applied sciences

MDPI

Article

Chinese Character Image Completion Using a Generative Latent Variable Model

In-su Jo [1], Dong-bin Choi [1] and Young B. Park [2,*]

[1] Department of Computer, Dankook University, Yongin-si, Gyeonggi-do 16890, Korea; 72200121@dankook.ac.kr (I.-s.J.); 72200118@dankook.ac.kr (D.-b.C.)
[2] Department of Software Science, Dankook University, Yongin-si, Gyeonggi-do 16890, Korea
* Correspondence: ybpark@dku.edu; Tel.: +82-031-8005-3220

Abstract: Chinese characters in ancient books have many corrupted characters, and there are cases in which objects are mixed in the process of extracting the characters into images. To use this incomplete image as accurate data, we use image completion technology, which removes unnecessary objects and restores corrupted images. In this paper, we propose a variational autoencoder with classification (VAE-C) model. This model is characterized by using classification areas and a class activation map (CAM). Through the classification area, the data distribution is disentangled, and then the node to be adjusted is tracked using CAM. Through the latent variable, with which the determined node value is reduced, an image from which unnecessary objects have been removed is created. The VAE-C model can be utilized not only to eliminate unnecessary objects but also to restore corrupted images. By comparing the performance of removing unnecessary objects with mask regions with convolutional neural networks (Mask R-CNN), one of the prevalent object detection technologies, and also comparing the image restoration performance with the partial convolution model (PConv) and the gated convolution model (GConv), which are image inpainting technologies, our model is proven to perform excellently in terms of removing objects and restoring corrupted areas.

Keywords: variational autoencoder; class activation map; object removal; image

Citation: Jo, I.-s.; Choi, D.-b.; Park, Y.B. Chinese Character Image Completion Using a Generative Latent Variable Model. *Appl. Sci.* **2021**, *11*, 624. https://doi.org/10.3390/app11020624

Received: 25 November 2020
Accepted: 7 January 2021
Published: 11 January 2021

1. Introduction

As the technology for handling images has gradually developed, techniques for image completion have emerged [1–4]. The techniques for image completion include object removal technology for erasing unnecessary objects in an image and image restoration technology for restoring corrupted images. The object detection technology is used to remove specific objects present in images [5,6]. The object detection technology classifies objects in an image and detects their location [7–12]. It is possible to delete an object by using a function to find a specific object or to remove unnecessary objects by leaving only the object. This method is suitable for use with simple images where the background of the image is not complex. Concerning the removal of objects in an image, not only the object detection technology but also image inpainting technologies have been developed [2–4]. This technology works by directly masking unnecessary objects in an image using a masking tool and naturally filling the removed area using an inpainting model; this has the advantage that it can be used even in images with complex backgrounds, and it is able to naturally fill corrupted areas and thus can also be used to restore corrupted images.

The Chinese character image data covered in this paper comprise images extracted from ancient books. Ancient books have many corrupted characters due to poor storage conditions, and the gap between letters is not constant, and so unnecessary objects are often included in the process of extracting letters. Because these incomplete Chinese character images are difficult to use as data, image completion technology is required to make them usable images. Chinese characters can be transformed into characters with different meanings if the shape changes even slightly. Therefore, when restoring Chinese

character images, it is important to restore them to the correct shape. Image inpainting technologies are used to restore corrupted images to ensure high-quality images; however, if the corrupted area accounts for a large part of the overall image, it is difficult to restore it to the correct shape, which is not suitable for restoring corrupted Chinese character images. It is also not appropriate to use image inpainting technologies to remove unnecessary objects within Chinese character images. Chinese character images are black and white and very simple images; therefore, image inpainting technologies, which involve separately masking unnecessary objects using a masking tool, is not efficient. The object detection technology used to remove objects easily removes unnecessary objects from simple images such as Chinese character images. However, the unnecessary objects present in Chinese character images are part of other Chinese characters, and so they have very similar characteristics to necessary objects, reducing the ability to detect unnecessary objects. The variational autoencoder with classification (VAE-C) model proposed in this paper can remove unnecessary objects without separately masking them and without performance degradation, even if the necessary and unnecessary objects are similar. Furthermore, this model can accurately restore Chinese characters so that they are not altered when restoring corrupted characters.

In addition, the noise in extracted Chinese character image was removed using fuzzy binarization [13,14]. The fuzzy binarization method minimizes information loss compared to the normal binarization method [1,2].

The VAE-C model is a generative model based on the VAE [15] that generates artificial data using a probability distribution learning method. The VAE has the strength of being advantageous in terms of generating new data in which features of input data are interpolated. Using the fact that images with the interpolated features can be outputted, we changed the VAE so that data with desired features could be outputted. The latent variable used as the input value of the decoder area that generates the data can control the desired features by adjusting the node values. This method is similar to the conditional generative models [16–18], but there is a difference in not using additional latent variables.

However, it is difficult to find which node may correspond to the feature to be controlled, and it is much more difficult for the relevant node to obtain a distribution of the features in a disentangled state. Nodes with an entangled distribution present the problem that they cannot be controlled as desired even if the numerical value is adjusted. This problem was resolved by adding a classification area and utilizing the method of the class activation map (CAM). In terms of VAE-C, we caused the desired feature to be settled as a disentangled distribution in the latent variable by adding a classification area to the VAE model. To find a node with the greatest influence to target features from the nodes that have disentangled distribution, we use the CAM method. By lowering the value of nodes found in this way and outputting the image, it becomes possible to generate an image in which the desired features are offset. With the VAE-C, it is not only possible to remove unnecessary objects in the image, but it can also be used as a function to restore corrupted images.

2. Materials and Methods

2.1. Materials

The VAE-C model takes advantage of the model's structure and learning methods used by the VAE. The difference is that a classification area is added to the latent variable area for supervised learning. CAM, one of the key technologies of VAE-C, is also used in the classification area and is used to track nodes. Figure 1 shows the overall process to help understand the process of data processing.

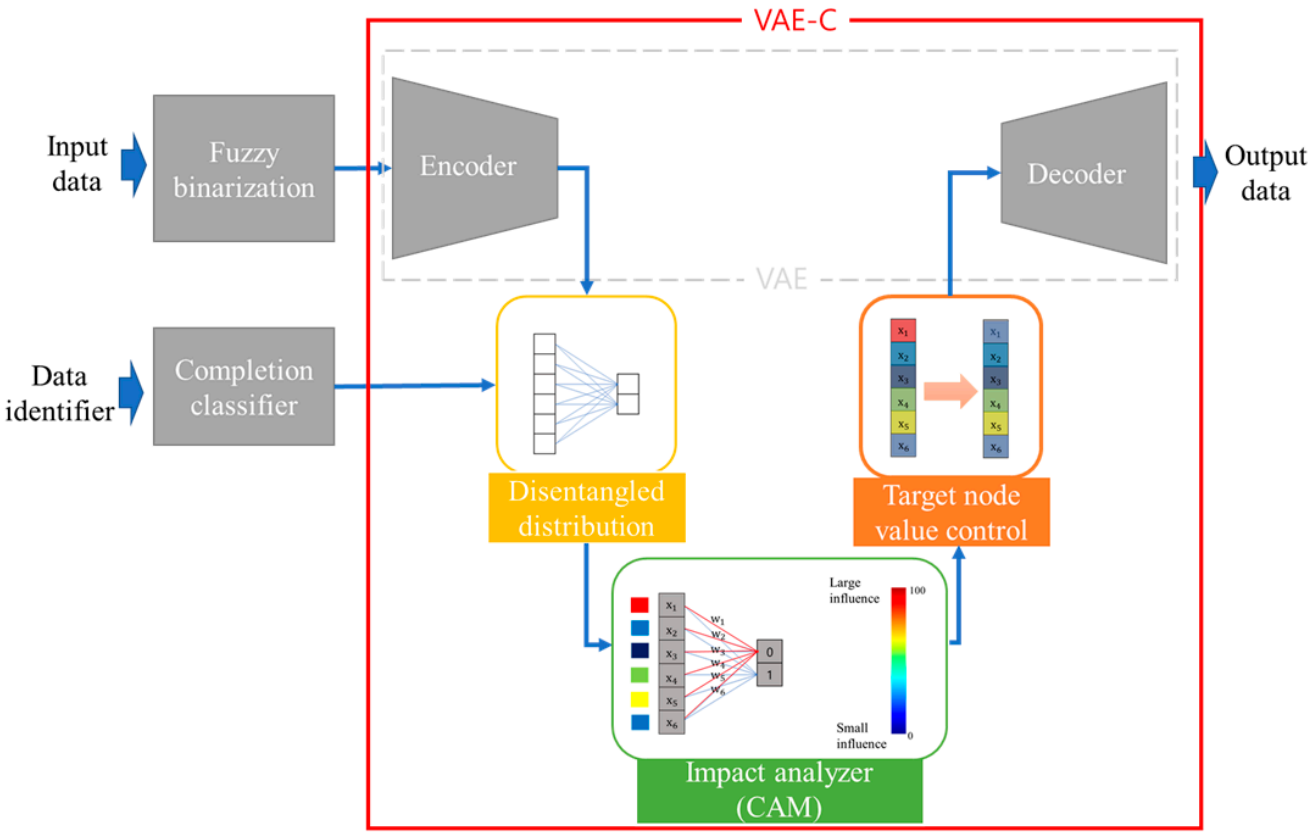

Figure 1. The overview system diagram of the variational autoencoder with classification (VAE-C). VAE-C has two core technologies: disentangled distribution and target node value control.

