applied sciences

MDPI

Article

A Method for Style Transfer from Artistic Images Based on Depth Extraction Generative Adversarial Network

Xinying Han [1,*], Yang Wu [1] and Rui Wan [2]

[1] Virtual Reality Research Institute, Nanchang Hangkong University, Nanchang 330063, China
[2] Jiangxi Institute of Science and Technology Information, Nanchang 330046, China
* Correspondence: hxy_2201@foxmail.com

Abstract: Depth extraction generative adversarial network (DE-GAN) is designed for artistic work style transfer. Traditional style transfer models focus on extracting texture features and color features from style images through an autoencoding network by mixing texture features and color features using high-dimensional coding. In the aesthetics of artworks, the color, texture, shape, and spatial features of the artistic object together constitute the artistic style of the work. In this paper, we propose a multi-feature extractor to extract color features, texture features, depth features, and shape masks from style images with U-net, multi-factor extractor, fast Fourier transform, and MiDas depth estimation network. At the same time, a self-encoder structure is used as the content extraction network core to generate a network that shares style parameters with the feature extraction network and finally realizes the generation of artwork images in three-dimensional artistic styles. The experimental analysis shows that compared with other advanced methods, DE-GAN-generated images have higher subjective image quality, and the generated style pictures are more consistent with the aesthetic characteristics of real works of art. The quantitative data analysis shows that images generated using the DE-GAN method have better performance in terms of structural features, image distortion, image clarity, and texture details.

Keywords: generative adversarial network; style transfer; image processing; artistic design

Citation: Han, X.; Wu, Y.; Wan, R. A Method for Style Transfer from Artistic Images Based on Depth Extraction Generative Adversarial Network. *Appl. Sci.* **2023**, *13*, 867. https://doi.org/10.3390/app13020867

Academic Editor: Jan Egger

Received: 13 December 2022
Revised: 27 December 2022
Accepted: 29 December 2022
Published: 8 January 2023

1. Introduction

Artworks represented by drawings are the oldest form of artistic expression and an important carrier of human civilization, containing the rich and unique thoughts and emotions of their creators. Any excellent work of art contains the unique creative style of the artist. The study of the uniqueness of this artistic style is of great value to creators for improving their creative skills. In this regard, in addition to traditional art theory training, computer vision and image processing are receiving more and more attention with the rapid development and application of computer technology. The rational application of computer vision technology in the creation of artworks can help artists to systematically understand how to present a unique artistic creation style by observing real scenes or photos and using appropriate painting techniques.

In recent years, the generative adversarial network (GAN), as an important branch of deep learning, has been gaining attention from experts and scholars in the field of artificial intelligence. The GAN is a kind of generative network with antagonism, the main body of which is a generative network stacked with self-encoders, and this part of the structure is called generator; the network also has the structure of a binary classification network, and this part of the structure is called discriminator. The generator generates data close to the real sample using noise; the discriminator is used to bifurcate the generated data and the real sample and adjusts the parameters of the generator to improve the authenticity of the generated data. Through the adversarial training of the generator and the discriminator, the classification accuracy of the discriminator and the prediction accuracy of the generator

are improved at the same time; finally, the network generates data close to the real sample to accomplish the target task. Moreover, with the progress of research, more and more GAN variants have been applied, and the application of the GAN in the image field has become more and more extensive. For example, in the field of image coloring, the GAN architecture is widely used in automatic coloring models based on deep learning, etc.

GAN-based style migration strategies for artworks have also been adopted by many scholars. For example, Efros et al. [1,2] achieved the migration of painting styles to images using the texture synthesis of the underlying image features, but they ignored the semantic information of the images. Gatys et al. [3] used a Convolutional Neural Network (CNN) [4,5] to achieve the extraction of high-level semantic information from images for style transformation to show realistic images. Li et al. [6] proposed a generalized style migration method using whitening and coloring transformations to directly match the feature covariance of content and style images to achieve the single-model migration of arbitrary image styles. Zhu et al. [7] proposed a generative adversarial network named CycleGAN to implement image style migration. CycleGAN designs a cyclic consistency loss and builds a framework on which image transformation can be performed using unpaired data. DistanceGAN [8] builds on the CycleGAN architecture by adding the constraint that the distance between two samples in a domain remains constant when mapping to another domain.

With the progress of GAN research, this architecture has been employed in computer vision, medicine, natural language, processing, and other fields [9–11]. In the field of image processing, Chen et al. [12] proposed CartoonGAN, a network architecture applicable to animation style migration that works by extracting generic features of animated images and adding edge enhancement loss to GAN networks, which successfully achieves the image animation effect. He et al. [13] proposed an architecture based on an end-to-end generative adversarial network-based architecture, ChipGAN, to achieve migration from photos to traditional Chinese ink painting style. Karras et al. [14] proposed the StyleGAN image style migration method by improving the network architecture of generators, reducing feature entanglement, and improving style control. Upchurch et al. [15] put forward a method of depth feature interpolation to modify the content of images with high quality. Li et al. [16] added an optical flow regression module to the usual U-Net-like person image generation model to guide pose transformation. Hicsonmez et al. [17] proposed a GANILLA network model for style migration in children's illustrated comics, effectively improving the effect of style migration by adding jumping connections between layers of the network.

It can be seen that GAN-based image style migration methods are increasingly, widely used in the design of various types of painting artworks, and good product design results have been achieved. However, based on current research work, if the training data are sufficient, though most of the artworks generated using style migration methods can retain most of the contour and overall color information of real pictures, they are still not artwork-level perfect. For example, in some images with significant subject features and more complex semantic information, it is difficult to use style migration to match the content structure of the images, resulting in some generated images with missing and blurred local details, local color distortion, and non-emphasized image subjects. In order to better realize style migration from artworks, from the perspective of artistic deconstruction, we propose a solution for image-to-artwork style migration based on depth extraction generative adversarial network (DE-GAN). Deep learning modeling is applied to the conversion of real pictures to artwork styles; the aesthetic characteristics of artworks are combined with the elements of color, texture, shape, and spatial features, and the design is realized to generate artwork images in three-dimensional artistic styles.

