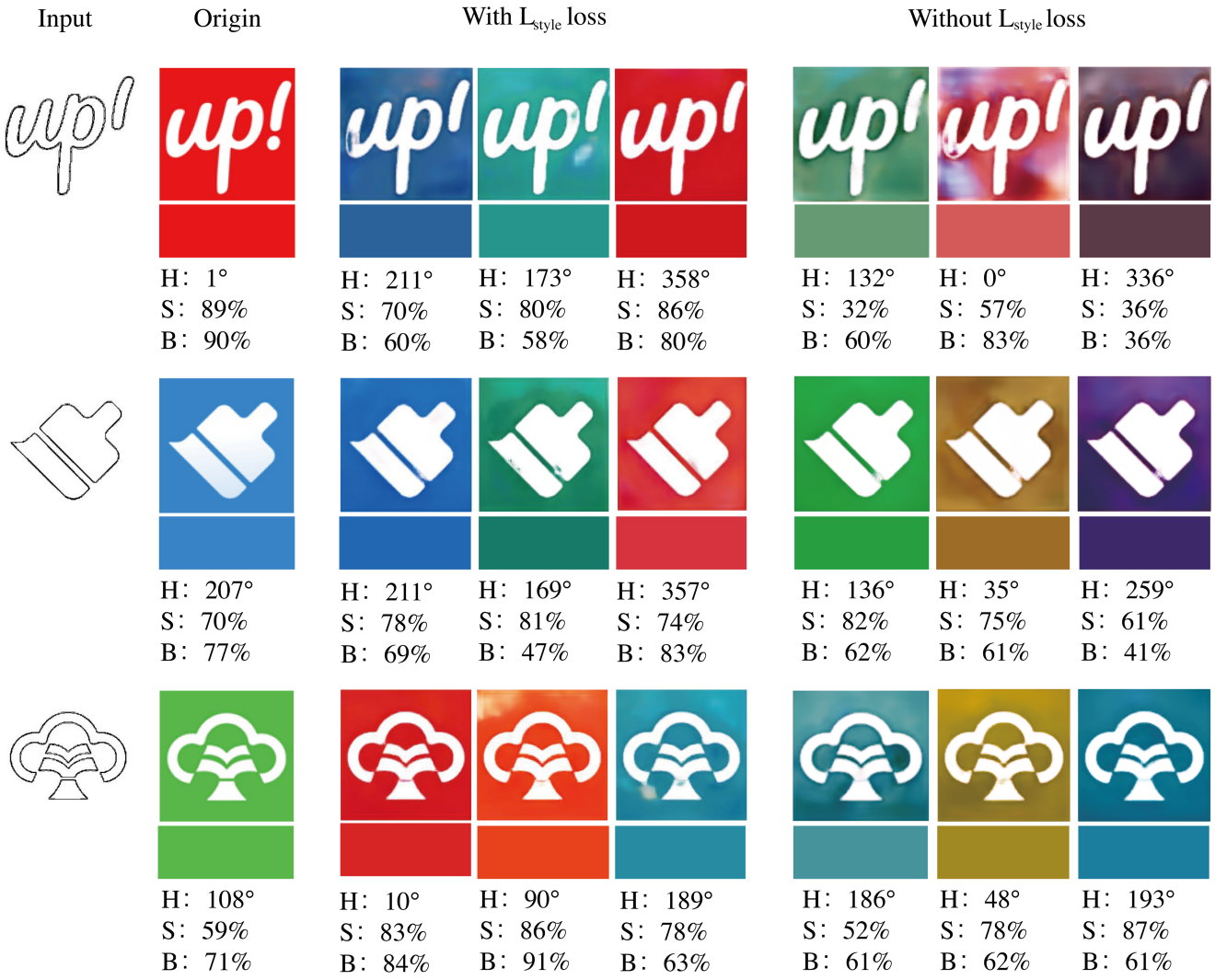

Figure 3. The comparison results of the loss functions (L_{style}) on edges2logos dataset.

Figure 4. The comparison results of the loss functions (L_{style}) on edges2shoes dataset.

σ_{rf} presents the standard covariance correlation of reference and target image patch. Thus, the first part of Equation (12) is the luminance distortion, the contrast distortion is given by the second part, the loss of correlation is denoted by the third part. C_1, C_2 and C_3 are parameters to stabilize the algorithm.

The range of $SSIM$ is [0, 1]. If the target image is more like the reference image, the value of $SSIM$ is close to 1. In our experiments, to evaluate the structural preservation performance ($SSIM$), the generate images and original images are convert to gray scale.

