mathematics

MDPI

Article

Novel Creation Method of Feature Graphics for Image Generation Based on Deep Learning Algorithms

Ying Li [1] and Ye Tang [2,*]

[1] School of Design, Anhui Polytechnic University, Wuhu 241000, China
[2] Department of Mechanics, Tianjin University, Tianjin 300350, China
* Correspondence: tangye2010_hit@163.com; Tel.: +86-188-9531-6533

Abstract: In this paper, we propose a novel creation method of feature graphics by deep learning algorithms based on a channel attention module consisting of a separable deep convolutional neural network and an SENet network. The main innovation of this method is that the image feature of sample images is extracted by convolution operation and the key point matrix is obtained by channel weighting calculation to create feature graphics within the channel attention module. The main problem of existing image generation methods is that the complex network training and calculation process affects the accuracy and efficiency of image generation. It greatly reduced the complexity of image generation and improved the efficiency when we trained the image generation network with the feature graphic maps. To verify the superiority of this method, we conducted a comparative experiment with the existing method. Additionally, we explored the influence on the accuracy and efficiency of image generation of the channel number of the weighting matrix based on the test experiment. The experimental results demonstrate that this method highlights the image features of geometric lines, simplifies the complexity of image generation and improves the efficiency. Based on this method, images with more prominent line features are generated from the description text and dynamic graphics are created for the display of the images generated, which can be applied in the construction of smart museums.

Keywords: intelligent algorithms; feature graphic; image generation; deep learning networks; smart museum

MSC: 68T07

Citation: Li, Y.; Tang, Y. Novel Creation Method of Feature Graphics for Image Generation Based on Deep Learning Algorithms. *Mathematics* **2023**, *11*, 1644. https://doi.org/10.3390/math11071644

Academic Editor: Catalin Stoean

Received: 3 March 2023
Revised: 22 March 2023
Accepted: 27 March 2023
Published: 29 March 2023

1. Introduction

With the development of artificial intelligence, there is an urgent need for the construction of smart museums through generating images for historical relics such as pottery scribing symbols from the Neolithic Age. In the last decade, deep learning algorithms [1–3] have played an important role in intelligent graphic design and image generation [4]. Omri et al. [5] proposed a novel hyperparameter tuned DL technique for automated image captioning generation based on a deep learning model. Zhang et al. [6] proposed a novel knowledge-based model with adjustable visual contextual dependency to generate scene graphics. The method of generating images according to interrelated text, lines, and graphics has been a basic problem in art design and machine vision [7].

The existing image generation methods are mostly based on generative adversarial networks (GANs) [8–10], which generate images from text. The purpose of text-to-image generation is to learn the semantic alignment and multi-modal representation between image and text features [11,12]. For example, Li et al. [13] proposed a network structure based on neural architecture search to achieve image generation by searching text. Quan et al. [14] developed a new generative adversarial image generation framework based on an attention regularization module and area suggestion network to solve the generation problem

for images with complex backgrounds by locating text keywords accurately and reducing the interference of complex background information. Zhang et al. [15] presented a dual generator attention GAN based on cooperative up-sampling to generate high-quality images from description text by establishing two generators with an individual generation purpose to decouple image generation for the object and background. To achieve a more realistic visual effect of generated images, Chen et al. [16] developed a two-stage deep generative adversarial quality enhancement network to generate high-quality three-dimensional images. Zhang et al. [17] proposed an image generation network model with multiple discrimination to improve the recognition ability of the discriminator and accelerate the generator's production of high-resolution images by introducing segmented images into the discriminator. Tan et al. [18] proposed a self-supervised method to synthesize images according to a given text that is more realistic visually using self-supervised learning, feature matching, and the L1 distance function excitation generator.

In this paper, we propose a novel creation method of feature graphics by deep learning algorithms based on a channel attention module [19,20], which could be used to train the network to reduce the complexity of image generation and improve the accuracy and efficiency. Based on this method, we can generate images with more prominent line features from the description text and create dynamic graphics for the display of the images generated, which can be applied in the construction of smart museums. Through the experimental study, we found that it obviously reduced the complexity of image generation and improved the efficiency when we trained the image generation network with feature graphic maps.

2. Creation Method of Feature Graphics Based on Channel Attention Module

A channel attention module was constructed to extract the key image feature and create feature graphics for network training and dynamic graphic generation, which includes a separable deep convolutional neural network (DCNN) and an SENet network [21]. The image feature of sample images is extracted by a convolution operation within the DCNN, and the key point matrix of the convolution feature map is obtained by a channel weighting calculation within the SENet network to create feature graphics, as shown in Figure 1.

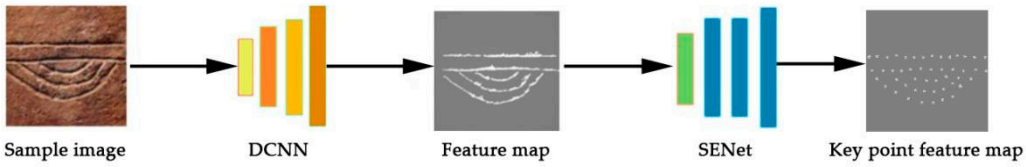


Figure 1. Feature graphic creation with the channel attention module.

This method pays more attention to the key part of images than the traditional image generation methods. In the channel attention module, partial image features are enhanced using channel weighting. Through extracting the key feature information of text and sample images, it improves the semantic matching between texts and images, reduces the complexity of image generation and improved the efficiency.

2.1. Image Feature Extraction by Convolution Operation

Considering pottery scribing symbols from the Neolithic Age as an example, most of them consist of lines and gullies formed by a pressing and carving process; hence, there is a significant brightness difference between the carved lines and surrounding part. Compared with complex images, for example, animals, plants, clothing, and portraits, the images of pottery scribing symbols from the Neolithic Age have the characteristics of geometrical form, line features, and specific types.

When we input sample images of pottery scribing symbols into the channel attention module, the image feature maps are extracted by convolution operation within the separable deep convolutional neural network. Taking image brightness feature processing for

sample images is considered to be very helpful for simplifying the image feature extraction process and improving efficiency and accuracy. The detailed steps of the method are summarized as follows:

Step 1. The image storage mode is converted from a color value matrix to a brightness value matrix, and then a linear method is used to improve image brightness and contrast. The method can be expressed as:

$$L_{i,j} = l_e \frac{R_{i,j} + G_{i,j} + B_{i,j}}{3} + l_o \quad (1)$$

where R, G, and B represent the matrices of the red color value, green color value, and blue color value, respectively, l_e is the enhancement coefficient, and l_0 is the offset value.

Step 2. The image features are extracted by the separable deep convolutional neural network in the channel attention module. Through the judgment of each weighted sum in the moving process of convolutional kernels on the original image matrix, the image brightness feature is extracted. The specific method can be described as:

$$U_{i,j} = L_{i,j} \Delta K_S(m, n) \quad (2)$$

where $L_{i,j}$ is the brightness value matrix after image preprocessing, which is input into the deep convolutional neural network, and K_S is the convolution kernel moving on the image matrix, where m and n represent the step size of the convolution kernel moving in the horizontal and vertical directions, respectively. In this study, a 3×3 convolution kernel is used for the convolution operation, and an $H \times W$ feature map is obtained, where H and W are the height and width of the feature map, respectively.