2. Basic Theory

2.1. GAN

Generator and discriminator together constitute the basic GAN. The classic generator has a self-encoder as the bone network, whose function is to generate fake images using

Gaussian white noise or random noise. The discriminator is usually a classifier whose function is to clarify whether the input is from the training dataset or the generator. Therefore, the output probability of the discriminator complies with binomial distribution.

Generators and discriminators in GAN networks are trained in turn. In the initial training, the weights of the discriminator are frozen. Gaussian noise z, complying with noise distribution P_z, is fed to the generator with real images from the training dataset as the label. The generator is trained to reduce the mean square error between the output of the generator and the real image, so as to make the generator outputs fit the distribution of the real images. Then, the weights of the generator are frozen, and those of the discriminator are activated. With the discriminator trained, the classification performance is increased, and the true-positive rate of identifying the real image is improved. When training the generator again, the discriminator weight is frozen; the mean square error of the discriminator is taken as the loss function, and the generator weights are trained to improve the quality of output images from the generator. The weights of the generator and discriminator are fixed in turn so that the generator and discriminator have optimal generation performance and discrimination performance, respectively. Since this process is similar to the maximum and minimum game between the generator and the discriminator, the objective function of the GAN network is as shown in Equation (1).

$$\min_{G}\max_{D} V(D,G) = E_{x\sim P_{data}}[\log D(x)] + E_{z\sim P_z}[\log(1 - D(G(z)))] \quad (1)$$

where E_x is the mathematical expectation of the discriminator distinguishing the correct image; E_z is the mathematical expectation that the discriminator misjudges the generated image as true; z is a noise vector (such as Gaussian noise) that obeys the P_z distribution; and x is a training set image that obeys the real data distribution, P_{data}. By maximizing the discriminator (D) loss, the discriminant ability of discriminator D on the generated images is improved. This process implicates that the closer the generated image is to an image in the training set, the smaller the loss is. Therefore, the generator is not able to produce creative images. Such a system can only imitate existing images but cannot create new images. The reason is that there is a lack of motivation within the system to encourage generators to explore creative space, which is also a basic limitation of the process of artistic creation using generators versus networks.

2.2. Depth Extraction GAN

Depth extraction GAN consists of three sub-networks: multi-feature style encoder, style transfer network, and discrimination network. Among them, style encoder and style transfer network are used to form image generator G. The discrimination network is used as a discriminator to distinguish whether the generated image and style image are in the same style. Given content image $x \in X$, image content is provided; style image $y^c \in Y$ provides the style, and y^c is an image with the cth kind of style class extracted from N style images. Generator G in DE-GAN uses content image x to generate image $\widetilde{x}^c$. Discriminator D guarantees whether $\widetilde{x}^c$ is consistent with the style image, and its network structure is shown in Figure 1.

In order to assemble the contribution of the four kinds of features to the image style, four feature extractors are used to extract shape feature F_s, texture feature F_t, color feature F_c, and spatial depth F_d of the style image. Image $x_s \in R^{3\times H\times W}$ provides the style as the input of the multi-feature extractor, and the multi-feature extractor outputs are multi-meta features $F_m = \{x_s, F_s, F_t, F_c, F_d\} \in R^{C\times H\times W}$. The structure of the input style extractor is shown in Figure 2.

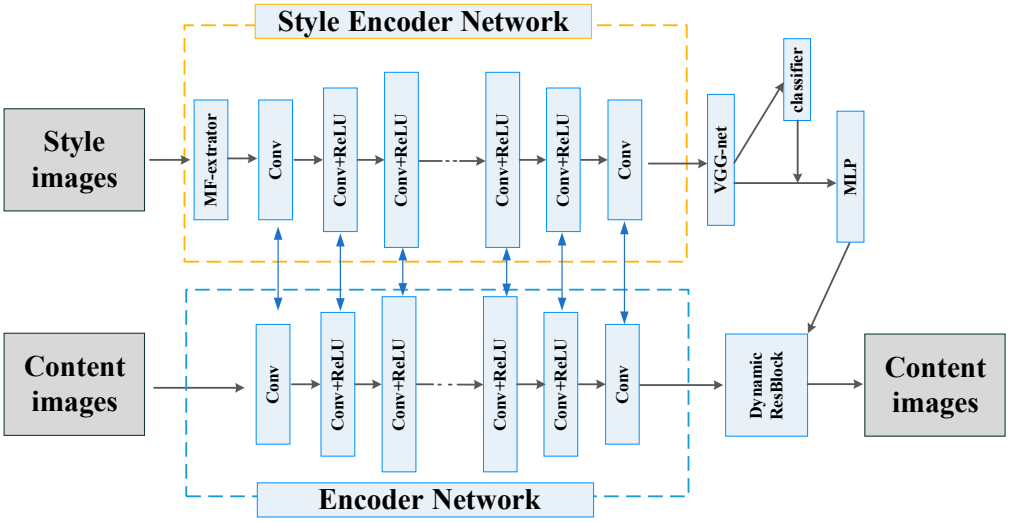

Figure 1. Structure of DE-GAN.

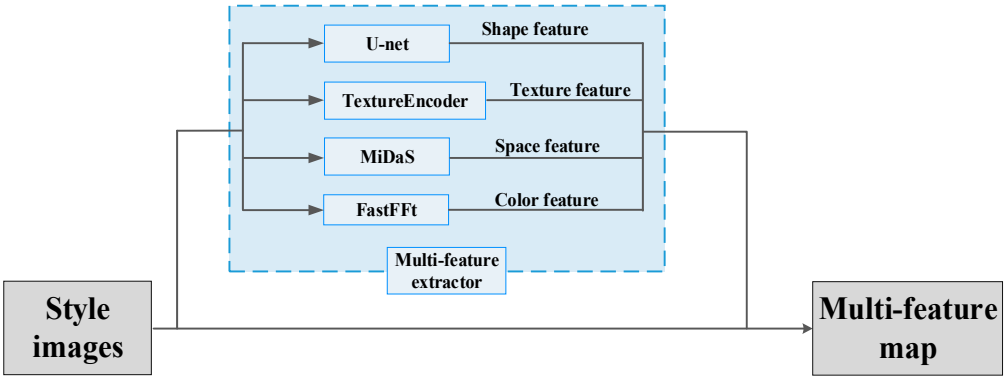

Figure 2. Structure of multi-feature extractor.