2.1.1. Variational Autoencoder

The variant autoencoder (VAE), the model underlying VAE-C, is a type of generative model that uses latent variables [15]. As with the autoencoder model, the VAE is composed of an encoder model and a decoder model.

However, there is a difference between the VAE and the autoencoder, Firstly, the encoder model outputs the mean μ and variance σ on the latent variable distribution as a result value. The mean μ and variance σ outputted in this way are used as parameters of the normal distribution equation to form one normal distribution [19]. Randomly sampled values from the formed distribution are used as latent variables. The reason for using this type of structure is to learn $p(x)$—the probability distribution of real data x.

$$p(x) = p(x|z) \tag{1}$$

$$p(x) = E_{z\sim p_\theta(z)}[\, p(x|z)] \tag{2}$$

$$p(x) = E_{z\sim p_\theta(z|x)}[\, p(x|z)] \approx E_{z\sim q_\varphi(z|x)}[\, p(x|z)] \tag{3}$$

Equation (1) presents the equation used to calculate the probability of x by using the latent variable z. It has the role, similar to a decoder, of receiving the latent variable z as an input value and reconstructing it into real data x. Equation (2) adds the fact that the latent variable x has a certain probability distribution $p_\theta(z)$ to Equation (1). In the VAE, the latent variable z is outputted with the form of a probability distribution from the encoder, so Equation (2) is applied. Equation (3) is an equation showing the use of variation inference. The latent variable z is encoded from the real data x to form the probability distribution $p_\theta(z|x)$. The probability distribution constructed in this way is utilized as $q_\varphi(z|x)$ simplified into the normal distribution form. Variation inference is a method of

further simplifying the complex distribution (intractable posterior) in this way and then inducing it into the actual distribution.

$$L_v = E_{q_\varphi(z|x)}[\log\{p(x|z)\}] - D_{KL}(q_\varphi(z|x) \,||\, p(z)) \quad (4)$$

The loss function of VAE takes the same form as Equation (4) using the induction formula [15], playing the role of bringing the normal distribution $q_\varphi(z|x)$ closer to the real data distribution $p_\theta(z|x)$ by Kullback–Leibler divergence (D_{KL}) [20].

The fact that the latent variable is extracted by sampling from the normal distribution works as an obstacle in training the model through backpropagation. To solve this, reparameterization was used. Using $\epsilon \sim N(0,1)$ randomly sampled from the Gaussian distribution, the latent variable was expressed in the form of a function that can be differentiated, as shown in Equation (5).

$$z = \mu + \sigma \times \epsilon \quad (5)$$

Using a method of learning the probability distribution of data, the VAE has the advantages that the distribution of the data has continuity and that it is able to generate the data in a form that is interpolated between data. These advantages laid the foundation for new artificial data to be generated by adjusting the features of the desired data.

2.1.2. Class Activation Map

The class activation map (CAM) has the function of informing us through visualization which features were viewed and judged in the image when the convolutional neural network (CNN) model classified image data [21,22]. Although many studies have already been conducted on how the feature maps of CNN are expressed [23–25], CAM has a difference in that it visually shows which features are viewed as important when classifying an image. The result of CAM is expressed with position information in the actual image. When classifying the data, the areas with high influence are expressed in red, while the areas with low influence are expressed in blue. Here, the extent of influence refers to the value of the relevant nodes.

The reason why the position information can be expressed in this way is that the model is in the form of full convolution networks. The CNN has taken the form of a fully connected layer as the last layer to classify data. Even though topological information is preserved by convolution, since the feature map would be flattened when passing through the fully connected layer, the information regarding the position is lost. Therefore, in order to represent accurate positions, it is essential for the solutions that all layers can be designed in the form of convolution. Global average pooling (GAP) was used as a solution. GAP is one of the pooling layers of CNN and uses a method of extracting one mean value and targeting all nodes in the feature map [26]. Since the form of the extracted nodes is identical to the flattened form, it is possible to classify them with the same process as in the fully connected layer.

2.2. Methodology

2.2.1. Disentanglement

Generative models such as VAE can output image results whose specific features are altered by adjusting the node values present in the latent variable [15,17,18]. The characteristic of these generative models is that they learn the probability distribution of the data. Learning the probability distribution of data helps to create a more natural output when reconstructing data. Each node of the latent variable has each feature of the data in the form of a probability distribution, and the distribution of these nodes is mixed to produce the results. However, the distribution of the latent variable, including the features of data, is entangled. If data features are organized in an entanglement, the desired feature cannot be accurately controlled, even if the node value is adjusted.

There are two things to be considered in order to adjust the desired features in an image by using VAE. First, the distribution of the latent variable must be disentangled so that the node value can be controlled. This is necessary to accurately control the desired features. This method is similar to the one used in the conditional generative models [17,18] and information maximizing generative adversarial nets (InfoGAN) [27].

Second, it must be determined which node contains the feature that is desired to be controlled in the mean of the latent variable distribution. The numbers of nodes belonging to the mean of the latent variable distribution can be designed by the user, but too few nodes can make it difficult to reconstruct the data. Accordingly, a measure is necessary which allows us to clearly determine which node must be controlled of the many nodes existing in the mean of the latent variable distribution.

The VAE-C model suggested in this paper is a model focusing on resolving the two considerations as mentioned above. In order to make the latent variable distribution disentangled, a classification area is added to the mean of the latent variable distribution. The classification model can classify the inputted data by supervised learning. When classifying the data, the model requires features that serve as the basis for classification [21] and these features are created through the supervised learning where models classify data. Therefore, in order for the latent variable used to reconstruct the data in the VAE to obtain the disentangled distribution, a classification area should be added. For a classification area, labeled data which are created from the completion classifier, are needed to carry out supervised learning. Figure 2 shows that the distribution of the latent variable means is disentangled by adding classification areas and conducting supervised learning.

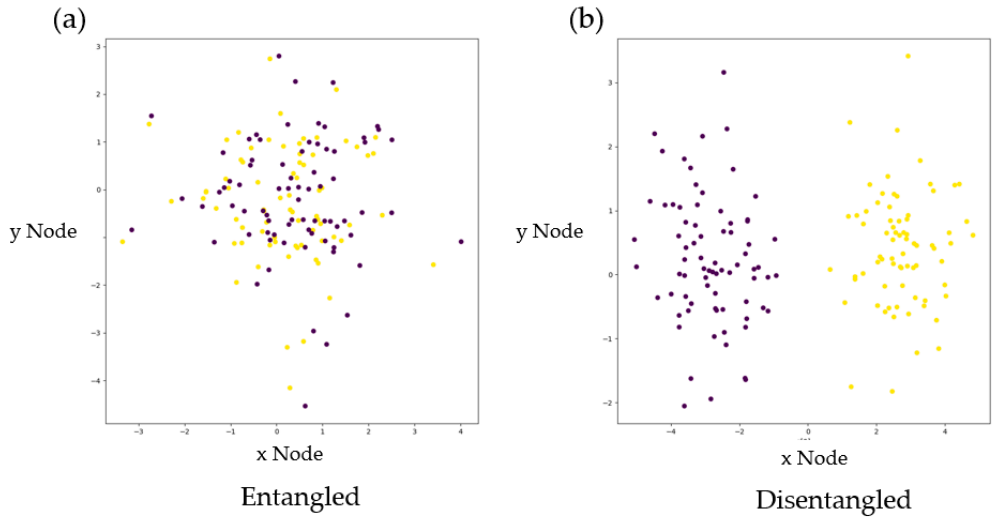

Figure 2. Graph of the latent variable of data with two labels. The x-axes and y-axes on the graph represent the different nodes of the latent variables, and the corresponding values represent node values. (**a**) Entangled distribution of data features, (**b**) disentangled distribution of data features. The distribution of data across x-nodes is disentangled; an entangled distribution is disentangled by adding the classification area.