Step 3. The feature map extracted by convolution operation needs to undergo the operation of removing scattered small areas with the method of remove_small_objects (T_A), and the size of removed areas can be controlled by setting the connected area threshold T_A. After this operation, the feature map will be more clear and significant.

The operation flow and results of the above steps are shown in Figure 2.

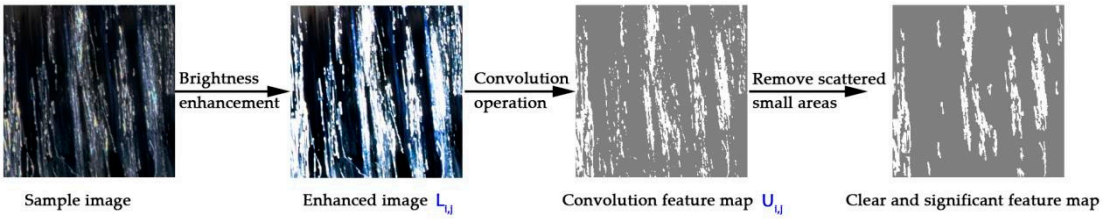


Figure 2. Operation process for image feature extraction.

2.2. Feature Graphic Creation by the Channel Weighting Calculation

When feature maps extracted by the convolution operation are directly used to train the image generation network, the major problem is that network training becomes extremely complex, with a long calculation time and low efficiency if the parameters of the convolution kernel are large. However, the complexity of model training is simplified and efficiency is improved when attention is converted to the key part of images. Hence, the image feature map is continuously input into the SENet network, where the key feature point matrix is formed as a result of the enhancement of the response values of some feature channels and the suppression of meaningless feature channels using the channel weighting method that includes three basic operations, as follows.

1. Squeeze operation

Through the global pooling operation, the information of all feature points in the space is compressed into an average value, which eliminates spatial distribution information, obtains the global view, and improves relativity between adjacent channels. The two-

dimensional feature matrix becomes $1 \times 1 \times C$ after the squeeze operation, where C is the channel matrix. The method for the squeeze operation is as follows:

$$z_c = f(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=i}^{W} u_c(i,j) \tag{3}$$

where c is the channel number, $c \in C$, u_c is the pixel value matrix of some feature point in the convolution feature map, f is the global average pooling method, and z is the output value of each channel.

2. Excitation operation

Through the excitation operation, the feature dimension of the first fully connected layer is reduced to 1/16 of the original feature dimension and the feature dimension of the second fully connected layer is restored to the same value as the original feature dimension. The weight value of each channel is calculated using the parameters, and then the parameters are normalized using the sigmoid activation function. The excitation operation process can be expressed as:

$$E = \sigma(M\delta(Wz)) \tag{4}$$

where Wz is the channel feature point matrix composed of each output from each channel, σ represents the sigmoid activation function, and δ represents the ReLU activation function. The dimension reduction operation M can be written as:

$$M = R^{\frac{c}{r} \times c} \tag{5}$$

where r is the dimension reduction ratio, and $r = 16$.

3. Scale operation

By multiplying the original feature map with the channels of the weighting matrix individually, the weight of each feature channel in the original feature map is changed, and an interesting part of the feature map is focused on. The method is as follows:

$$s_c = x_c \cdot u_c \tag{6}$$

where X is the weighting matrix and $X = [x_1, x_2, \ldots, x_c]$. Thus, x_c means the channels of the weight matrix. Through the weighting operation, some pixel points of the original image feature matrix are highlighted as the key point feature matrix.

Through the series of operations above with the convolution feature map (shown in Figure 3a), the key point feature matrix is obtained, and the key point feature map is shown in Figure 3b. To facilitate the creation of a feature graphic, we take the interval sampling in the X direction with the step of two points to the key point feature matrix, and the result is shown in Figure 3c. Then, the feature graphic is created with the method of drawMatches() by connecting the key feature points under certain conditions, for example, the distance between two points lower than a certain value, and the feature graphic obtained is as shown in Figure 3d. Then, we remove the discrete points. When we set the threshold of removing discrete points as the scope of two points, we obtain the clear graphic shown in Figure 3e.

Figure 3. Creation process of feature graphic based on image feature map: (**a**) image feature map; (**b**) key point feature map; (**c**) sampling point map; (**d**) feature graphic; (**e**) clear graphic map.

2.3. Method Summary Based on Practice of Feature Graphic Creation

Taking a semi-box pottery scribing symbol as an example, We input the sample image into the channel attention module to extract its image feature and create the feature graphic. The specific processing process within the channel attention module can be summarized as follows, which is shown in Figure 4.

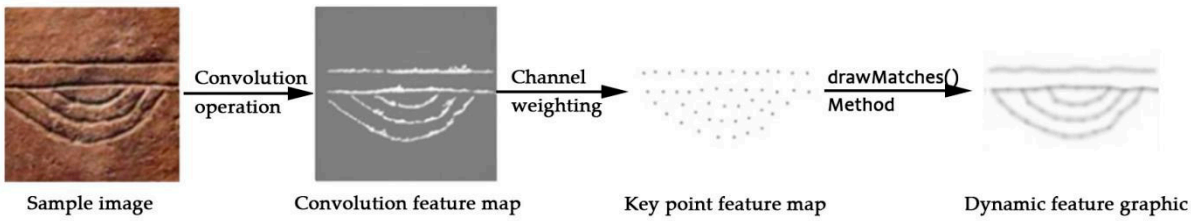


Figure 4. Feature graphic generation process by deep learning algorithms.

1. Image features of sample images are extracted by a convolution operation. It improves the image feature extraction effect that enhance the brightness feature by taking image brightness feature processing in advance. Additionally, it highlights the key image features by removing the scattered small areas afterwards.
2. A key point matrix is obtained by the channel weighting calculation including squeeze, excitation, and scale operations. It obtains the key point feature map with a relatively clear display by taking interval sampling to the key point matrix afterwards.
3. Feature graphics are created with the method of drawMatches() by connecting the key feature points with the limited conditions that the distance between two points is lower than a certain value, which is set as 6 pixels in this practice.

3. Method of Image Generation from Description Text

3.1. Text-to-Image Generation Network Model

In terms of generating images from description text, the generative adversarial networks proposed by Goodfellow et al. [22] play an important role, which is composed of a generator and discriminator. The depth generation model based on deep learning can generate visual realistic images without supervision through appropriate training. In the training process, the optimization generator captures the distribution of the sample data and generates data similar to the real training sample with noise z obeying a certain distribution, such as the uniform distribution and Gaussian distribution.