Shape feature extractor. The semantic segmentation model can quickly extract multi-object objects from the image and output the mask information of multiple objects. Due to the lack of effective panoramic semantic annotation for image datasets, the shape feature extractor is applied to a pretrained U-net model [18]. The output of the shape feature extractor is shape feature $F_s \in R^{H \times W}$. The value of F_s corresponds to the category result of each pixel with semantic segmentation. The network structure of the shape feature extractor is shown in Figure 3.

Although we use U-net as the shape feature extractor, it gives no importance to the correctness of the category of the object corresponding to the label in the mask, because the shape feature extractor plays a semi-supervised learning role here. The most valuable information in F_s refers to the segmentation boundaries of different objects.

Texture feature extractor. The shape special texture extractor can extract texture feature F_t from the original image, y^c. The texture feature is mixed with discontinuous color information, including the three channels of RGB, so $F_s \in R^{3 \times H \times W}$. In order to avoid the interference of shape features and depth features, the object in the texture extractor should be different from the original image [19]. The network structure of the texture feature extractor is shown in Figure 4. In the texture feature extractor, G^i is the loss value of texture features in different dimensions. If the weighted loss of each layer G^i is minimized, the output texture map can better express the original image texture.

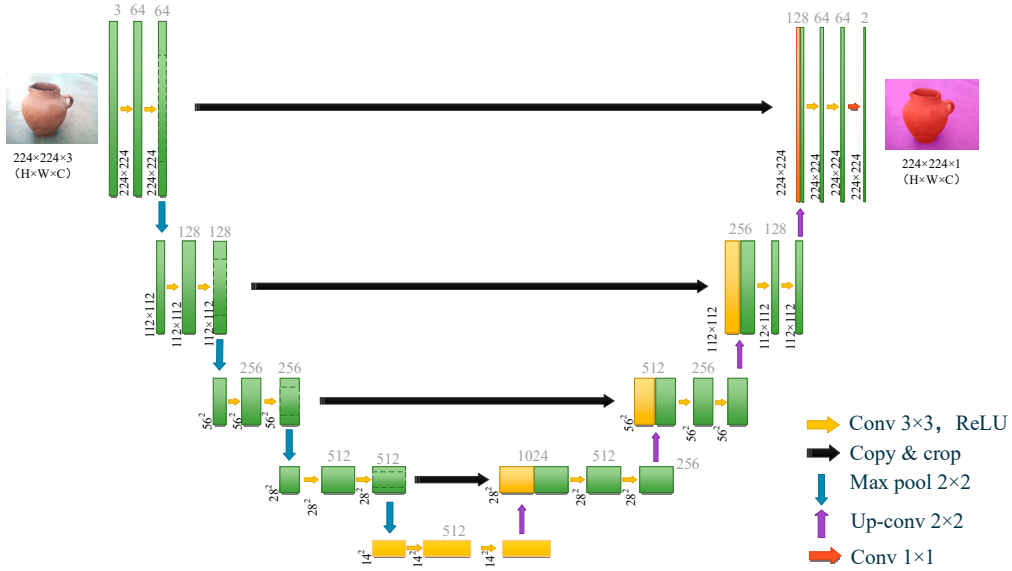

Figure 3. Structure of shape feature extractor.

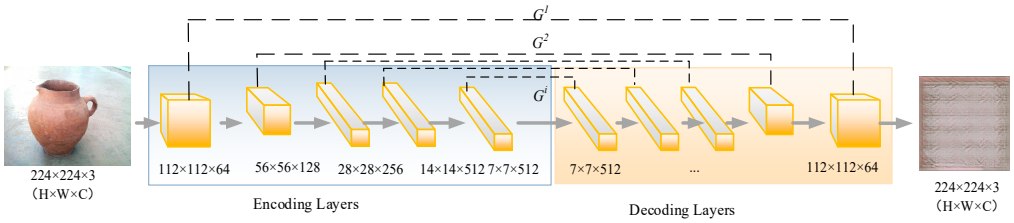

Figure 4. Structure of texture feature extractor.

Color feature extractor. Color is the visual experience resulting from the mixing of photons at different frequencies in the range of visible light. In digital images, the distribution of color can be represented by the distribution of pixel values of the three RGB channels within the range of [0, 255]. The most common feature representation method is the color histogram, but it is difficult to keep the same feature shape using color histograms in multi-feature extractors. Therefore, in the color extractor with different pixels, the fast Fourier transform is used to calculate the color histogram to calculate the color distribution of the style image, and the kurtosis maps of the three color channels are calculated as color features $F_c \in \mathrm{R}^{H \times W \times 3}$.

Spatial feature extractor. The spatial distribution of objects in artworks cannot be represented accurately in the world coordinate system with cartesian coordinates. Thus, local spatial relationships mainly refer to the pixel coordinate system and the depth values obtained by means of monocular estimation of the image.

The depth extractor is used to extract the spatial features of different objects in works of art, $F_d \in R^{H \times W}$. In the depth image, due to the limitation of pixel values, the range of depth values is [0, 255]. In order to recover the real depth information from the depth map, the output of the depth estimation network can be de-normalized [20]. The network structure of the spatial feature extractor is shown in Figure 5.

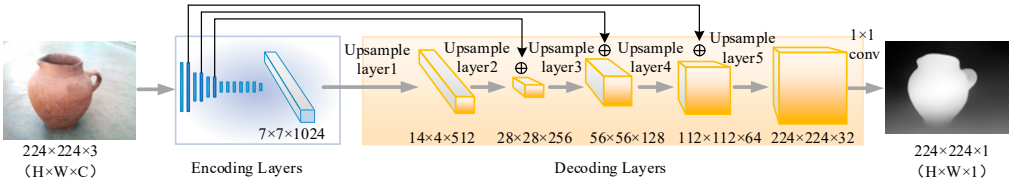

Figure 5. Structure of depth feature extractor.