Table 1 shows the MAE, MSE and $SSIM$ values on the two testing sets, the best values are indicate in bold. It can be seen that the MAE and MSE values containing L_{style} loss are larger, indicating that the output images contain richer color information. The larger $SSIM$ values indicate that our network can generate richer and smoother color features, which are also presented in Figures 3 and 4.

Table 1. The MAE, MSE and $SSIM$ values for different loss functions in different datasets.

DataSet	Loss	MAE	MSE	SSIM
edgaes2logos	without L_{style}	98.87	307.64	0.64
	L_{style}	**110.89**	**572.55**	**0.77**
edgaes2shoes	without L_{style}	21.73	86.89	0.58
	L_{style}	**35.01**	**110.45**	**0.69**

4.4. Att-UNet Analysis

We integrate spatial attention and channel attention in the U-Net structure to produce more realistic and beautiful logo color. To analyze the advantages of attention mechanism in Att-UNet structure, we compared the generated images of U-Net and Att-UNet.

Figure 5 shows the generated images. Visually, the image output from U-Net network is not pure in color. The color distribution is disorderly and uneven. There is the situation of deface and blur. On the contrary, the Att-UNet images have uniform color distribution on the overall area. There is almost no pollution in the images, with higher color saturation, higher brightness, and higher purity.

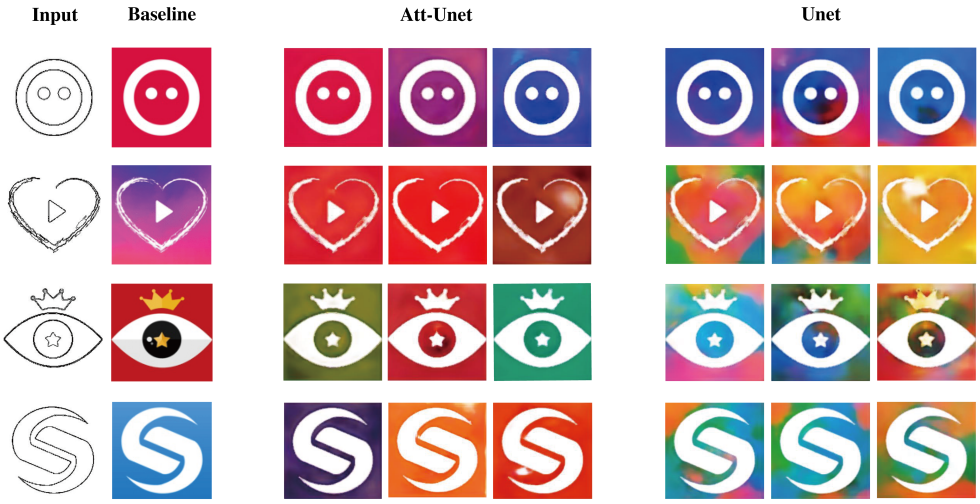

Figure 5. The comparison between attention (Att-Unet) and non-attention (Unet) mechanisms.

In addition, we output the attention weight. Figure 6 shows the attention weight heat map of some logo images, which are come from our method with CGAN and GAN, Pix2Pix [20], LogoSM [17], LoGAN [18] and LoGANv2 [19].

As shown in Figure 6, the heatmaps obtained by our method with GAN are random and the color distribution is uneven.

Compared with GAN, our method with CGAN can clearly distinguish the foreground contour area and the background area of the focus, so as to make the color distribution more reasonable and balanced, which avoid the appearance of stains.

4.5. Comparison and Analysis

For the second set of experiments, the performance of our network was compared against Pix2Pix [20], LogoSM [17], LoGAN [18] and LoGANv2 [19]. We selected some representative images for comparison. Since our network can output three images for one sketch, so we manually select the most ideal image for comparison. The generated results and their heatmaps are given in Figures 6 and 7, respectively.

In Figures 6 and 7, for Pix2Pix and LogoSM, they have good stability; however, comparing with our method, the changes in the color of the generated icons are small. The icons generated by LoGAN are blurry, with unclear shapes and colors. The resolution is also very low.

The LoGANv2 method is relatively random in the shape of the generated icon, and the colors of the two data have more impurities, which are more blurry and more difficult to identify.

Comparing with other methods, the LOGO images obtained by our method have better visual effects, such as higher color purity and brighter. In addition, they have less impurities and the overall color distribution is more uniform.