This depends on the relationship between the text and image for the image generation from the description text. However, traditional GANs do not have a clear concept of whether the training image matches the description text; hence, it is necessary to design a more complex network model for image generation from description text. Given this shortcoming, an end-to-end attention GAN was developed to the application of text-to-image generation [23]. This model enables the generation network to draw different sub-regions of the image according to the words most related to them. Simultaneously, more advanced

discriminators have been designed and applied. For example, Feng et al. [24] proposed a modal separation discriminator, which is used to distinguish the content and style in a specific layer, and enable the discriminator to capture the relevance of the text and image more effectively through the extraction of the content. The Attn GAN network mainly uses a multi-generator/discriminator structure to explore coarse/fine-grained text content, such as words and sentences, to synthesize images [25]. The accuracy of generating images from description text can be improved by establishing the corresponding relationship between the key text unit and the sample images through repeated training with the network.

For generating images from description text, a network model with channel attention is proposed. The resulting method is called CA-Attn GAN, as shown in Figure 5. When description text is input into the network model, it is encoded by the text encoder, after which the key text feature is extracted and enhanced by the SA (self-attention) and CA (condition enhancement) modules. Then, the images are generated according to the key text feature by G_i (generators at different levels), and discriminated by D_i (discriminators at different levels) to obtain its exact image features. The semantic alignment calculation between text and image is then conducted in the DAMSM module. The DAMSM module includes the text encoder and image encoder, which are used to process the mapping relationship between the text and image.

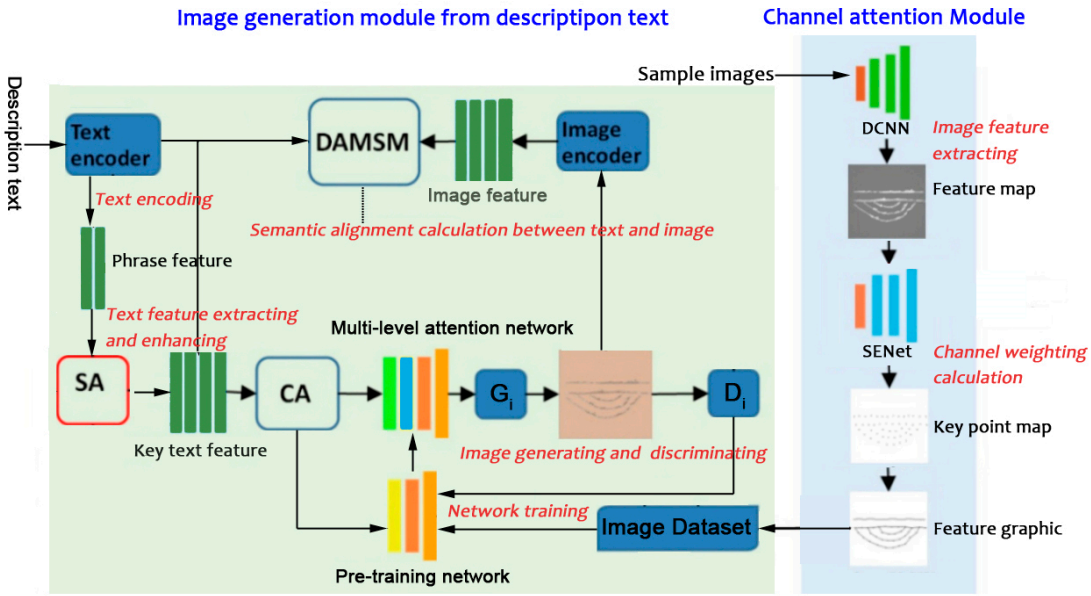


Figure 5. Image generation method from description text.

3.2. Image Generating and Discriminating

The text-to-image generation network model based on Attn GAN includes a multi-level attention network and a multi-modal similarity model with deep attention, which is called the DAMSM module. When users input a phrase or sentence into the network model, the keyword feature vector is extracted and weighted by the text encoder in the DAMSM module, which is processed by the condition enhancement module to form the text feature vector $\bar{e}$. The multi-level attention GAN is composed of several pairs of generators and discriminators that generate images at different resolution levels through multiple iterations of network model training between the generator and discriminator. The iterative relationship between each level of the GANs can be expressed as:

$$h_0 = F_0[z, F^{ca}(e)] \tag{7}$$

$$h_i = F_i\left(h_{i-1}, F_i^{atten}(\bar{e}, h_{i-1})\right) \tag{8}$$

where i represents different levels of generators and discriminators. h is the hidden node that contains the image feature as the input of the generator, z is the noise vector sampled from the standard normal distribution, F is the operation module of each level of the network for generating images, F^{ca} is the condition enhancement module, which is used to extract and weight the text feature vector e of key text, $\bar{e}$ is the enhanced text feature vector, and F^{atten} is the attention module at each level. The attention model $F_i^{atten}(\bar{e}, h_{i-1})$ has two input parameters: the word feature vector $\bar{e}$ and image features from the previous hidden layer h_{i-1}.

3.3. Text Feature Extracting and Enhancing

Pointed at the pottery scribing symbols from the Neolithic Age, the image generation depends on the text that describes the type, such as water-wave, double-circle, half-frame, box, and multi-line-fish. There is a clear correspondence between the type-description text and sample images. These type-description texts play a key role in image generation, and are weighted in the text feature extracting process. We have listed at least 25 type-description texts, and some of them include sub-types for monomer image generation. Additionally, some pottery scribing symbols are composed of several monomer images. Their type-description texts may include several type-description texts. Because each type or sub-type corresponds to a certain number of pottery scribing symbols, the images generated according to a type-description text may be random and uncertain, but those images belong to the same type and have similar image features.

When generating images from description text, the entire description text is usually converted into conditional variables as the input of the generator while processing text information. However, due to the particularity of the research object in this study, only the type-description text affects the image generation. Hence, the type-description text is weighted and the text feature is enhanced to improve the accuracy of image generation. Additionally, one sentence may contain several type descriptions. An example of processing description text when a user inputs the text "cross circle triangle combined pottery scribing symbol" is shown in Figure 6. First, the entire sentence is processed into word fine-grained information and the self-attention mechanism focuses on the key word information of the type-description text. The type-description text is weighted and the text feature vectors are extracted by the text encoder.