Through the extraction performed by the multi-feature extractor, multi-channel feature F_m is obtained. F_m is used as the input to the style coding network.

We use style codes for the shared parameter of the dynamic residual block (Dynamic Res-Block) in the style transfer network. Dynamic convolution (D-Conv) and adaptive instance normalization structure (Ada-IN) are used in the dynamic residual block structure. The style coding network generates style codes based on the style image.

We connect pre-trained VGG-net and the self-learning encoder in parallel; the parameters of VGG-net are frozen, while the parameters of the self-learning encoder are kept active. Style coding is used in the migration network of content images. VGG-net is pre-trained on the COCO dataset; this is so that VGG-net learns the rich image textures and materials in the COCO dataset and thus has the generalization ability to extract image textures. Considering the gap between the COCO dataset and the images of different works of art, the encoding network with frozen parameters is not enough to adapt to complex style feature extraction. Therefore, a self-learning encoder is introduced as a supplement to VGG-net to realize the extraction of different styles.

Inspired by class activation mapping [21], we use a classification weight s_c to recalibrate our style code F_s. The attention mechanism is based on the trained auxiliary classifier, D_{cls}, to predict style classification probability w^c. w^c is used to represent the maximum likelihood estimation that the input style image belongs to the cth class. The style code is recalibrated to

$$s^c = w^c F_s \tag{2}$$

The recalibrated style code is used as the input of multilayer perceptron H, and the output of H is used as the shared parameter of the dynamic residual block.

Considering that style coding is applicable to any image style, two multilayer perceptron style codes are used as shared parameters for the DConv layer and the AdaIN layer of the dynamic residual blocks.

$$\left\{\theta_w^c, \theta_{\gamma,\beta}^c\right\} = \left\{H_w(s^c), H_{\gamma,\beta}(s^c)\right\} \tag{3}$$

where θ_w^c is the filter weight of the DConv layer and $\theta_{\gamma,\beta}^c$ is the affine parameter of the Ada-IN layer.

We adapt a weighted average strategy to extend the arbitrary style code of set style transformation. Specifically, we calculate the overall style code as the weighted average and the corresponding weight of the "style code" of several representative works by the same artist. For the kth style image in the set, its weight is determined according to the similarity between the style image and the content image. Therefore, we can express the overall style code as

$$\{\overline{\theta}_w^c, \overline{\theta}_{\gamma,\beta}^c\} = \{\frac{1}{K}\sum_{k=0}^{K} \pi_k \theta_{wk}^c, \frac{1}{K}\sum_{k=0}^{K} \pi_k \theta_{\gamma k}^c | c \sim N\} \tag{4}$$

where K is the number of style images used to calculate the average weight in the test phase and c is the style type of the target style domain.

2.3. Loss Function

Global loss function L is the weighted sum of loss L_{adv} of adversarial capability, perceived loss L_{per}, and style classification loss L_{cls}. Moreover, λ_{per} and λ_{cls} are the weights of L_{per} and L_{cls}, respectively.

$$L = L_{adv} + \lambda_{per} L_{per} + \lambda_{cls} L_{cls} \tag{5}$$

L_{adv} is the loss value of the discriminator, which is used to distinguish whether the generated image and the input batch of the original style images belong to the same style category.

$$L_{adv} = E_{y^c, y_i^c \sim Y, c \sim N}[-\log D(y_i^c, \{y_i^c\}_{i=0}^M)] + E_{\tilde{x}^c \sim G(x), y_j^c \sim Y, c \sim N}[-\log(1 - D(\tilde{x}^c, \{y_j^c\}_{j=0}^M))] \quad (6)$$

where M is the number of style images with value. When $M > 2$, anti-loss L_{adv} should have a good convergence effect with iteration.

L_{per} of perceived loss can be expressed as follows:

$$L_{per} = \lambda_c L_c + \lambda_s L_s \quad (7)$$

where L_s is calculated with the mean and label difference of style features.

$$L_s = E_{l \sim Nl}((\mu_{y^c}^1 - \mu_{\tilde{x}^c}^1)^2 + (Gram_{y^c}^1 - Gram_{\tilde{x}^c}^1)^2) \quad (8)$$

Content loss L_c is the expected L_2 distance between the target feature (the extracted feature of the style image) and the output image feature.

$$L_c = E_{x \sim X, c \sim N} \| \phi(x) - \phi(\tilde{x}^c) \| \quad (9)$$

where $\phi(x)$ represents the characteristics of layer 41 of Relu layers. When training the network, the style image is randomly selected from the target style dataset. The content image is transferred to the target style domain, and all the mappings in multiple artistic style areas are learned.

Loss of style classification L_{cls} uses discriminator function D_{cls} to determine whether the style category of the generated image is correct.

$$L_{cls} = E_{y \sim Y, c \sim N}(-\log D_{cls}(c|y^c)) \quad (10)$$

3. Process of Artistic Style Transfer Based on DE-GAN

We propose DE-GAN to transfer artistic design styles, which mainly integrates U-net network, MiDaS network, texture extraction network, and color histograms. It is used for style transfer from works of art using four features: shape, texture, color, and space. After determining the content image and style image, the network art image is used to train DE-GAN. In each iteration of training, two groups of pictures are randomly selected as DE-GAN style image input x_s and content image input x_c. The network is generated to generate the image, and network extraction is lost to generate the image in the training process; then, the generator generates image x_g. When the discriminator loss of DE-GAN is no longer reduced, the DE-GAN model with optimal weight is obtained. At this time, the style image of the target is used as the style input, and the target content image is used as the content input; DE-GAN outputs the migrated artistic appearance image. The whole process is shown in Figure 6.