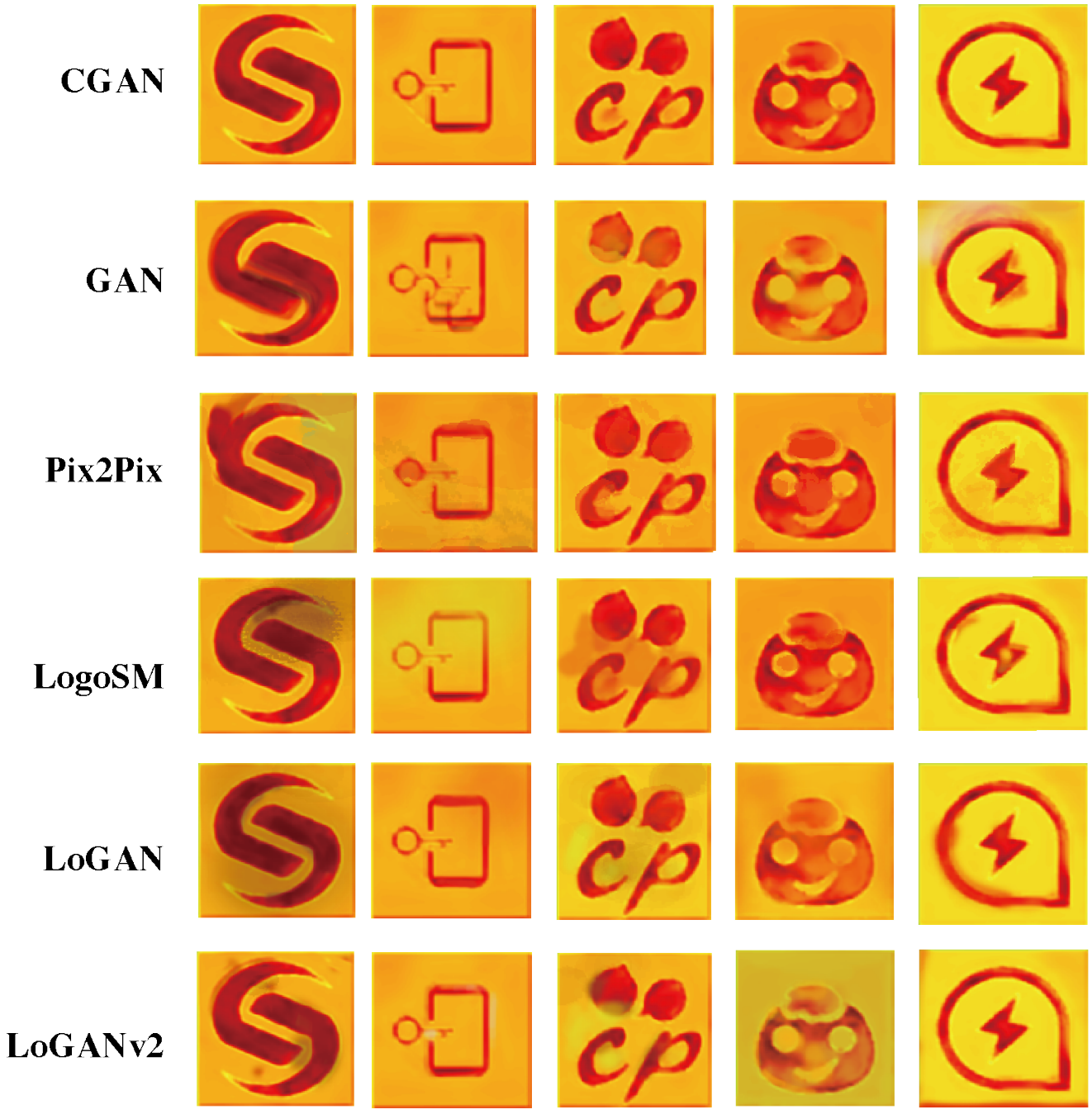

Figure 6. Heat maps of attention weight on our proposed method with CGAN and GAN and other four comparison methods (Pix2Pix, LogoSM, LoGAN and LoGANv2).

Figure 7. The comparison between our method with CGAN and GAN and other four comparison methods (Pix2Pix, LogoSM, LoGAN and LoGANv2) on two different datasets.

4.6. Future Research on Generating Colorful LOGO

As shown in the experimental results, the logo images generated by our network are low degree of color variations. These are mainly limited by the insufficient color features of the training dataset and the insufficient prior information of our network. These two points lead to the low degree of color variations of the logo images generated by our network.