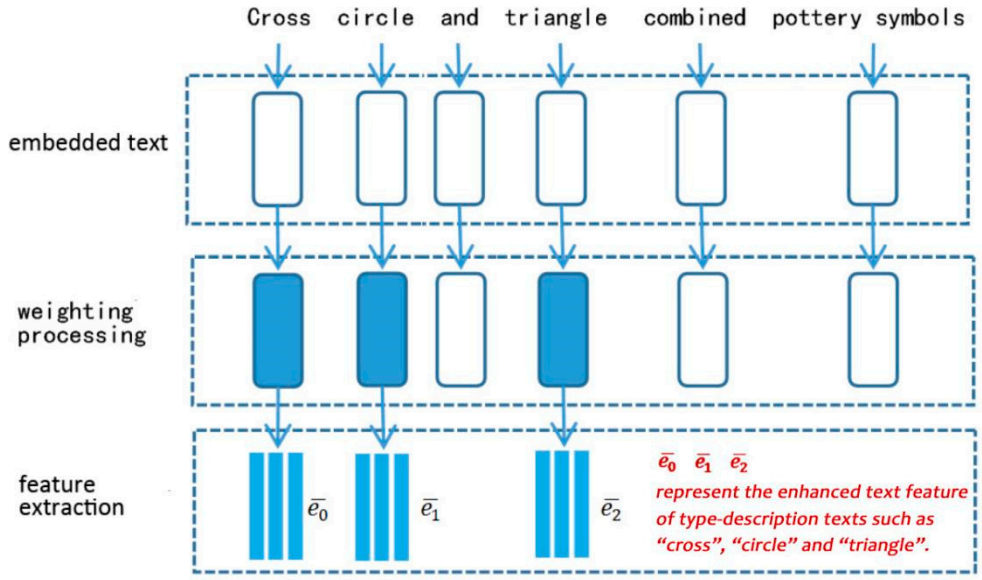


Figure 6. Text coding and feature vector extraction.

In the proposed network model, a self-attention module is introduced to extract key text information based on word granularity, and combined with the condition enhance-

ment module to enhance the text feature vectors of the type-description text. The specific steps can be expressed as follows:

1. The LSTM text encoder is applied to encode the sentences or phrases in the description text, and then they are divided into a text matrix that includes several text units according to semantics, which can be described as:

$$s = \{q_1, q_2, \ldots, q_n\} \tag{9}$$

where s is the encoded text feature, $q_i (i = 1, 2 \ldots, n)$ represents the phrase feature divided from the sentence, and n is the number of phrases.

2. The word matrix is weighted by the text encoder. The word context vector for each sub-region of the image is calculated by the hidden layer feature. Additionally, the calculating method for the jth sub-area can be described as:

$$w_j = \sum_{i=0}^{T-1} \beta_{j,i} e_i \tag{10}$$

where:

$$\beta_{j,i} = \frac{\exp(s_{j,i})}{\sum_{k=0}^{T-1} \exp\left(s_{j,k}\right)} \tag{11}$$

where $\beta_{j,i}$ is the weight of the ith word. $s_{j,i} = h_j^t e_i$, e_i is the text feature vector of the ith word, and h_j^t represents the hidden layer feature from the type-description text feature calculated by Equation (7).

3. The A1-dimensional deep convolutional neural network is applied to perform the convolution operation on the weighted text matrix q_i^w , and the result is input into LSTM to encode and extract the key text feature vector matrix $q_i^{w,k}$. For the ith text unit, the extracted text feature vector can be expressed as:

$$\bar{e}_i = f_i\left(q_i^{w,k}\right) = f_i(\tanh(q_i^w \Delta K_L + b)) \tag{12}$$

where Δ represents the convolution operation, K_L is the convolution kernel, b is the offset value, and f is the operation of text coding and feature extraction in LSTM.

After a series of operations, the key text units are strengthened and their feature vectors are extracted as the matching condition for image generation.

3.4. Semantic Alignment Calculation between Text and Image

Next, the text feature vectors and image feature vectors extracted by the image encoder are mapped into the semantic vector space in the DAMSM module. Additionally, feature matching between the text and image is achieved using the correlation calculation. Generating images from description text is mainly based on sentence or keyword retrieval to achieve the alignment of words and images [26].

The word features are converted into a common semantic space of image features by the self-attention module combined with the DAMSM module. For this method, the word context matrix for the image feature set is provided as:

$$F^{atten}(\bar{e}, h) = (c_0, c_1, \ldots, c_{t-1}) \in R^{D \times T} \tag{13}$$

where $R^{D \times T}$ is the word-text feature matrix, D is the dimension of the word vector, and T is the number of words.

The image encoder is a convolutional neural network that maps image feature vectors to text semantic vectors. The matching loss of the fine-grained text-to-image calculated by the DAMSM module is as follows:

$$\tau = -\sum_{i=1}^{n} \log P(S_i/M_i) - \sum_{i=1}^{n} \log P(M_i/S_i) \tag{14}$$

where P represents the matching probability between the text vector S_i and image vector M_i. n is the number of image vectors or the corresponding text semantic vectors. The DAMSM is trained to minimize the difference between the image vector and the word vector described above.

3.5. Loss Calculation for Image Generation

To synthesize a real image with multiple condition levels (multiple key word text conditions), the loss function of the generator is defined as:

$$L = L_G + \lambda L_{DAMSM} \tag{15}$$

where:

$$L_G = \sum_{i=0}^{m-1} L_{G_i} \tag{16}$$

and:

$$L_{G_i} = -\frac{1}{2} E_{x_i \sim p_{G_i}}[ln\ D_i(\ x_i)] - \frac{1}{2} E_{x_i \sim p_{G_i}}[ln\ D_i(\ x_i, \bar{e})] \tag{17}$$

where L_{DAMSM} is the loss function obtained using the pre-training network and λ is a hyperparameter that determines the influence of the DAMSM module on the loss function of the generator. $D_i(i = 0, 1, \cdots, m-1)$ represents the discriminators at different stages. $G_i(i = 0, 1, \cdots, m-1)$ represents the generators at different stages, and P_{Gi} represents the distribution of images generated from G_i. $x_i(i = 0, 1, \cdots, m-1)$ represents the images generated from different generators, which can be expressed as:

$$x_i = G_i(h_i) \tag{18}$$

4. Network Training and Test Experiment

4.1. Network Model Training with the Feature Graphic Maps

The network training for generating images relies on an image dataset. Elasri et al. [27] discussed various image datasets that are suitable for types of image generation and developed evaluation metrics to compare their performance. In this study, we selected the MNIST dataset to import feature graphic maps of pottery scribing symbols, train the network, and generate images. We selected 1293 representative scribing symbols from images gathered from 4 famous cultural sites of Shuangdun, Houjiazhai, Liulinxi, and Liuwan from the Neolithic Age, and then divided them into 6 types and dozens of sub-types for each type. The category and number of each type are shown in Table 1.

Table 1. Sample image data of pottery scribing symbols.

Line-Shape (249)	Polygon-Shape (201)	Circular Arc-Shape (288)	Cross-Shape (235)	Chinese Character-Shape (112)	Pictographic-Shape (208)
Horizontal line (66)	Triangle (41)	Dot (16)	Cross (103)	万 shape (32)	Animal (107)
Vertical line (65)	Box (109)	Semi-circle (19)	Grid (48)	田 shape (51)	Plant (36)
Polygonal line (43)	Semi-box (31)	Circle (111)	Fork (84)	人 shape (29)	House (26)
Number (39)	Polygon (20)	Arc (58)			Other (39)
Hook (19)		Wave (31)			
Arrow (17)		Fish (53)			

In the next step, we expanded the sample image dataset to more than 5000 pottery scribing symbols. We generated feature graphics using a channel attention module and imported them into the MNIST dataset for network model training. We provided each feature graphic map with a corresponding type-description text, such as "semi-circular," "cross," and "triangle". We composed the type-description text out of combined graphics from individual graphics. We trained the network with 5000 feature graphic maps and tested the network model on the image dataset.