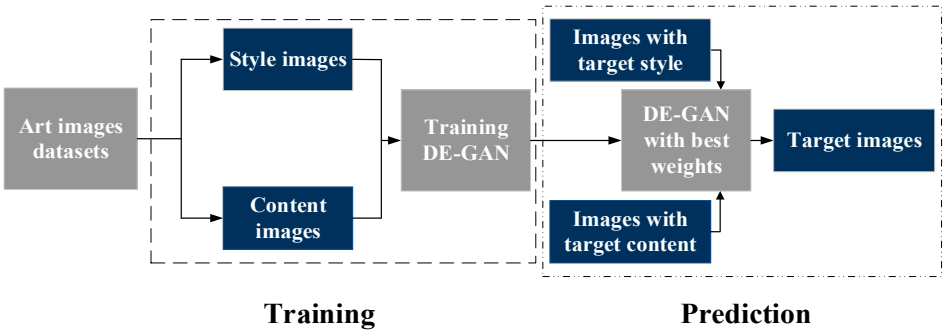

Figure 6. Tree of artificial style transfer with DE-GAN.

The result of applying our method to a real image is shown in Figure 7. The left pictures are the real pictures, while the middle images are the pictures of the selected styles, and the right images are the pictures obtained by applying the method in this paper. The styles chosen are the oil painting style and the sketch style. Through the subjective feelings provoked by the generated images, we can see that this method can be applied to different scenes for photo-generated color unity, style harmony, and details, preserving the integrity of the artistic image; the method can keep the structure of the real picture as much as possible and realize the artistic style of the picture. Further subjective and objective comparative analyses are introduced in detail in the next part of the experimental analysis.

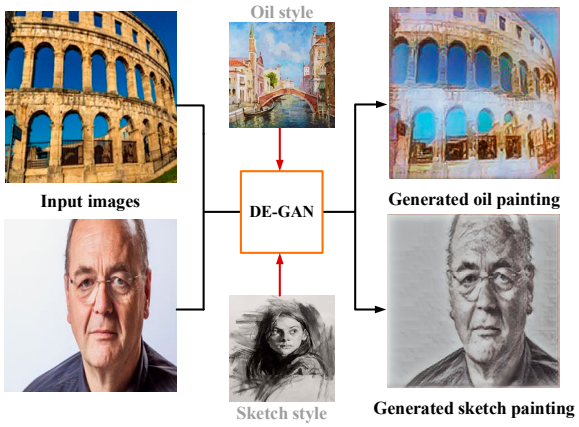

Figure 7. Results of artistic style transfer with DE-GAN.

4. Experiment and Discussion

4.1. Convergence of Loss Function

The loss function curve corresponding to the training of DE-GAN is shown in Figure 8. With the iteration of training, the confrontation loss keeps decreasing and tends to converge, which indicates that the training of DE-GAN tends to be stable and that the migration ability of the network gets stronger. Content loss reflects the consistency of image content of the generated image and content image, and the smaller the content loss is, the more consistent the style is. The content loss keeps decreasing until convergence, which indicates that the difference in content between the generated image and content image is gradually reduced and stabilized while DE-GAN learns migration ability. Style loss reflects the style loss between the generated image and the style image, and the reduction in style loss indicates that the probability of the generator "cheating" the discriminator increases and finally stabilizes, after which the style learning ability of DE-GAN is not further improved to prevent the transition of the learning style from causing content loss and confrontation loss.

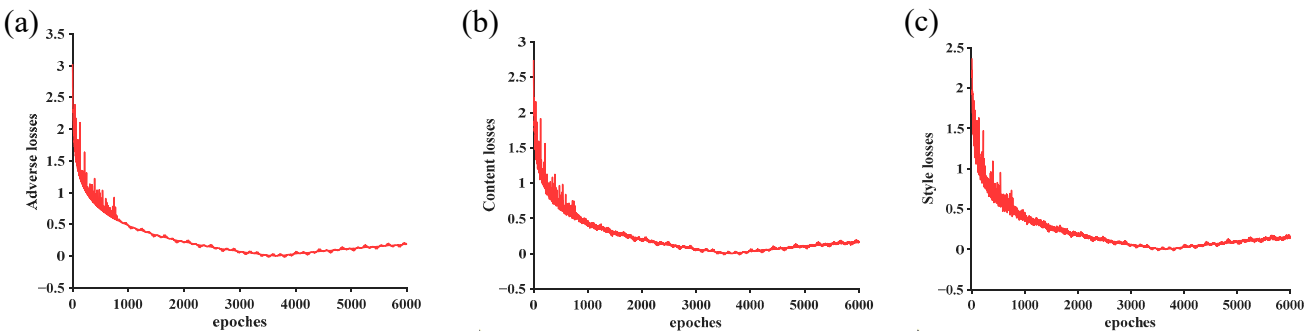

Figure 8. Convergence diagram of loss functions: (**a**) adverse image loss; (**b**) content image loss; (**c**) style image loss.

4.2. Experimental Setup

The hardware environment used for the experiments was a A4000 graphics card with 16G video memory, and the deep learning framework was PaddlePaddle. We used the Kaggle open-source artistic image dataset and part of the image material collected by the authors from the network as the training set for DE-GAN, which contained 4410 sample images of artworks; the artistic styles covered sketches and reliefs. The training set contained 4410 sample images of artworks, covering various types of art, such as drawings, reliefs, prints, paintings, and sculptures. In the experiments, Batchsize was set to 4; the learning rate was set to 0.0002; and the learning rates of the generator and discriminator were set to 0.0005. The Adam optimizer was used to optimize the network.

4.3. Qualitative Evaluation

In this section, DE-GAN is compared with two state-of-the-art GAN methods, StyleGAN [12] and CycleGAN [7], using unpaired data. Figures 9 and 10 show the qualitative comparison effect diagrams of DE-GAN and the two compared methods. The same real photo was used as the input test image, and two types of artworks, sketch (Figure 9) and oil painting (Figure 10), were used as style images, respectively. It can be seen that all three methods successfully generated artistic images that were consistent with the style images. In the sketch style pictures in Figure 9, the reconstructed sketch renderings of different algorithms are subjectively visual. The vase image reconstructed using DE-GAN has a clearer visual effect, and the transition between the main content of the picture and the background is more natural and clearer. The relationship between light and shade is more harmonious, and the main content is more three-dimensional. In contrast, the image background obtained by applying the StyleGAN and CycleGAN algorithms is slightly messy, which does not conform to the characteristics of real images. In the vase image obtained by applying StyleGAN, it can be clearly seen that the overall image is blurred; the main content of the image is not prominent enough; and the main body and the background are integrated. It can be seen that the sketch style images reconstructed using the algorithm in this paper have relatively better contour edge information, and the lines contain more detailed information, which can better highlight the main content of the images and present a more realistic sense of brush strokes. Novice painters can learn the painting techniques of artists by examining the subtle differences between the original photo and our generated painting.