2.2.2. Tracking Nodes Using Class Activation Map

If the desired features in the latent variables are made into a disentangled distribution, it must be determined which node has the information of these features. To find the corresponding node, the class activation map (CAM) technique [21,22] is utilized. By applying this technology, it is possible to find which node has the most influence when the model performs classification.

Figure 3 shows the process of seeking the node with the greatest influence in the latent variable by using the CAM method, in the process of removing unnecessary objects in the image using the VAE-C model.

$$S_c = \sum_k W_k^c \sum_{x,y} f_k(x,y) = \sum_{x,y} \sum_k W_k^c f_k(x,y) \quad (6)$$

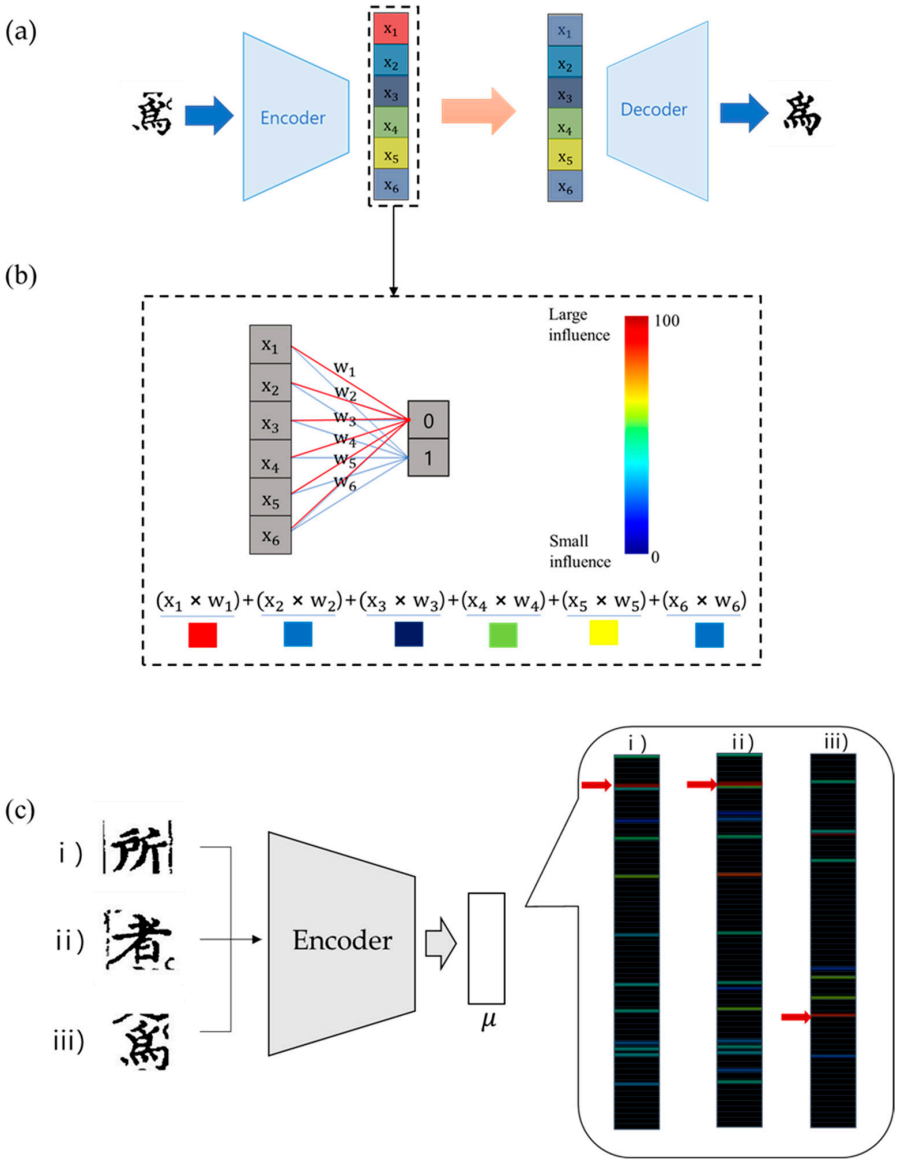

Figure 3. The process by which the variational autoencoder with classification (VAE-C) model removes unnecessary objects in the image: (**a**) The process of changing the node value in the latent variable to eliminate unnecessary objects. (**b**) The process of finding the node with the greatest influence among the nodes in the latent variable by using the class activation map (CAM) method. (**c**) The actual CAM image outputted using Chinese character images. The black area represents an area with value of less than 20%. The red arrow points to the node with the highest value.

Equation (6) represents how to calculate the score S_c, which becomes a measure for classifing data when CAM is applied to the model. It should be noted that the value of multiplying all the nodes $f_k(x,y)$ in the feature map by a single weight W_k^c and adding them all is identical to the value of multiplied by weight W_k^c after adding all the nodes $f_k(x,y)$ in the feature map. Eventually, the feature map is treated in the same manner as a single node and used to calculate the score S_c. Using this property, it is possible to find the

most influential node in the form of a dense layer, just like finding the most influential area in the form of a feature map.

VAE-C uses the CAM method to seek the node corresponding to the desired feature and then is able to control the image result to be outputted by modifying the value of the relevant node. This model can control the degree of removal with a value rather than simply removing the feature.

2.3. Model Construction

The model structure of VAE-C is very similar to that of VAE. In the VAE, if the classification area is added to the mean μ of the latent variable distribution, it becomes the VAE-C model. It is more efficient to connect the classification area to the mean μ corresponding to the center of the latent variable than to directly connect to the latent variable z. Using values randomly sampled from the normal distribution of the latent variable z acts as an obstructive factor to supervised learning. The encoder and decoder models are organized using the convolution technique so that they can learn the topological information of the data. In the VAE-C model, the convolution technique [28–31] plays a very significant role in separating features independently. Table 1 shows in detail the values of layers in the model structure.

Table 1. Layer values for the VAE-C model. Conv: convoluted; Deconv: deconvoluted.

No	Type	Kernel	Stride	Outputs
1	Conv	3 × 3	1 × 1	16
2	Conv	3 × 3	2 × 2	32
3	Conv	3 × 3	2 × 2	64
4	Conv	3 × 3	2 × 2	128
5	Conv	3 × 3	2 × 2	256
6	Dense			128
		Reparameterization		
7	Dense			128
8	Deconv	3 × 3	2 × 2	256
9	Deconv	3 × 3	2 × 2	128
10	Deconv	3 × 3	2 × 2	64
11	Deconv	3 × 3	2 × 2	32
12	Deconv	3 × 3	1 × 1	1

The fully connected layer model [32–34] has a structure in which one node has barely any independent features because one node affects all the other nodes in the next layers. The overall structure of the VAE-C model is shown in Figure 4.

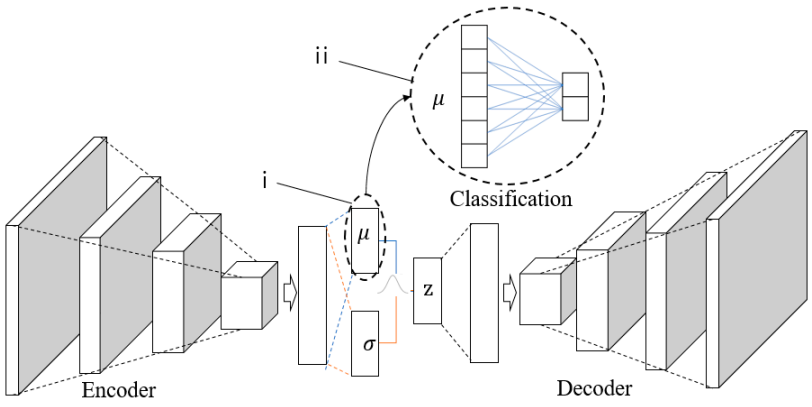

Figure 4. Model structure of the VAE-C, representing a structure in which classification is added to the layer corresponding to the mean μ in the VAE model.

$$S_{cl} = \text{Sofmax}\left(\sum_{k} W_k^c x_k\right). \tag{7}$$

$$L_c = -\sum_{c} t_c \log(S_{cl}) - (1 - t_c)\log(1 - S_{cl}) \tag{8}$$

The learning of VAE-C can be done by adding the learning of the classification area to the learning method used in the VAE [15]. The cost function of classification uses binary cross entropy loss, as shown in Equation (8). Passing through softmax, the weight W_k^c and the mean nodes x_k of the calculated latent variable distribution are randomized to utilize cross entropy. Equation (7) represents the process of calculating the probability value S_{cl}. VAE learning with Equation (4) and classification learning with Equation (8) are performed alternately, and the encoder model learns to extract the feature maps, satisfying both purposes.