4.2. Test Experiment on Image Dataset

We tested various network models on the expanded MNIST dataset and the test results are shown in Table 2, in which the "Inception Score" and "Wasserstain-2 distance" indexes can be used to measure the image quality and diversity of image generation (the results reveals better when the "Inception Score" gets higher and the "Wasserstain-2 distance" gets lower), and "R-precision" can be used to measure the accuracy of image generation. Thus, the accuracy and efficiency improvement of image generation by the proposed method compared with other network models was estimated and is shown in Table 3. From the experimental results and data analysis, we could conclude that the CA-Attn GAN significantly improved the accuracy and efficiency of image generation for pottery scribing symbols compared with the existing image generation methods of GAN and Attn GAN.

Table 2. Test results on different network models.

Method	Inception Score	Wasserstain-2 Distance	R-Precision (%)
GAN	3.52 ± 0.03	28.02	52.23 ± 3.42
Attn GAN	4.08 ± 0.05	16.58	66.52 ± 4.28
CA-Attn GAN	6.34 ± 0.03	14.72	79.06 ± 4.64

Table 3. Accuracy and efficiency improvement compared with other network models.

Method	Image Quality and Diversity	Accuracy and Efficiency
Compared with GAN	47.47–80.11%	33.73–71.48%
Compared with Attn GAN	11.22–55.39%	15.11–34.48%

Furthermore, to study the role of multi-level attention modules further, we tested the proposed network model by changing the super-parameter of the DAMSM influence degree λ. By analyzing the results shown in Table 4, we concluded that the super-parameter of the DAMSM influence degree affected the experimental results and achieved the best result when $\lambda = 5$.

Table 4. Test results under different super-parameters of DAMSM impact degree.

Method	Inception Score	Wasserstain-2 Distance
multi-level attention module, $\lambda = 0.1$	4.08 ± 0.05	18.42
multi-level attention module, $\lambda = 1$	4.32 ± 0.04	16.43
multi-level attention module, $\lambda = 5$	4.56 ± 0.04	15.64
multi-level attention module, $\lambda = 10$	4.23 ± 0.05	16.58

When $\lambda = 5$ and the dimension reduction ratio is $r = 16$, we continuously tested the proposed network model by changing the channel number of the weighting matrix c from 16 to 128 to study the role of the channel attention module. The results are shown in Table 5.

We found that the accuracy of image generation increased (CC-Loss decreased) when we increased the number of matrix channels. In the intervening time, the computational complexity and burden of the CPU also increased (GFLOPs increased).

Table 5. Test results under different channel numbers of the weighting matrix.

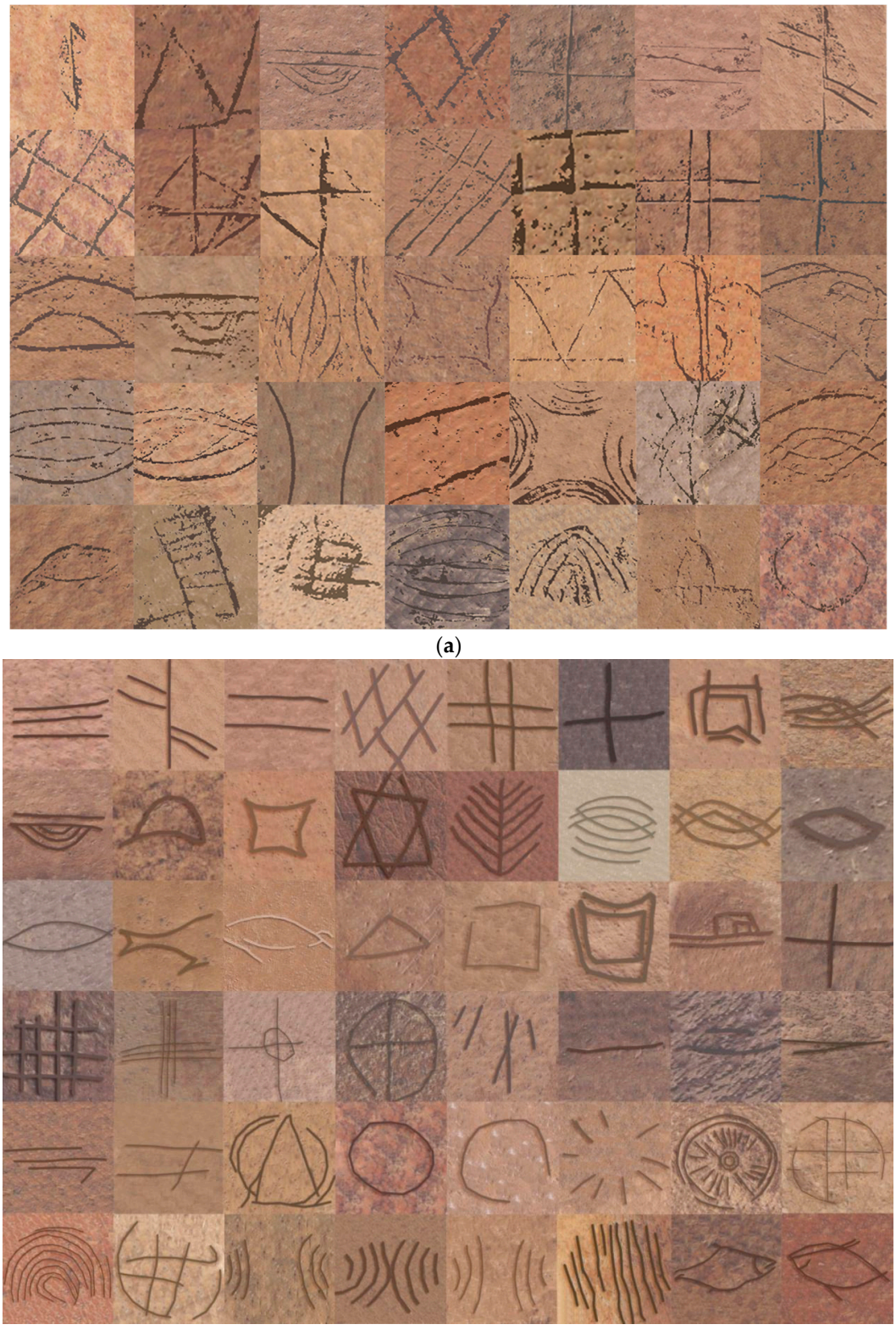
Method	CC-Loss	GFLOPs
f_c, [16, 256]	98.72 ± 0.08	8.54
f_c, [32, 512]	98.53 ± 0.09	11.87
f_c, [64, 1024]	97.87 ± 0.07	13.24
f_c, [128, 2048]	97.42 ± 0.06	14.98

4.3. Comparative Experiment for Image Generation

We also found that the text-images generated by the proposed model had more geometric graphic features than others, which is more convenient for the creation of dynamic feature graphics. In the same experimental environment, we generated a set of images using both the Attn GAN and CA-Attn GAN. For the DAMSM impact degree $\lambda = 5$, the images generated by the Attn GAN is shown in Figure 7a. For the channel number of the weighting matrix $c = 64$, the images generated by the CA-Attn GAN are shown in Figure 7b.