To solve the above problems and generate a logo with rich colors, we believe that in our future research, our logo generate network can be improved from the following three points:

1. Increase the diversity of training dataset.
   (1) Randomly add different colors to the original data to enrich the color features of the training dataset.

(2) Add conditional information (such as semantics, combine contours and semantics) to GAN networks which is used to generate icons and logos that designers want as our training data.

2. Add prior information to the network input.
   To generate colorful logo images, the input of our network is not just the logo sketch, but also other prior information (such as the basic color provided by the designer, the desired color information).
3. Constrain the loss function of our network.
   In our method, L_{style} is used to increase the variety of logo color. On the basis of increasing the diversity of training samples, we can increase the weight of L_{style} in our loss function to expand the color diversity of network output. Thus, the color of the logo images can be enriched.

5. Conclusions

We propose a method to generate colorful logos based on logo sketch. In the generator, we modify the traditional U-Net structure to add channel attention and spatial attention during the process of skip-connection.

The attention mechanism can help to distinguish the key sketch areas and make the color distribution uniform and reasonable. Moreover, we stack multiple Att-UNet blocks so that the generator can output multiple logo images. We add a new style loss function to improve the variety of styles of output images. From the test results on the edges2logo and edges2shoes datasets, comparing with other existing methods, the output images obtained by our proposed method are reasonable and have different color, achieving good visual effects.

In the future, combining the conditional information to generate the colorful logo is still the key direction. As we discussed in Section 4.6, We can continue to combine the conditional information of multiple dimensions and improve our network architecture to generate the colorful logo.

Author Contributions: All authors contributed to this work. Conceptualization, N.T.; methodology, N.T., B.W. and Yuan Liu; software, N.T.; validation, N.T., B.W. and Y.L.; formal analysis, N.T. and B.W.; investigation, N.T. and B.W.; resources, N.T.; data curation, N.T., B.W. and Y.L.; writing—original draft preparation, N.T.; writing—review and editing, Y.L. and X.L.; visualization, N.T.; supervision, Y.L.; project administration, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Postgraduate Research & Practice Innovation Program of Jiangsu Province grant number 1132050205198113 and special funds for distinguished professors of Shandong Jianzhu University.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to some source data need to be processed. Our source codes and data will be publicly available as soon as possible.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *arXiv* **2014**, arXiv:1406.2661.
2. Wang, X.; Yu, K.; Wu, S.; Gu, J.; Liu, Y.; Dong, C.; Qiao, Y.; Change Loy, C. Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
3. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.