2.4. Chinese Character Images Dataset

The data covered in the paper comprise Chinese character image data. These Chinese character images are data obtained from ancient books, which are extracted in the form of a bounder box through object detection techniques [8,9]. Chinese character image data have two characteristics: the first characteristic is that there are many corrupted Chinese characters because ancient books are not kept well. To take advantage of the corrupted Chinese character images, the technology used to restore images is essential. There are some precautions which must be taken when restoring corrupted Chinese character images. Chinese characters become other Chinese characters with different meanings due to small differences in shape; thus, it is important to restore the exact shape, not just converting the image to a high-definition image. Figure 5 shows examples of Chinese characters with similar shapes but different meanings.

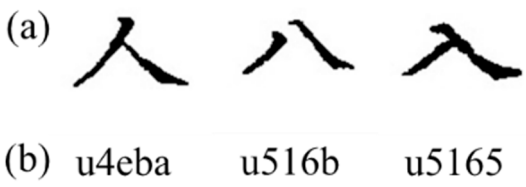

Figure 5. Examples of Chinese characters whose shapes are similar but not identical. (**a**) Chinese character images; (**b**) represents the Unicode of the image.

The second characteristic is that there are cases in which some of the other Chinese characters are mixed into the Chinese character images extracted in the form of a bounding box (bbox) because the spacing between the characters in the ancient books is not constant. The technology used to remove unnecessary objects is essential to remove parts of other Chinese characters that are unnecessary within each Chinese character image. There is also a method of extracting letters into the mask area, but the method of extracting letters into the bbox area was used to rule out the possibility of incorrect extraction. Figure 6 shows an example of the Chinese character images extracted from ancient books.

Before using Chinese character images as input data, a pretreatment process is undertaken to eliminate noise. Eliminating noise makes it possible for the model to learn about the objects in the image more accurately. The fuzzy binarization method was used to eliminate noise; as the fuzzy binarization method dynamically selects thresholds while considering different types of objects, there is less information loss than in the normal binarization method when noise is eliminated. Figure 7 shows a comparison of images before applying the fuzzy binarization method with images after applying it.

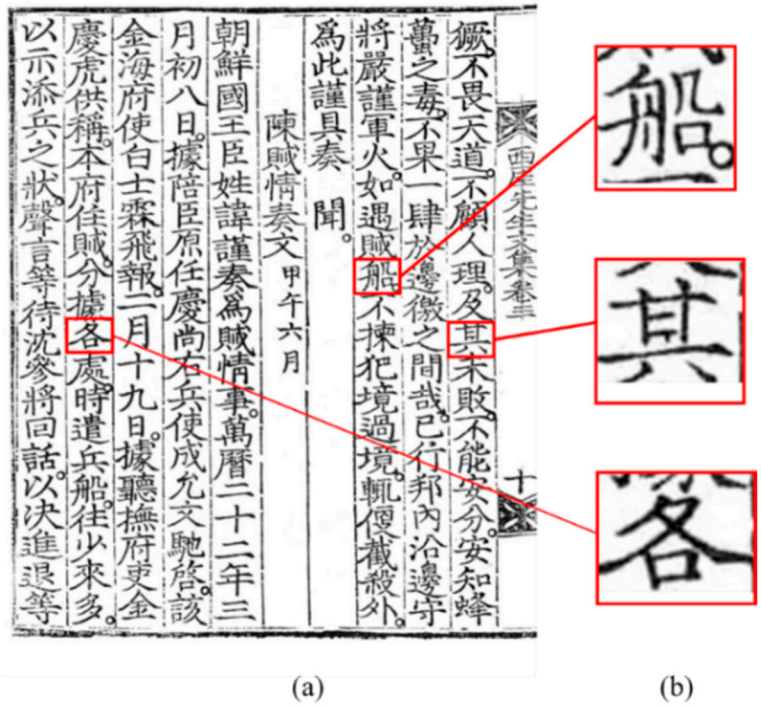

Figure 6. The appearance of Chinese character images extracted from ancient books. (**a**) The actual ancient book image. (**b**) Chinese character images from ancient books, extracted in the form of a bounding box (bbox) using object detection technology.

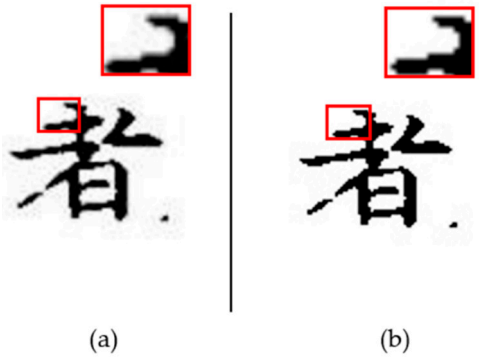

Figure 7. An image that eliminates noise using the fuzzy binarization method. (**a**) An image without the fuzzy binarization method applied; (**b**) image with the fuzzy binarization method applied.

3. Results

3.1. Variation of Output Value as Node Value Changes

The output image of the VAE-C model is generated based on the learned data. Therefore, although they are the same letter images, when there are multiple types of data, the method is advantageous for generating good results. The data used in the experiment were Chinese letters used in 'The Building and Application of Database of Various Traditional Chinese Character Shapes Dictionary in Korea' project. Figure 8 shows the results of removing unnecessary objects and restoring corrupted images by using the VAE-C model. By reducing node values with a high influence from unnecessary objects or corrupted areas within the latent variable, we can see the outputted result of a clean image. This means that the relevant feature was well induced in a disentanglement, and that the node with that feature was well detected.

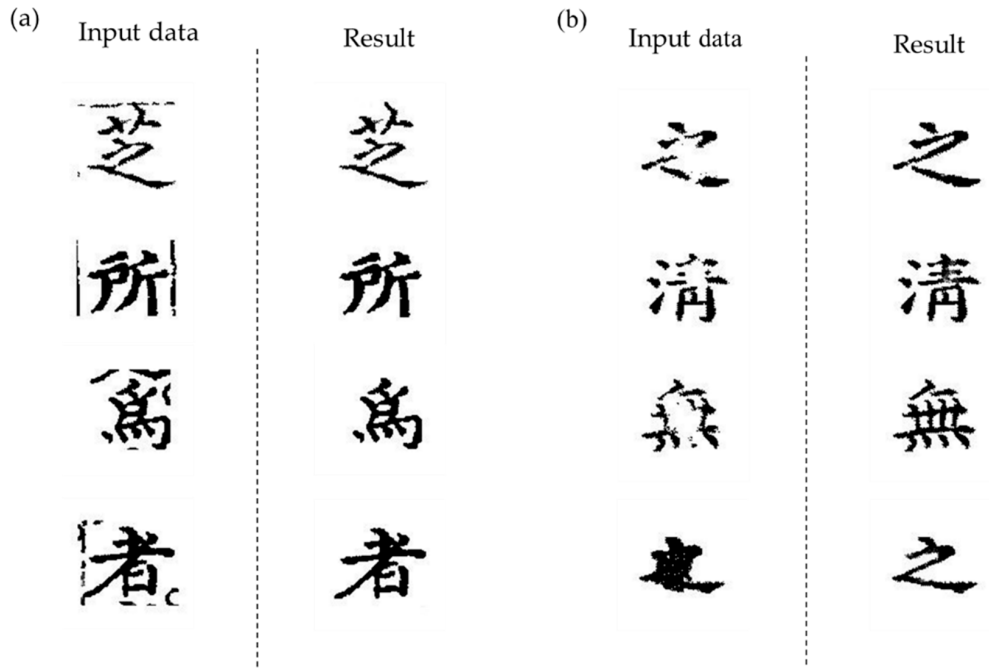

Figure 8. The results outputted by using the VAE-C model: (**a**) Treating the Chinese letter image with an unnecessary object as an input value and outputting the result of removing noise, (**b**) treating the corrupted Chinese letter image as an input value and outputting the restored result.