Taking the fish-shape scribing symbols belonging to the circular arc-shape group as an example, when users input a phrase or sentence including the type-description of "fish", it may generate images as shown in Figure 8a with the multi-level attention network, and generate images as shown in Figure 8b with the channel attention network.

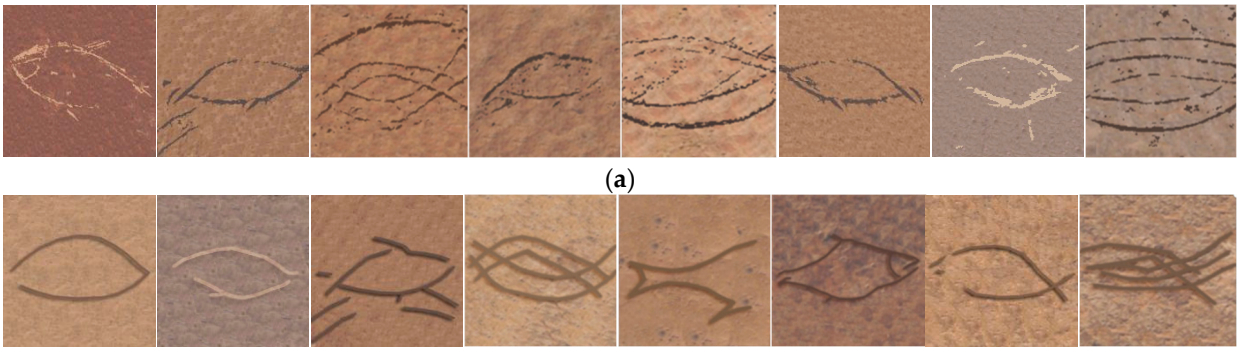
By comparing all the experimental results, we found that the images generated by the proposed network model had more significant image features, with clear geometric lines and more prominent key feature points. However, the image details were not as rich and real as the images generated by the Attn GAN. We can conclude that the proposed network model is more effective for the image generation of traditional scribing graphics with a simple form, line features, and specific type. The advantage of this method is that it highlights the image features of geometric lines, simplifies the complexity of image generation, and is more convenient for generating dynamic feature graphics.



(**a**)

(**b**)

Figure 7. Images generated by various network models. (**a**) Images generated by the multi-attention network; (**b**) images generated by the channel attention network.



(**a**)

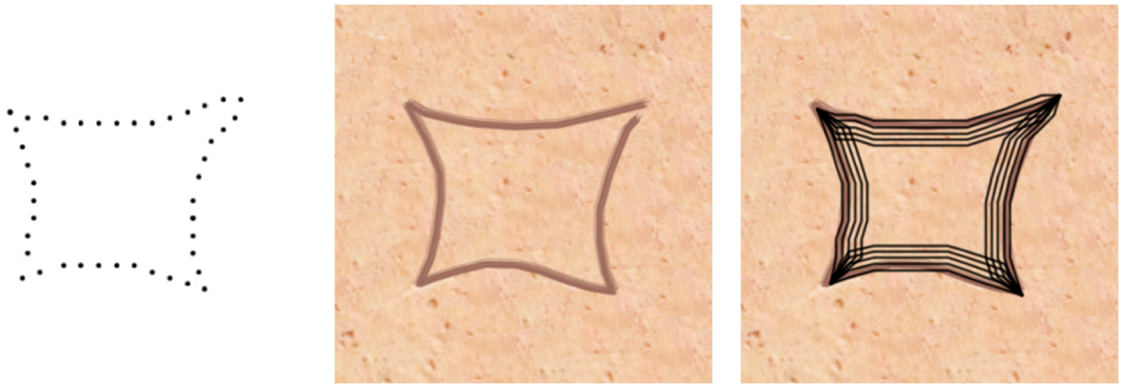
(**b**)

Figure 8. Fish-shape images generated by various network models. (**a**) Fish shape images generated by the multi-attention network. (**b**) Fish shape images generated by the channel attention network.

5. Generation Method of Dynamic Graphics

Another advantage of this method is that dynamic graphics can be created based on the feature graphics output by the channel attention module to achieve the dynamic display effect for text-images. Most previous studies on image generated networks focused on the generation of static images. Some studies have also been conducted on the generation of dynamic text images. For example, Maheshwari et al. [28] applied a GAN in conjunction with a cyclic neural network to learn the time and structure representation of dynamic graphs. Otberdout et al. [29] presented a method to solve the problem of dynamic facial expression generation based on a GAN by applying the manifold-valued Wasserstein GAN to convert generated motion into a landmark sequence and then into an image sequence. Yi et al. [30] proposed a novel progressive fusion network for video SR to generate more realistic and temporally consistent videos by processing image frames using progressive separation and fusion.

In this study, we created dynamic feature graphics based on feature graphics created by the channel attention module through creating graphic instances and calling the FuncAnimation class to generate graphic animation. The specific methods are shown in Appendix A. Considering a box-shaped pottery carving symbol as an example, when users input a descriptive phrase or sentence that contains the keyword "box", the text-image synthesized by the network model may be one of dozens because it includes many box-shaped scribing symbols. Hence, the images generated by the network modal are various and random. In particular, one of the key point feature maps and the feature graphics generated by the channel attention module are shown separately in Figure 9a,b, respectively. The dynamic feature graphic created based on this for the text-image is shown in Figure 9c with an overlay display effect.



(**a**) (**b**) (**c**)

Figure 9. Process of dynamic feature graphic creation for text-image: (**a**) key point; (**b**) feature map; (**c**) dynamic display effect.

It is worth noting, that in the process of dynamic feature graphic creation, the four corners of the box are selected as fixed key feature points, and other key feature points, such as the midpoints of each line, are set using the position change given by the set_ xdata() and set_ ydata() methods in the X and Y directions, respectively, to drive the lines to change. The specific change process is shown in Figure 10.

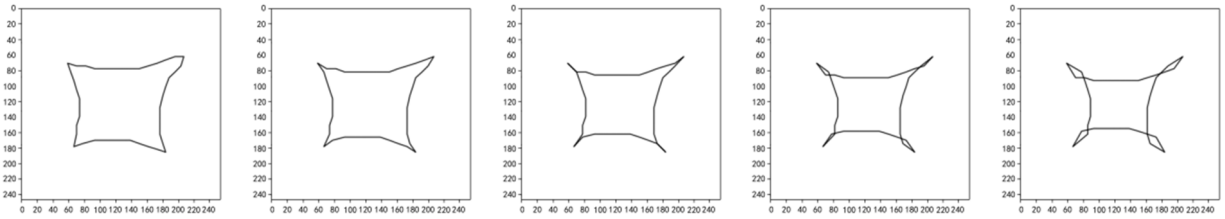


Figure 10. Change process of the dynamic feature graphic for box-shape graphic.