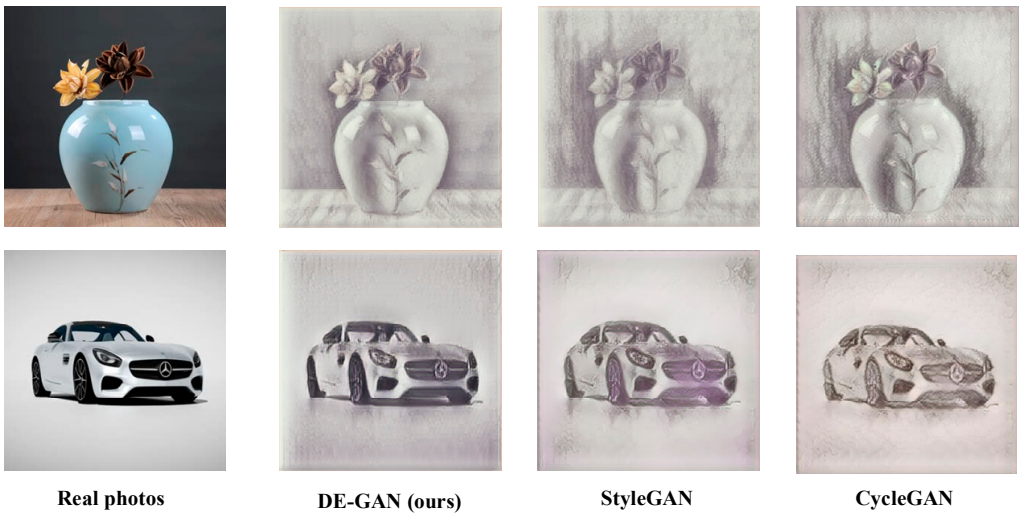

Figure 9. Comparison of subjective visual effects of sketch paintings.

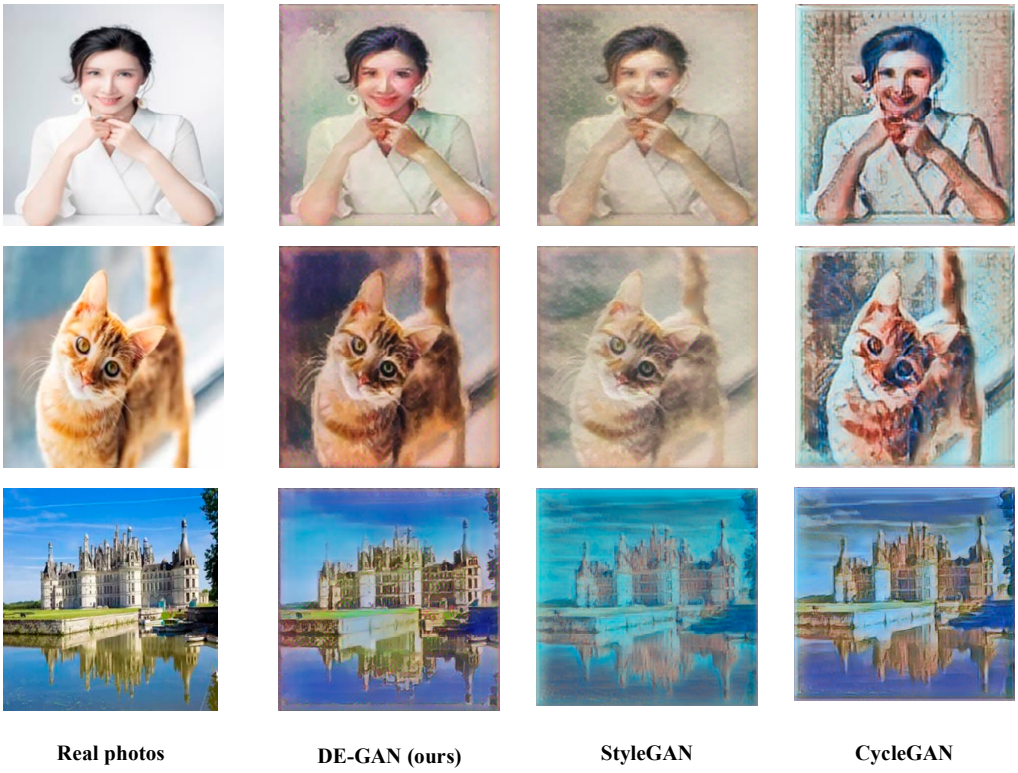

Figure 10. Comparison of subjective visual effects of oil paintings.

From the style transfer of the oil painting style shown in Figure 10, it is obvious that DE-GAN, StyleGAN, and CycleGAN could all effectively transfer the oil painting style; however, it can be clearly seen that because StyleGAN and CycleGAN only produce simple image styles, the obtained artistic images have the problems of gradation blur, image distortion, and so on, and local areas are easily over-stylized. The resulting image loses the content of the original photograph. Relatively speaking, the artistic images obtained using our method have better performance. In this method, the feature information style of the four elements is extracted, and the relevant features of the content image are matched, so that the sense of hierarchy in the content image is well preserved, while the artifacts are eliminated; thus, style images with clear structure and fine details can be obtained.

From the style transformation of the traditional Chinese ink painting style shown in Figure 11, it can be seen that although all three methods achieved style transfer, the migrated images obtained using DE-GAN and StyleGAN show contrast between the light and shade of the main body of the image and retain most of the details of the main body. This makes the generated image look clearer and more three-dimensional, which is where CycleGAN falls short. Compared with DE-GAN, the overall color obtained with the StyleGAN algorithm is redder. However, it should be noted that with the traditional Chinese ink painting style, the three methods cannot deal with the image background details and cannot keep the details of the original image background.