4. Souly, N.; Spampinato, C.; Shah, M. Semi supervised semantic segmentation using generative adversarial network. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5688–5696.
5. Luc, P.; Couprie, C.; Chintala, S.; Verbeek, J. Semantic segmentation using adversarial networks. *arXiv* **2016**, arXiv:1611.08408.
6. Zhang, H.; Xu, T.; Li, H.; Zhang, S.; Wang, X.; Huang, X.; Metaxas, D.N. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5907–5915.
7. Reed, S.; Akata, Z.; Yan, X.; Logeswaran, L.; Schiele, B.; Lee, H. Generative adversarial text to image synthesis. *arXiv* **2016**, arXiv:1605.05396.
8. Ak, K.E.; Lim, J.H.; Tham, J.Y.; Kassim, A.A. Semantically consistent text to fashion image synthesis with an enhanced attentional generative adversarial network. *Pattern Recognit. Lett.* **2020**, *135*, 22–29. [CrossRef]
9. Shi, C.; Zhang, J.; Yao, Y.; Sun, Y.; Rao, H.; Shu, X. CAN-GAN: Conditioned-attention normalized GAN for face age synthesis. *Pattern Recognit. Lett.* **2020**, *138*, 520–526. [CrossRef]
10. Zhang, H.; Sindagi, V.; Patel, V.M. Image de-raining using a conditional generative adversarial network. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 3943–3956. [CrossRef]
11. Perez, L.; Wang, J. The effectiveness of data augmentation in image classification using deep learning. *arXiv* **2017**, arXiv:1712.04621.
12. Liu, Y.; Chen, X.; Cheng, J.; Peng, H. A medical image fusion method based on convolutional neural networks. In Proceedings of the 2017 20th International Conference on Information Fusion (Fusion), Xi'an, China, 10–13 July 2017; pp. 1–7.
13. Ma, J.; Yu, W.; Liang, P.; Li, C.; Jiang, J. FusionGAN: A generative adversarial network for infrared and visible image fusion. *Inf. Fusion* **2019**, *48*, 11–26. [CrossRef]
14. Liu, M.Y.; Breuel, T.; Kautz, J. Unsupervised image-to-image translation networks. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 700–708.
15. Karras, T.; Laine, S.; Aila, T. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4401–4410.
16. Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; Aila, T. Analyzing and improving the image quality of stylegan. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 8110–8119.
17. Sage, A.; Agustsson, E.; Timofte, R.; Van Gool, L. Logo synthesis and manipulation with clustered generative adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5879–5888.
18. Mino, A.; Spanakis, G. LoGAN: Generating logos with a generative adversarial neural network conditioned on color. In Proceedings of the 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, 17–20 December 2018; pp. 965–970.
19. Oeldorf, C.; Spanakis, G. LoGANv2: Conditional Style-Based Logo Generation with Generative Adversarial Networks. In Proceedings of the 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), Boca Raton, FL, USA, 16–19 December 2019; pp. 462–468.
20. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1125–1134.
21. Kim, T.; Cha, M.; Kim, H.; Lee, J.K.; Kim, J. Learning to discover cross-domain relations with generative adversarial networks. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, Sydney, Australia, 6–11 August 2017; pp. 1857–1865.
22. Zhang, L.; Ji, Y.; Lin, X.; Liu, C. Style transfer for anime sketches with enhanced residual u-net and auxiliary classifier gan. In Proceedings of the 2017 4th IAPR Asian Conference on Pattern Recognition (ACPR), Nanjing, China, 26–29 November 2017; pp. 506–511.
23. Woo, S.; Park, J.; Lee, J.Y.; So Kweon, I. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
24. Zhang, R.; Isola, P.; Efros, A.A. Colorful image colorization. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 649–666.
25. Larsson, G.; Maire, M.; Shakhnarovich, G. Learning representations for automatic colorization. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 577–593.
26. Deshpande, A.; Lu, J.; Yeh, M.C.; Jin Chong, M.; Forsyth, D. Learning diverse image colorization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6837–6845.
27. Mouzon, T.; Pierre, F.; Berger, M.O. Joint cnn and variational model for fully-automatic image colorization. In Proceedings of the International Conference on Scale Space and Variational Methods in Computer Vision, Hofgeismar, Germany, 30 June–4 July 2019; pp. 535–546.
28. Wan, S.; Xia, Y.; Qi, L.; Yang, Y.H.; Atiquzzaman, M. Automated colorization of a grayscale image with seed points propagation. *IEEE Trans. Multimed.* **2020**, *22*, 1756–1768. [CrossRef]
29. Suárez, P.L.; Sappa, A.D.; Vintimilla, B.X. Infrared image colorization based on a triplet dcgan architecture. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 18–23.
30. Nazeri, K.; Ng, E.; Ebrahimi, M. Image colorization using generative adversarial networks. In Proceedings of the International Conference on Articulated Motion and Deformable Objects, Palma de Mallorca, Spain, 12–13 July 2018; pp. 85–94.

31. Sharma, M.; Makwana, M.; Upadhyay, A.; Singh, A.P.; Badhwar, A.; Trivedi, A.; Saini, A.; Chaudhury, S. Robust image colorization using self attention based progressive generative adversarial network. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA, 16–17 June 2019; pp. 2188–2196.
32. Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv* **2014**, arXiv:1411.1784.
33. Odena, A.; Olah, C.; Shlens, J. Conditional image synthesis with auxiliary classifier gans. In Proceedings of the International Conference on Machine Learning, Fort Lauderdale, FL, USA, 20–22 April 2017; pp. 2642–2651.
34. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2223–2232.
35. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
36. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
37. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
38. Mnih, V.; Heess, N.; Graves, A. Recurrent models of visual attention. *arXiv* **2014**, arXiv:1406.6247.
39. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In Proceedings of the International Conference on Machine learning, Lille, France, 6–11 July 2015; pp. 2048–2057.
40. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
41. Ramchoun, H.; Idrissi, M.A.J.; Ghanou, Y.; Ettaouil, M. Multilayer Perceptron: Architecture Optimization and Training. *IJIMAI* **2016**, *4*, 26–30. [CrossRef]
42. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
43. Willmott, C.J.; Matsuura, K. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Clim. Res.* **2005**, *30*, 79–82. [CrossRef]
44. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]