In terms of removing unnecessary objects from simple images such as Chinese character images, there are many examples using object detection technology [5,6]. However, when it is difficult to distinguish a necessary object from an unnecessary object, such as the case of Chinese letter data, the performance of noise removal using object detection technology begins to decrease. An image inpainting technique has also been used to remove unnecessary objects; the technology, which aims to naturally fill in the removed area, involves the process of masking unnecessary objects directly using a tool. Because Chinese characters are simple images in black and white, objects are already removed naturally during masking. Therefore, it is inefficient to use image inpainting techniques to remove objects within Chinese character images.

When noise was removed or a corrupted image was restored using the VAE-C model, we evaluated the images with the similarity comparison scale in order to investigate how similar they were to the actual images. We used the peak signal-to-noise ratio (PSNR), mean square error (MSE) [35–37], and structural similarity index measure (SSIM) [38] as the similarity comparison scale. Figure 9 indicates the degree to which unnecessary objects are removed when reducing node values, and Figure 10 indicates the degree to which images are restored.

The peak signal-to-noise ratio (PSNR) represents the power of noise to the maximum power that a signal can have. PSNR is a measure representing how little noise is in the generated image compared to the original image, which has been used often as a tool to measure whether two images are similar. The value increases as the noise of the image outputted by the model is reduced compared to the actual image. Considering the PSNR comparison graph shown in Figure 9, we can see that the value increases when the unnecessary object is removed. The difference in PSNR between when the node value was not decreased and when the node value was decreased by 12 is about 1 db. Figure 10 shows that the lower the node value, the more the image is restored, whereas the PSNR comparison graph shows little quantitative change when the node value is reduced. The reason for this is that, due to the model characteristics, frequent noise occurs in the process of restoring the image after connoting it as a latent variable.

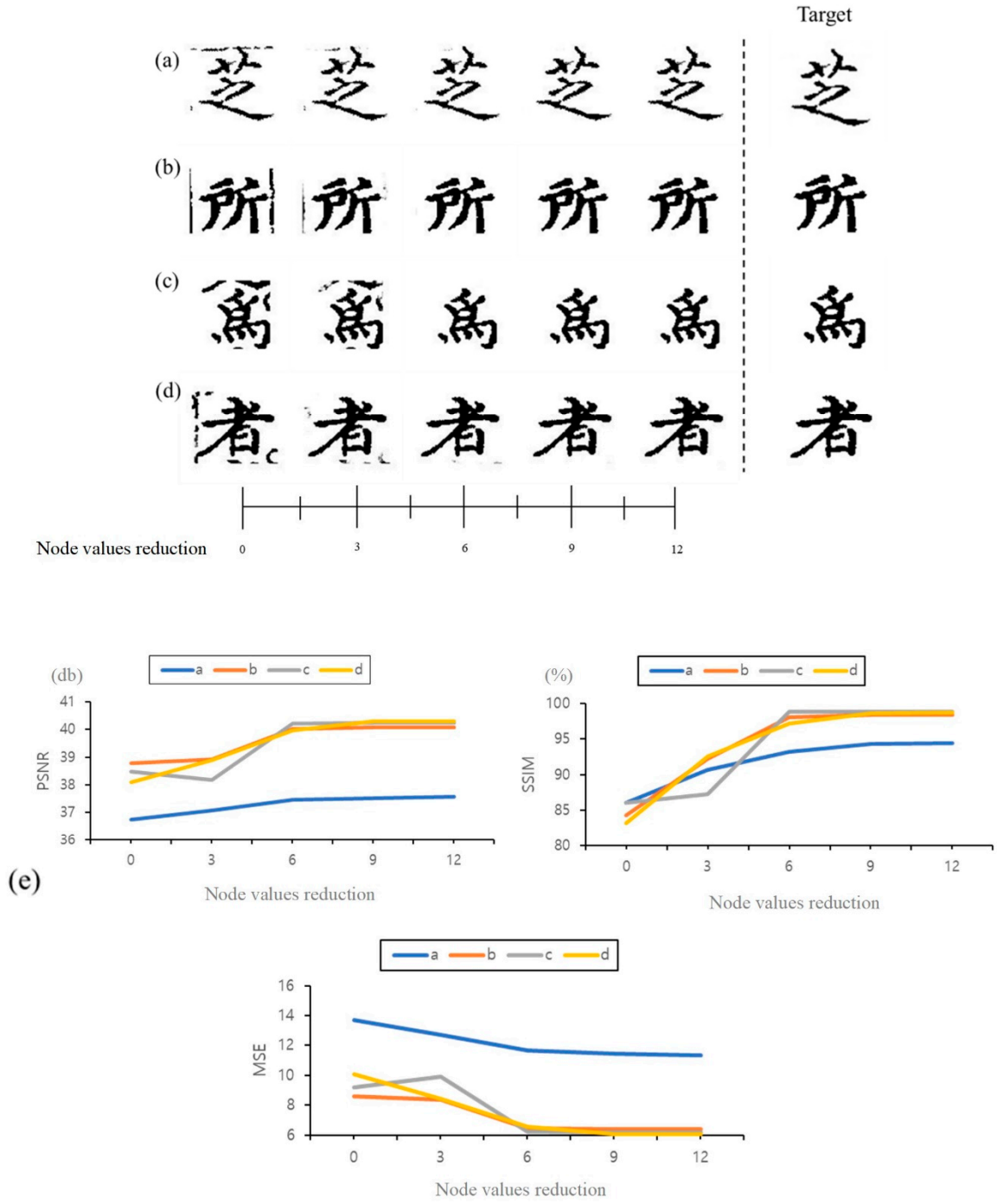

Figure 9. Changes in the output value according to the decrease in the node value with the greatest influence among the latent variables. (**a**–**d**) The qualitative result of removing unnecessary objects due to reduced node values. (**e**) The quantitative result for the scale of decreasing node values. MSE: mean square error; PSNR: peak signal-to-noise ratio; SSIM: structural similarity index measure.

The structural similarity index measure (SSIM) [38] is a tool used to measure the similarity of an original image to the distortion caused by compression and conversion. SSIM relies on the principle that structural information of images is derived when actually comparing two images. The higher the number, the more similar two images are. When examining the SSIM comparison graph in Figure 9, we can see that if an unnecessary object is removed, the value increases, similar to PSNR. The difference in SSIM when the node value decreased by 12 is about 14% compared to when the node value was not decreased. Figure 10 also shows that the SSIM value for corrupted image restoration increases. Unlike

PSNR, since SSIM does not judge similarity via image noise but uses image structure information, even in the event of image restoration, judging similarity is much easier. The difference in SSIM is about 6% when the node value was decreased by 12 compared to when the node value was not decreased.