6. Conclusions

In this paper, we propose a novel method of feature graphics based on a channel attention model consisting of a separable deep convolutional neural network and an SENet network. The image feature of sample images is extracted by a convolution operation within the separable deep convolutional neural network and the key point matrix is obtained by a channel weighting calculation to create feature graphics within the SENet network. This method greatly reduced the complexity of image generation and improved the efficiency when we trained the image generation network with the feature graphic maps. We applied this method to the image generation of pottery scribing symbols from the Neolithic Age and achieved a good result. Training the network model with the feature graphic maps output from the channel attention module greatly reduced the complexity of image generation and improved the efficiency. The testing experiment conducted on the MNIST dataset demonstrated the superiority of the efficiency and accuracy of the proposed network model for image generation. Additionally, a comparative experiment for image generation demonstrated that the proposed method was more effective for the image generation of traditional scribing graphics. Furthermore, the feature graphics created by the channel attention module provides a basis for creating dynamic feature graphics, which achieves the dynamic display effect for images generated. For further research, we need to expand the sample image database with more types of pottery scribing symbols from more cultural sites to improve the network model training effect and image generation efficiency.

Author Contributions: Conceptualization, Y.L. and Y.T.; Methodology, Y.L.; Investigation, Y.L. and Y.T.; Validation, Y.L. and Y.T.; Data curation, Y.L.; Writing—original draft preparation, Y.L.; Software, Y.T.; Formal analysis, Y.T.; Project administration, Y.T.; Writing—review and editing, Y.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Natural Science Foundation of China (no. 11902001).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets supporting the conclusion of this article are included within the article.

Acknowledgments: This research is supported by the Support Program for Outstanding Young Talents in Colleges and Universities in Anhui Province (no. gxyq2020166), the Natural Science Research Project of Institutions of Higher Education in Anhui Province of China (no. KJ2017A114), and the Middle-aged Topnotch Talent Support Program of Anhui Polytechnic University.

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

Appendix A

The Creation Method for Dynamic Feature Graphics: The general method for creating dynamic feature graphics based on feature graphics created by the channel attention module is as follows: import the Matplotlib and NumPy packages to build a basic animation environment, load the feature graphics to create graphic instances, and then call the FuncAnimation class to generate graphic animation by moving the key feature points of each line. Considering a double-arc scribing symbol as an example, the specific method for creating dynamic feature graphics is analyzed. The pseudo-code for the algorithm can be described briefly as follows:

# import graphic and animation tools:
import matplotlib.pyplot as plt from matplotlib;
import FuncAnimation as animation class;
# create graphic instance: fig = plt.figure();
axi=fig.add_subplot(x_1, y_1, x_2, y_2, . . . , x_n, y_n);
linei= axi.plot(color, linewidth);
# set keyframe feature points:
point_ani=plt.plot(x_a, y_a, 'keypoint1', x_b, y_b, 'keypoint2', . . .);
# defupdata(num):
point_ani.set_data(x_a, y_a, x_b, y_b, . . .);
return [point_ani];
# generate dynamic graphic:
ani=animation.FuncAnimation(fig=fig,func=updata,frames=np.arange(num,num), interval=num).

In this method, the add_subplot() method is used to load the key feature points to create a graphic instance and the plot() method is used to create lines. The set_xdata() and set_ydata() methods are used to set the position change of the key feature point in the X and Y directions, respectively, and the FuncAnimation() class is used to define the animation using the change method set above. The dynamic change effect of the feature graphic created by this method is shown in Figure A1.

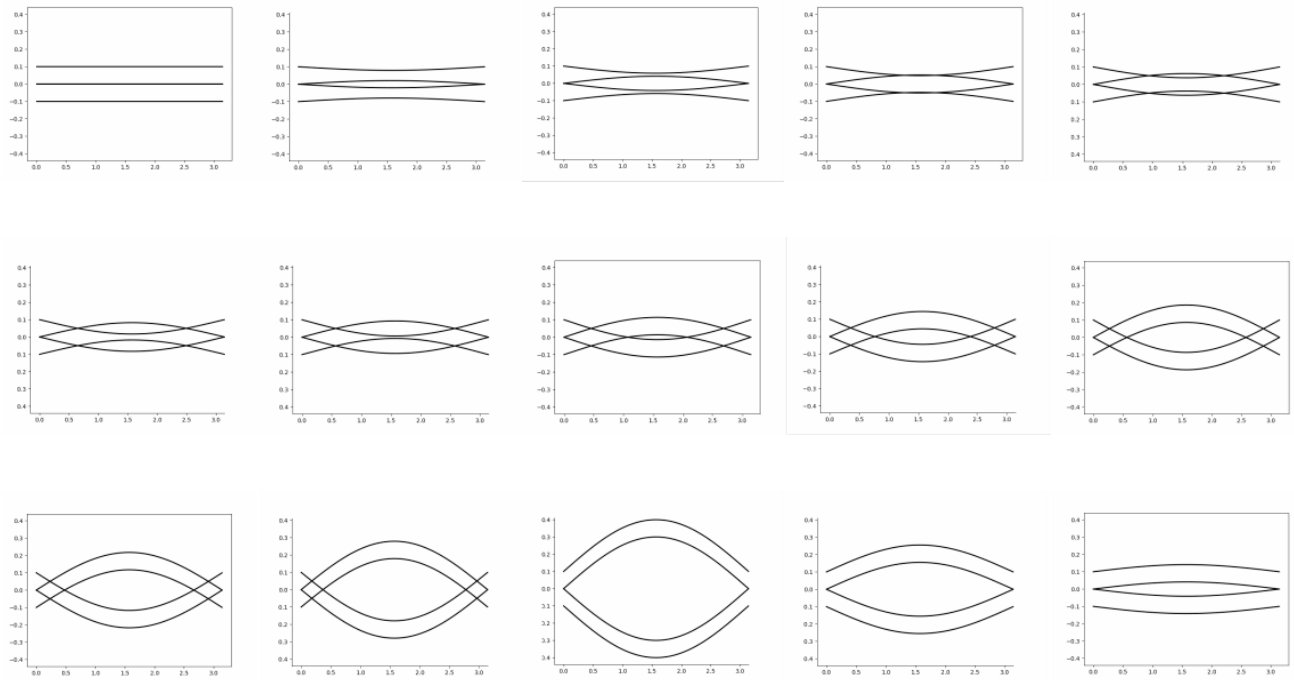


Figure A1. Dynamic change effect of the double-arc feature graphic.