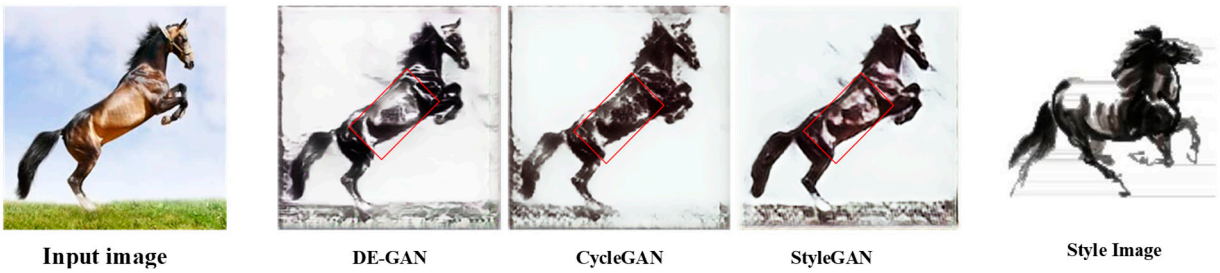

Figure 11. Comparison of subjective visual effects of ink paintings.

4.4. Quantitative Evaluation

It is difficult to objectively evaluate the algorithms using only visually subjective evaluation of the stylized images; therefore, in terms of quantitative objective evaluation, this section introduces multi-metric evaluation criteria to quantitatively compare the performance of each method in multiple aspects.

- Feature similarity index (FSIM)

In terms of objective metrics for image quality assessment, the structural similarity index method (SSIM) [22], which evaluates image quality in terms of three dimensions, image structure, brightness, and contrast, is widely used. SSIM takes values in the range of 0 to 1, and larger values indicate higher image similarity. Moreover, the feature similarity index matrix (FSIM) [23] is an extension of the SSIM based on the algorithm that considers that not all pixels in an image have the same importance; for example, the pixel points at the edges of an object are certainly more important for defining the structure of the object than the pixel points in other background areas, which can be described by the following equation:

$$\mathrm{FSIM} = \frac{\sum_{x \in \Omega} S_L(x) PC_m(x)}{\sum \sum_{x \in \Omega} PC_m(x)} \tag{11}$$

where PC_m denotes phase consistency and S_L is a function of the gradient.

- Mean SSIM index (MSSIM)

The mean SSIM index (mean SSIM; MSSIM) [24] characterizes the average of the SSIM values of content images and migrated images, and the SSIM values of style images and migrated images. The formula for calculating the index of mean SSIM can be expressed as

$$\mathrm{MSSIM}(X, Y) = \frac{1}{M} \sum_{j=1}^{M} \mathrm{SSIM}(x_j, y_j) \tag{12}$$

where X and Y are the reference image and the distorted image, respectively; and x_j and y_j are the image contents of the jth local window. Moreover, M is the number of local windows in the image.

- Image average gradient

The average gradient of the image reflects the clarity and texture variation of the image; the larger the average gradient is, the clearer the image is, and the more detailed content it contains. It can be calculated with Equation (13).

$$\mathrm{AvG} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \sqrt{\frac{\left(\frac{\partial f}{\partial x}\right) + \left(\frac{\partial f}{\partial x}\right)}{2}} \tag{13}$$

where $M \times N$ denotes the size of the image, $\frac{\partial f}{\partial x}$ denotes the gradient of the horizontal direction of the image, and $\frac{\partial f}{\partial y}$ indicates the gradient of the image in the vertical direction. Figure 12 gives the quantitative evaluation results of three methods, DE-GAN, StyleGAN, and CycleGAN, using the three metrics. At the same time, to ensure the correctness of the comparison and to reduce chance, three sets of images were used for each method in the comparative analysis.

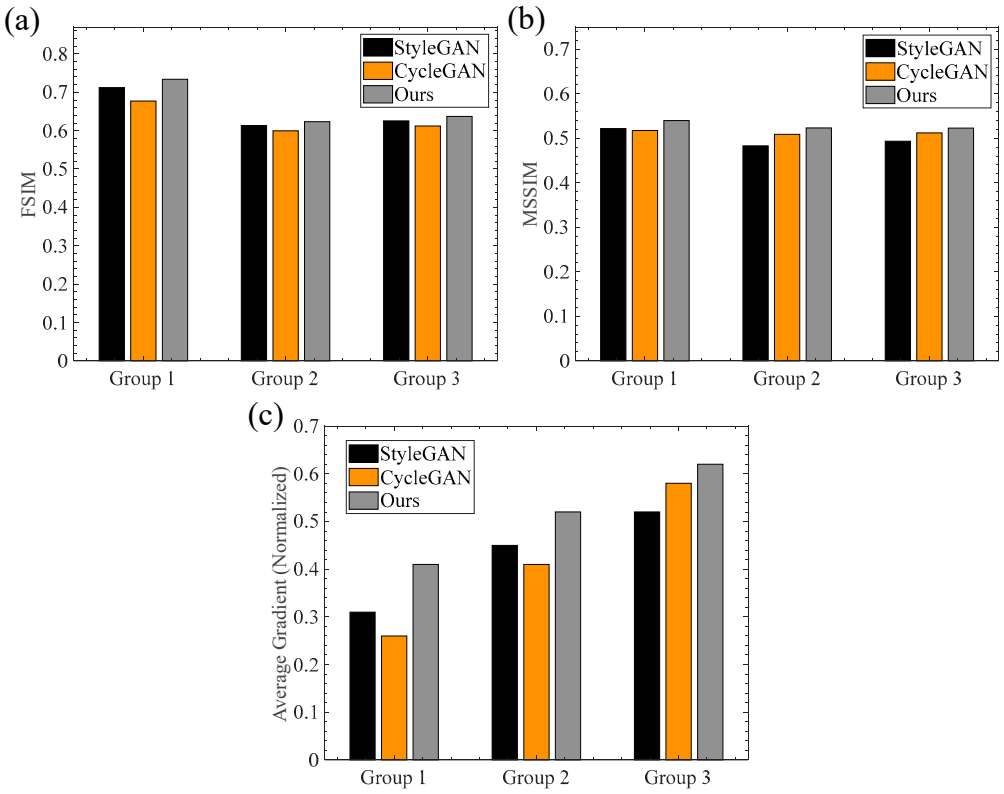

Figure 12. Quantitative evaluation results of the three methods: (**a**) FSIM; (**b**) MSSIM; (**c**) average gradient.