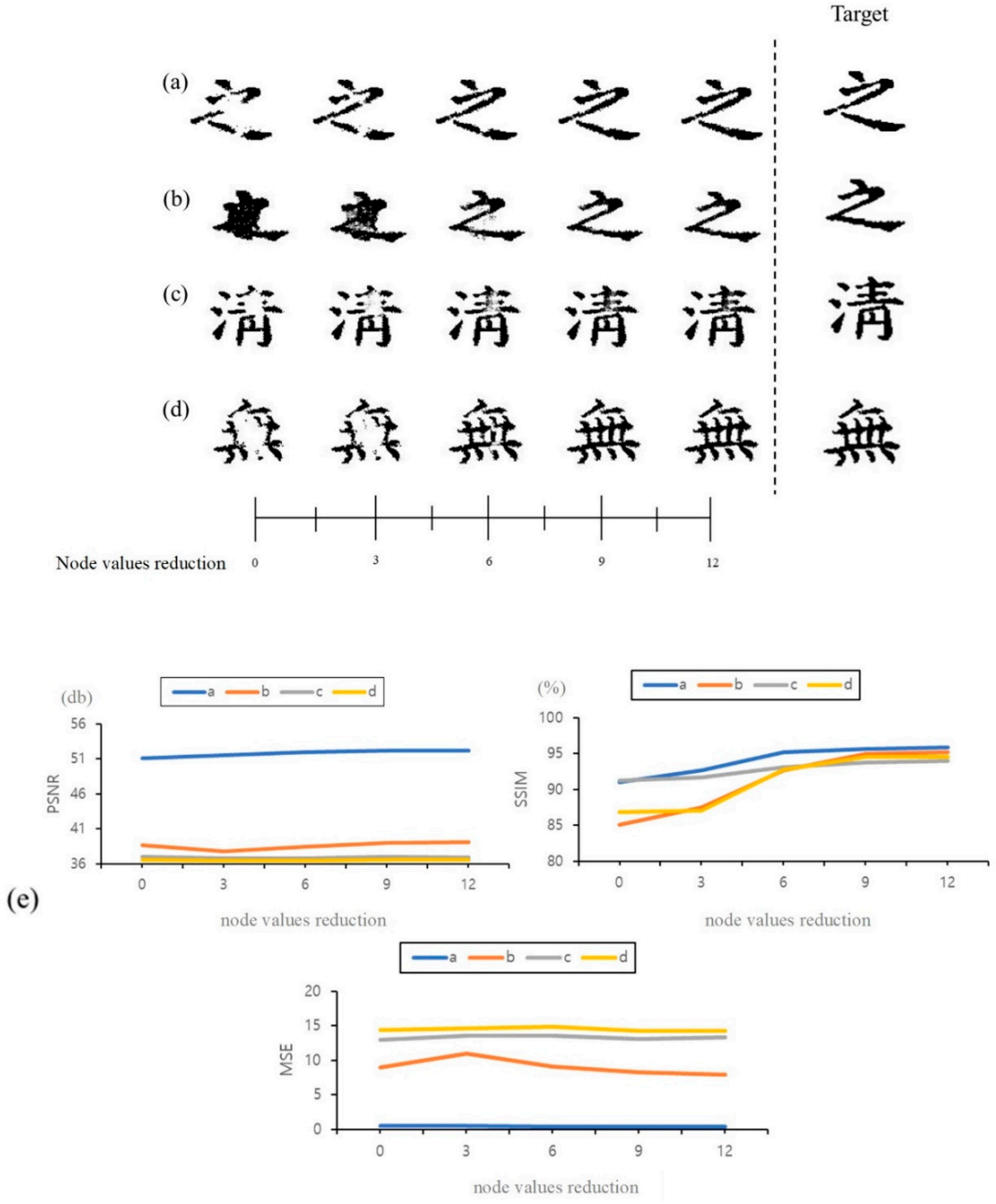

Figure 10. Changes in the output value according to the decrease in the node value with the greatest influence among the latent variables. (**a**–**d**) The qualitative results of image restoration due to reduced node values. (**e**) The quantitative result for the scale of the node value reduction.

The mean square error (MSE) refers to the difference between the pixel values of two images. The similarity is judged by investigating how much average difference occurs between the expected value and actual result. The smaller the MSE value, the higher the similarity between the two images. Considering the MSE comparison graph shown in Figure 9, we can see that when the unnecessary object was removed, the value decreased.

The difference in the MSE between when the node value was not decreased and when the node value was decreased by 12 is about 2.6. Figure 10 shows that for an image restoration that is corrupted, there is little change in the graph, as is the case for the PSNR result. This phenomenon is caused by frequent noise, as mentioned earlier.

This experimental result highlights the advantages of the VAE-C model. It is possible to adjust node values to control how much unnecessary objects are removed and how much corrupted images are restored. It is a function that does not exist in image inpainting technologies and object detection technology.

3.2. Image Restoration Performance Comparison

We compare the VAE-C model with image inpainting technologies. Image inpainting technologies are mainly used in a model for restoring corrupted images [2–4]. We randomly create a corrupted image using the mask algorithm provided by the partial convolution model (PConv) [3] and restored the corrupted image with each model. All of these experiments were conducted in the same environment, and the learning time was 3 s per 10 epoch based on 1 image for Graphics Processing Unit (GPU); the results of all models were the same. When outputting the result images, the VAE-C models may take more time than the other two models because they have a process to control the target node value.

Figure 11 shows the qualitative results of the restoration of the corrupted image that arbitrarily damaged the Chinese character images using each model. The image inpainting (PConv) model [3] and the gated convolution (GConv) model [2] show that some areas have not been restored or are over-injected compared to the VAE-C model, which is different from the original image. This result shows that it is not appropriate to use the image inpainting technologies PConv model and the GConv model for the restoration of corrupted Chinese character images. At this time, the GConv model was used without setting the user-guided option.

Figure 11. The results of restoring a corrupted image (**a**) that has corrupted the target image (**e**) using the partial convolution model (PConv) (**b**) [3], gated convolution model (GConv) (**c**) [4], and VAE-C model (**d**).

To further compare the corrupted image restoration performance of the two models—image inpainting and VAE-C—the same experiment was conducted with the Places2 dataset [39], celebrity faces attributes (CelebA) [40] dataset, and Canadian Institute for

Advanced Research (Cifar-10) dataset. The Places2 and CelebA datasets are frequently used to compare image restoration performance. The Places2 dataset, a repository of 10 million scene photographs labeled with scene semantic categories, comprises a large and diverse list of the types of environments encountered in the world [39]. CelebA is a large-scale face attributes dataset with more than 200,000 celebrity images, each with 40 attribute annotations [40]. Cifar-10 is a multi-class dataset consisting of 60,000 32 × 32 color images in 10 classes [41]. A Cifar-10 dataset with a relatively low-resolution image will serve to show what results will be obtained when the model is tested with a low-resolution image. Figure 12 shows the qualitative results of this experiment. PSNR, SSIM, and MSE scales were used to determine whether the VAE-C model was the best model to restore a corrupted image. Figures compared to each model can be found in Table 2.

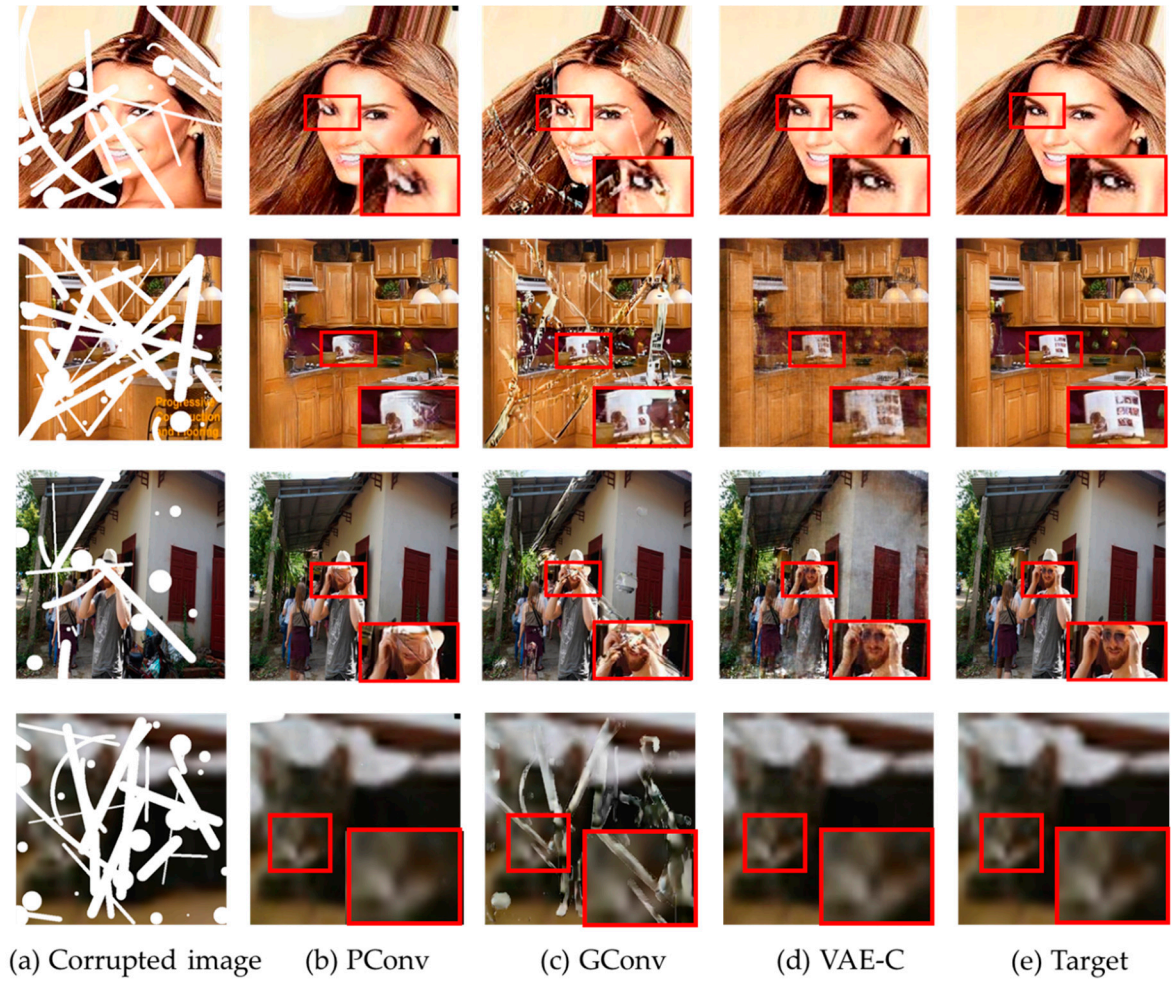

Figure 12. Using the Places2 dataset [39], CelebA (celebrity faces attributes) face dataset [40], and Cifar-10 (Canadian Institute For Advanced Research-10) dataset [41], the results of restoring corrupted images (**a**) with the inpainting model (**b**,**c**) and the VAE-C model (**d**) are shown. The bottom Cifar-10 dataset represents the result of using a low-resolution image, (**e**) is ground-truth. The red area shows some areas enlarged to see the restored results in more detail.