References

1. Ciano, G.; Andreini, P.; Mazzierli, T.; Bianchini, M.; Scarselli, F. A multi-Stage GAN for multi-organ chest x-ray image generation and segmentation. *Mathematics* **2021**, *9*, 2896. [CrossRef]
2. Lee, D.; Lee, J.; Ko, J.; Yoon, J.Y.; Hyun, R.K.; Nam, Y. Deep Learning in MR Image Processing. *Investig. Magn. Reson. Imaging* **2019**, *23*, 81–99. [CrossRef]
3. Marginean, R.; Andreica, A.; Diosan, L.; Balint, Z. Feasibility of automatic seed generation applied to cardiac MRI image analysis. *Mathematics* **2020**, *8*, 1511. [CrossRef]
4. Kim, J.; Jin, K.; Jang, S.; Kang, S.; Kim, Y. Game effect sprite generation with minimal data via conditional GAN. *Expert Syst. Appl.* **2011**, *211*, 118491. [CrossRef]
5. Omri, M.; Abdel-Khalek, S.; Khalil, E.M.; Bouslimi, J.; Joshi, G.P. Modeling of Hyperparameter Tuned Deep Learning Model for Automated Image Captioning. *Mathematics* **2022**, *10*, 288. [CrossRef]
6. Zhang, L.Z.; Yin, H.J.; Hui, B.; Liu, S.J.; Zhang, W. Knowledge-Based Scene Graph Generation with Visual Contextual Dependency. *Mathematics* **2022**, *10*, 2525. [CrossRef]
7. Xue, Y.; Guo, Y.C.; Zhang, H.; Xu, T.; Zhang, S.H.; Huang, X.L. Deep image synthesis from intuitive user input: A review and perspectives. *Comput. Vis. Media* **2021**, *8*, 3–31. [CrossRef]
8. Lee, H.; Kim, G.; Hur, Y.; Lim, H. Visual thinking of neural networks: Interactive text to image generation. *IEEE Access* **2021**, *9*, 64510–64523. [CrossRef]
9. Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; Bharath, A.A. Generative Adversarial Networks an overview. *IEEE Signal Process. Mag.* **2018**, *35*, 53–65. [CrossRef]
10. Pan, Z.Q.; Yu, W.J.; Yi, X.K.; Khan, A.; Yuan, F.; Zheng, Y.H. Recent progress on Generative Adversarial Networks (GANs): A survey. *IEEE Access* **2019**, *7*, 36322–36333. [CrossRef]
11. Frolov, S.; Hinz, T.; Raue, F.; Hees, J.; Dengel, A. Adversarial text-to-image generation: A review. *Neural Netw.* **2021**, *144*, 187–209. [CrossRef] [PubMed]
12. Agnese, J.; Herrera, J.; Tao, H.C.; Zhu, X.Q. A survey and taxonomy of adversarial neural networks for text-to-image generation. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2020**, *10*, e1345. [CrossRef]
13. Li, W.; We, S.P.; Shi, K.B.; Yang, Y.; Huang, T.W. Neural architecture search with a light-weight transformer for text-to- image generation. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 1567–1576. [CrossRef]
14. Quan, F.N.; Lang, B.; Liu, Y.X. ARRPNGAN: Text-to-image GAN with attention regularization and region proposal networks. *Signal Process. Image Commun.* **2022**, *106*, 116728. [CrossRef]
15. Zhang, H.; Zhu, H.Q.; Yang, S.Y.; Li, W.H. DGattGAN: Cooperative up-sampling based dual generator attentional GAN on text-to-image generation. *IEEE Access* **2021**, *9*, 29584–29598. [CrossRef]
16. Chen, H.G.; He, X.H.; Yang, H.; Feng, J.X.; Teng, Q.Z. A two-stage deep generative adversarial quality enhancement network for real-world 3D CT images. *Expert Syst. Appl.* **2022**, *193*, 116440. [CrossRef]
17. Zhang, Z.Q.; Zhang, Y.Y.; Yu, W.X.; Lu, J.W.; Nie, L.; He, G.; Jiang, N.; Fan, Y.B.; Yang, Z. Text to image generation based on multiple discrimination. In Proceedings of the International Conference on Artificial Neural Networks: Artificial Neural Networks and Machine Learning, Munich, Germany, 10–13 September 2019.
18. Tan, Y.X.; Lee, C.H.; Neo, M.; Lim, K.M. Text-to-image generation with self-supervised learning. *Pattern Recognit. Lett.* **2022**, *157*, 119–126. [CrossRef]
19. Tong, W.; Chen, W.T.; Han, W.; Li, X.J.; Wang, L.Z. Channel-attention-based DenseNet network for remote sensing image scene classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 4121–4132. [CrossRef]
20. Li, H.F.; Qiu, K.J.; Chen, L.; Mei, X.M.; Hong, L.; Tao, C. SCAttNet: Semantic segmentation network with spatial and channel attention mechanism for high-resolution remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **2021**, *18*, 905–909. [CrossRef]
21. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E.H. Squeeze-and-excitation networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 2011–2023. [CrossRef]
22. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. In Proceedings of the Twenty-Eighth Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014.
23. Costa, P.; Galdran, A.; Meyer, M.I.; Niemeijer, M.; Abramoff, M.; Mendonca, A.M.; Campilho, A. End-to-end adversarial retinal image generation. *IEEE Trans. Med. Imaging* **2018**, *37*, 781–791. [CrossRef] [PubMed]
24. Feng, F.X.; Niu, T.R.; Li, R.F.; Wang, X.J. Modality disentangled discriminator for text-to-image generation. *IEEE Trans. Multimed.* **2021**, *24*, 2112–2124. [CrossRef]
25. Yang, Y.Y.; Ni, X.; Hao, Y.B.; Liu, C.Y.; Wang, W.S.; Liu, Y.F.; Xie, H.Y. MF-GAN: Multi-conditional fusion Generative Adversarial Network for text-to-image generation. In Proceedings of the 28th International Conference on MultiMedia Modeling, Phu Quoc, Vietnam, 6–10 June 2022.
26. Zhou, R.; Jiang, C.; Xu, Q.Y. A survey on Generative Adversarial Network-based text-to-image generation. *Neurocomputing* **2021**, *451*, 316–336. [CrossRef]
27. Elasri, M.; Elharrouss, O.; Al-Maadeed, S.; Tairi, H. Image Generation: A Review. *Neural Process. Lett.* **2022**, *54*, 4609–4646. [CrossRef]

28. Maheshwari, A.; Goyal, A.; Hanawal, M.K.; Ramakrishnan, G. DynGAN: Generative Adversarial Networks for dynamic network embedding. In Proceedings of the NeurIPS, Vancouver, BC, Canada, 8–14 September 2019.
29. Otberdout, N.; Daoudi, M.; Kacem, A.; Ballihi, L.; Berretti, S. Dynamic facial expression generation on hilbert hypersphere with conditional Wasserstein Generative Adversarial Nets. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 848–863. [CrossRef]
30. Yi, P.; Wang, Z.Y.; Jiang, K.; Jiang, J.J.; Lu, T.; Ma, J.Y. A progressive fusion Generative Adversarial Network for realistic and consistent video super-resolution. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 2264–2280. [CrossRef]