As can be seen from Figure 12, the migrated images obtained using the DE-GAN algorithm in this paper have small improvements in FSIM, MSSIM, and average gradient compared with the StyleGAN and CycleGAN algorithms, which indicates that the method in this paper has better performance in terms of structural features, image distortion, image sharpness, and texture details than the other three methods and achieves stylization while preserving the details of the content images to the greatest extent. However, it should be noted that due to the more complex network, the inference speed of DE-GAN is lower than that of StyleGAN and CycleGAN. The comparison of the time taken is shown in Table 1.

Table 1. Comparison of time taken by CycleGAN, StyleGAN, and DE-GAN.

Method	Average Reasoning Time for a Single Picture (ms)
StyleGAN	15.64
CycleGAN	26.78
DE-GAN	42.63

5. Conclusions

In this paper, a style migration model based on DE-GAN for images of artworks is established. Applying this model can transform real images into artworks in different style types by means of style migration. In the model, U-net, multi-element extractor, fast Fourier transform, and MiDas depth estimation network are applied as multi-feature extractors to extract color features, texture features, depth features, and shape masks from style images, which are composed of multiple features, as the input of the style extraction network. The self-encoder structure is also used as the content extraction network kernel to generate the network, which shares the style parameters with the feature extraction network and finally achieves the generation of artwork images in three-dimensional artistic styles. After an experimental comparison with StyleGAN and CycleGAN, the images generated using DE-GAN are shown to possess higher image quality subjectively, and according

to the quantitative analysis, their structural features, image distortion, image sharpness, and texture details, as well as other related indexes, are better than those obtained using other advanced methods. However, we must point out that DE-GAN is a general artistic style migration network. Compared with some special style migration methods, such as CartoonGAN, in the field of animation, the style migration network of DE-GAN has a worse migration effect than CartoonGAN.

Author Contributions: Conceptualization, investigation, and methodology, X.H.; data processing, X.H. and Y.W.; original draft, X.H.; writing and editing of draft, X.H. and R.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by General projects of Humanities and Social Sciences in colleges and universities of Jiangxi province, China (grant No. YS18222) and General Art Planning Project of Jiangxi Province, China (grant No. YG2018074).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: https://github.com/weiliao168/A-Method-of-StyleTransfer-of-ArtisticImages-Based-on-DepthExtractionGenerative-Adversarial-Network (accessed on 27 December 2022).

Acknowledgments: The authors express their gratitude to Weida Lou for editing and English language assistance.

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

References

1. Efros, A.A.; Freeman, W.T. Image quilting for texture synthesis and transfer. In Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, Los Angeles, CA, USA, 12–17 August 2001; pp. 341–346.
2. Efros, A.A.; Leung, T.K. Texture synthesis by non-parametric sampling. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Corfu, Greece, 20–27 September 1999; Volume 2, pp. 1033–1038.
3. Gatys, L.A.; Ecker, A.S.; Bethge, M. Image style transfer using convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2414–2423.
4. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
5. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
6. Li, Y.; Fang, C.; Yang, J.; Wang, Z.; Lu, X.; Yang, M.H. Universal style transfer via feature transforms. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 386–396.
7. 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.
8. Benaim, S.; Wolf, L. One-sided unsupervised domain mapping. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 752–762.
9. Yu, Y.; Gong, Z.; Zhong, P.; Shan, J. Unsupervised representation learning with deep convolutional neural network for remote sensing images. In Proceedings of the International Conference on Image and Graphics, Shanghai, China, 13–15 September 2017; pp. 97–108.
10. Mokhayeri, F.; Granger, E. A paired sparse representation model for robust face recognition from a single sample. *Pattern Recognit.* **2020**, *100*, 107129. [CrossRef]
11. Ma, J.; Yu, W.; Chen, C.; Liang, P.; Guo, X.; Jiang, J. Pan-GAN: An unsupervised pan-sharpening method for remote sensing image fusion. *Inf. Fusion* **2020**, *62*, 110–120. [CrossRef]
12. Chen, Y.; Lai, Y.K.; Liu, Y.J. Cartoongan: Generative adversarial networks for photo cartoonization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 9465–9474.
13. He, B.; Gao, F.; Ma, D.; Shi, B.; Duan, L.Y. Chipgan: A generative adversarial network for chinese ink wash painting style transfer. In Proceedings of the 26th ACM International Conference on Multimedia, Seoul, Republic of Korea, 22–26 October 2018; pp. 1172–1180.
14. Karras, T.; Laine, S.; Aila, T. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4401–4410.
15. Upchurch, P.; Gardner, J.; Pleiss, G.; Pless, R.; Snavely, N.; Bala, K.; Weinberger, K. Deep feature interpolation for image content changes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7064–7073.

16. Li, Y.; Huang, C.; Loy, C.C. Dense intrinsic appearance flow for human pose transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3693–3702.
17. Hicsonmez, S.; Samet, N.; Akbas, E.; Duygulu, P. GANILLA: Generative adversarial networks for image to illustration translation. *Image Vis. Comput.* **2020**, *95*, 103886. [CrossRef]
18. Ge, Y.; Xiao, Y.; Xu, Z.; Wang, X.; Itti, L. Contributions of Shape, Texture, and Color in Visual Recognition. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–24 October 2022; pp. 369–386.
19. 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.
20. Gatys, L.; Ecker, A.S.; Bethge, M. Texture synthesis using convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2016**, *28*, 1–6.
21. Ranftl, R.; Lasinger, K.; Hafner, D.; Schindler, K.; Koltun, V. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 1623–1637. [CrossRef] [PubMed]
22. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.
23. Sara, U.; Akter, M.; Uddin, M.S. Image quality assessment through FSIM, SSIM, MSE and PSNR—A comparative study. *J. Comput. Commun.* **2019**, *7*, 8–18. [CrossRef]
24. 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]