The VAE-C model gives better results for all datasets compared to the PConv and GConv models. On average, the VAE-C models showed better values with an MSE of 3.3, PSNR of 2.1, and SSIM of 5.8 compared to the PConv model, and an MSE of 7.6, PSNR of 3.4, and SSIM of 9.4 compared to the GConv model. Image inpainting technologies and the VAE-C model have the characteristic in common that corrupted images can be restored. However, the restored image which has the most similar form to the original image is that produced by the VAE-C model.

Table 2. A table comparing the degree of restoration of corrupted areas when restored using PConv, GConv, and VAE-C models with Places2, CelebA (celebrity faces attributes) face, Cifar-10, and Chinese character images dataset. MSE: mean square error; PSNR: peak signal-to-noise ratio; SSIM: structural similarity index measure.

Dataset	Methods	MSE ↓	PSNR ↑	SSIM [38] ↑
Places2 [39]	Pconv	9.43	38.38 db	94.85%
	Gconv	13.06	36.97 db	91.70%
	VAE-C	8.23	38.97 db	98.21%
CelebA [40]	Pconv	17.99	35.81 db	90.20%
	Gconv	21.6	34.79 db	86.46%
	VAE-C	11.21	37.63 db	98.84%
Cifar-10	Pconv	3.87	42.24 db	99.59%
	Gconv	29.89	33.37 db	83.58%
	VAE-C	2.48	44.17 db	99.81%
Chinese character	Pconv	6.71	39.87 db	93.75%
	Gconv	10.17	38.06 db	90.07%
	VAE-C	2.6	43.97 db	99.32%

3.3. Object Removal Performance Comparison

Chinese character datasets were used to test the efficacy of object removal. The Places2, CelebA, and Cifar-10 datasets, which were tested earlier, are complex images, unlike Chinese character data, which are black and white images. The image inpainting method is used to remove complex image objects, and this technology uses the method of forcibly damaging and restoring the area of the object that is intended to be removed within the image. This paper conducted an experiment to remove objects without the process of forcibly damaging the area of the object within the image with simple images such as Chinese character images.

In simple images, such as Chinese character images, object detection techniques are sometimes used to remove unnecessary objects that exist within the images [5,6]. However, there is a problem in that the performance of object detection technology is degraded because the unnecessary objects present within the Chinese character images have similar characteristics to the desired object. Figure 13 compares the qualitative results of removing unnecessary objects in the image using VAE-C models and the object detection technology mask regions with convolutional neural networks (Mask R-CNN) [10]. The result of removing unnecessary objects using Mask R-CNN shows that all unnecessary objects were not removed or that the required objects were corrupted. In contrast, the VAE-C model neatly removed only unnecessary objects.

Figure 13. Comparison of the results of eliminating unnecessary objects by using the VAE-C model and Mask R-CNN (mask regions with convolutional neural networks).

Table 3 shows the degree to which Mask R-CNN and VAE-C models removed unnecessary objects using a similarity comparison scale. The VAE-C model performed well with a PSNR of 5.2, SSIM of 9.7, and MSE of 8.8 compared to Mask R-CNN. This result shows numerically that the VAE-C model is more efficient than the Mask R-CNN for eliminating unnecessary objects.

Table 3. Object removal performance of two models—VAE-C and Mask R-CNN model—compared using Chinese character images.

	MSE ↓	PSNR ↑	SSIM ↑
VAE-C	5.3	41.7 db	92.2%
Mask R-CNN	14.1	36.5 db	82.5%

4. Discussion

We proposed a VAE-C model that effectively removes objects and restores images more accurately than existing image completion models. However, the images outputted by the model have a lower quality than the actual images. Figure 14 qualitatively shows that the images outputted by the VAE-C model are of inferior quality compared to the actual images. This problem occurs because noise is generated in the process of implicating the image as a latent layer and then restoring it again. To address this, research should be done to add the skip connection technique, which increases image quality, to the VAE-C model.

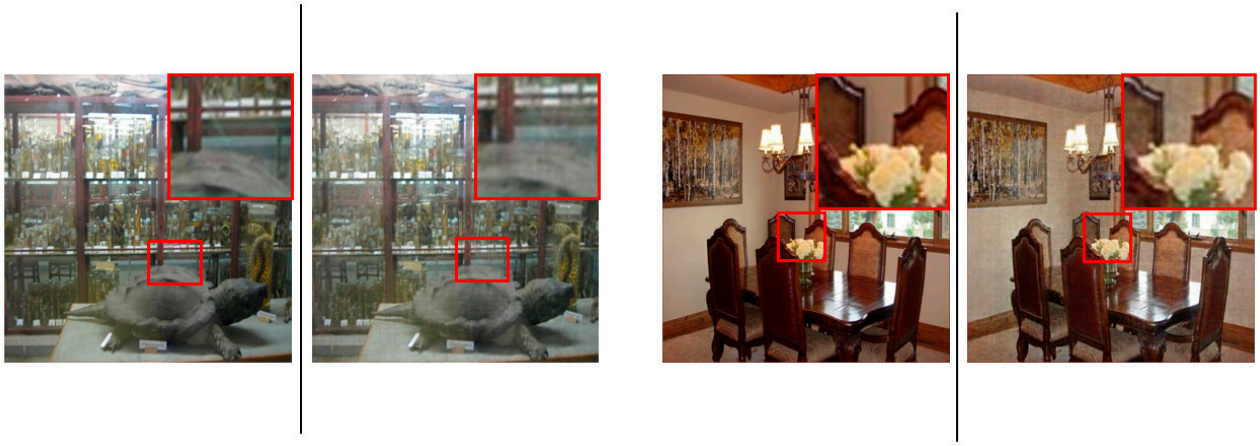

Figure 14. Comparison of actual images (**left**) and outputted images (**right**) using the VAE-C model. The outputted image using the VAE-C model has a lower quality than the actual image.

We also expanded our framework to change the background environment of the image by disentangling the distribution of features responsible for the background environment. The VAE-C model can control the features if the distribution of features is disentangled. The method of control is completely consistent with the method mentioned above. The distribution of features corresponding to the background environment are simply disentangled. Figure 15 shows the result of changing the background of the image from night to day and from winter to spring.

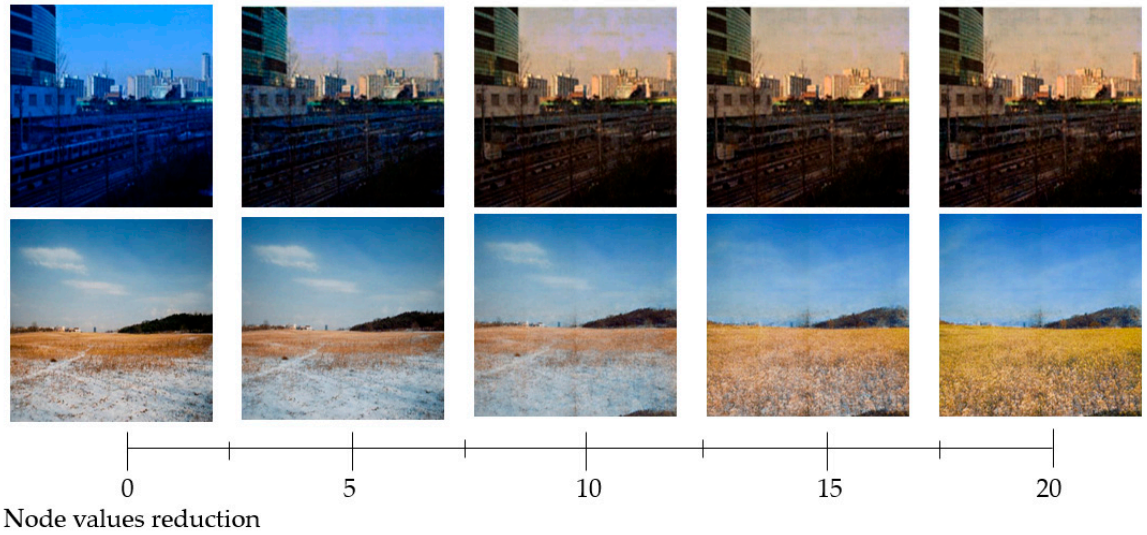

Figure 15. Result of changing the background environment of images using the VAE-C model. It can be observed that the lower the node value, the greater the change in the background environment.

5. Conclusions

In this paper, a VAE-C model for image completion is proposed to turn Chinese character images, which are incomplete data, into clean images so that they can be utilized as data. To determine the image completion performance of the VAE-C model, a comparative experiment was conducted using Mask R-CNN object detection technology and PConv and GConv image inpainting technologies. The VAE-C model showed a PSNR of 5.2, SSIM of 9.7, and MSE of 8.8 compared to Mask R-CNN. On average, the VAE-C models also showed better values—with an MSE of 3.3, PSNR of 2.1, and SSIM of 5.8—than the PConv model and the GConv model. The latter had an MSE of 7.6, PSNR of 3.4, and SSIM of 9.4. The experimental results showed that the VAE-C model had better image completion performance compared to other models. In addition to image completion functions such as object removal and image restoration, the VAE-C model can be used for more diverse purposes, such as changing the background environment of an image. In the future, it will be necessary to study the design of the VAE-C model with added skip connections to produce noise-free, high-quality results with the VAE-C model.

Author Contributions: Conceptualization: I.-s.J. and Y.B.P.; project administration: Y.B.P.; supervision: Y.B.P.; validation: D.-b.C.; writing—original draft: I.-s.J.; writing—review and editing: D.-b.C. All authors have read and agreed to the published version of the manuscript.

Funding: No funding has been perceived for this study.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available CelebA, Places2 and Chinese character datasets were analyzed in this study. This data can be found here: [http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html], [http://places2.csail.mit.edu/download.html] and [https://db.itkc.or.kr/].

Acknowledgments: The present research was supported by the research fund of Dankook University in 2020.

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

References

1. Iizuka, S.; Simo-Serra, E.; Ishikawa, H. Globally and locally consistent image completion. *ACM Trans. Graph.* **2017**, *36*, 1–14. [CrossRef]
2. Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; Huang, T. Free-Form Image Inpainting with Gated Convolution. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 1 March 2019; pp. 4470–4479.
3. Liu, G.; Reda, F.A.; Shih, K.J.; Wang, T.-C.; Tao, A.; Catanzaro, B. Image Inpainting for Irregular Holes Using Partial Convolutions. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention—MICCAI'99, Granada, Spain, 16–20 September 2018; pp. 89–105.
4. Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; Huang, T.S. Generative Image Inpainting with Contextual Attention. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 1–22 June 2018; pp. 5505–5514.
5. Han, S.; Ahmed, M.U.; Rhee, P.K. Monocular SLAM and Obstacle Removal for Indoor Navigation. In Proceedings of the 2018 International Conference on Machine Learning and Data Engineering (iCMLDE), Sydney, Australia, 3–7 December 2018; pp. 67–76.
6. Rakshith, S.; Mario, F.; Bernt, S. Adversarial scene editing: Automatic object removal from weak supervision. *arXiv* **2018**, arXiv:1806.01911.
7. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
8. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
9. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149. [CrossRef] [PubMed]
10. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask r-cnn. *arXiv* **2017**, arXiv:1703.06870.
11. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. *arXiv* **2015**, arXiv:1506.02640.
12. Zhao, Q.; Sheng, T.; Wang, Y.; Tang, Z.; Chen, Y.; Cai, L.; Ling, H. M2Det: A Single-Shot Object Detector Based on Multi-Level Feature Pyramid Network. *Proc. Conf. AAAI Artif. Intell.* **2019**, *33*, 9259–9266. [CrossRef]
13. Kim, K.-B. ART2 Based Fuzzy Binarization Method with Low Information Loss. *J. Korea Inst. Inf. Commun. Eng.* **2014**, *18*, 1269–1274. [CrossRef]
14. Lee, H.C.; Kim, K.B.; Park, H.J.; Cha, E.Y. An a-cut Automatic Set based on Fuzzy Binarization Using Fuzzy Logic. *J. Korea Inst. Inf. Commun. Eng.* **2015**, *19*, 2924–2932. [CrossRef]
15. Kingma, D.P.; Welling, M. Auto-encoding variational Bayes. *arXiv* **2013**, arXiv:1312.6114.
16. Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv* **2014**, arXiv:1411.1784.
17. Zhang, Z.; Song, Y.; Qi, H. Age progression/regression by conditional adversarial autoencoder. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5810–5818.
18. Sohn, K.; Lee, H.; Yan, X. Learning structured output representation using deep conditional generative models. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 3483–3491.
19. Bishop, C.M. *Mixture Density Networks*; Technical Report; NCRG/94/004; Aston University: Birmingham, UK, 1994.
20. Hershey, J.R.; Olsen, P.A. Approximating the Kullback Leibler Divergence between Gaussian Mixture Models. In Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP '07, Honolulu, HI, USA, 15–20 April 2007; Volume 4, p. IV-317.
21. 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 (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.
22. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Gradcam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Venice, Italy, 22–29 October 2017; pp. 618–626.
23. Zeiler, M.D.; Taylor, G.W.; Fergus, R. Adaptive deconvolutional networks for mid and high level feature learning. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2018–2025.
24. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 818–833.
25. Chatfield, K.; Simonyan, K.; Vedaldi, A.; Zisserman, A. Return of the devil in the details: Delving deep into convolutional nets. *arXiv* **2014**, arXiv:1405.3531.
26. Lin, M.; Chen, Q.; Yan, S. Network in network. *arXiv* **2013**, arXiv:1312.4400.
27. Chen, X.; Duan, Y.; Houthooft, R.; Schulman, J.; Sutskever, I.; Abbeel, P. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 2172–2180.
28. LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Comput.* **1989**, *1*, 541–551. [CrossRef]
29. Behnke, S. *Hierarchical Neural Networks for Image Interpretation*; Springer Science and Business Media LLC: Berlin, Germany, 2003; Volume 2766, pp. 64–94.

30. PSimard, P.; Steinkraus, D.; Platt, J. Best practices for convolutional neural networks applied to visual document analysis. In Proceedings of the Seventh International Conference on Document Analysis and Recognition, Edinburgh, Scotland, 3–6 August 2003; Volume 2, pp. 958–963.
31. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
32. McCulloch, W.S.; Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biol.* **1943**, *5*, 115–133. [CrossRef]
33. Rosenblatt, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychol. Rev.* **1958**, *65*, 386–408. [CrossRef] [PubMed]
34. Rumelhart, D.E.; Hinton, G.E.; McClelland, J.L. A general framework for parallel distributed processing. In *Parallel Distributed Processing*; MIT Press: Cambridge, MA, USA, 1986; Volume 1, pp. 45–76.
35. Eskicioglu, A.; Fisher, P. Image quality measures and their performance. *IEEE Trans. Commun.* **1995**, *43*, 2959–2965. [CrossRef]
36. Sheikh, H.R.; Bovik, A.C. Image information and visual quality. *IEEE Trans. Image Process.* **2006**, *15*, 430–444. [CrossRef]
37. Hore, A.; Ziou, D. Image Quality Metrics: PSNR vs. SSIM. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 2366–2369.
38. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Proc.* **2004**, *13*, 600. [CrossRef] [PubMed]
39. Zhou, B.; Khosla, A.; Lapedriza, A.; Torralba, A.; Oliva, A. Places: An image database for deep scene understanding. *arXiv* **2016**, arXiv:1610.02055. [CrossRef]
40. Liu, Z.; Luo, P.; Wang, X.; Tang, X. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August* **2018**, *15*, 2018.
41. Krizhevsky, A.; Hinton, G. *Learning Multiple Layers of Features from Tiny Images*; Technical Report; University of Toronto: Toronto, ON, Canada, 2009.