Finding Generative Image LoRA Model by Inputting Style Sample Image

NgocAnh VUTHI†, Yoshiyuki SHOJI†, Huu-Long PHAM††, and Hiroaki OHSHIMA††

† Faculty of Informatics, Shizuoka University
Johoku, Hamamatsu-shi, Shizuoka 432 – 8011, Japan
†† Graduate School of Information Science, University of Hyogo
Gakuen-nishimachi, Nishi-ku, Kobe, Hyogo 651–2197, Japan
E-mail: †vu.thi.ngoc.anh.20@shizuoka.ac.jp, ††shojiy@inf.shizuoka.ac.jp, †††huulongpham28@gmail.com, ††††ohshima@ai.u-hyogo.ac.jp

Abstract This paper proposes a method for finding fine-tuned image-generation LoRA models suitable for generating images that have a similar style to a given sample image. Sharing trained image-generation AI models on the Internet is becoming common. However, to determine which of these models can generate the image style the user wants, it is necessary to inspect the output samples posted on the site one by one. To enable search for the model that can generate an image that has a similar style to a given image, we adopt a CNN-based multi-class classifier. The ResNet50 model was fine-tuned to classify the distinctive features of each LoRA model. We transformed pre-prepared sample images with each LoRA model using img2img as training data. Experimental results show that our customized ResNet50 can find more correct LoRA models than the original ResNet50. Future research should explore more effective methods to address the unique challenges presented in this novel approach.

Key words Finding model, Stable Diffusion model, LoRA model, CNN.

1 Introduction

Even though it has been only a short time since image-generative AI was released, people have already begun to open and share their fine-tuned models. Stable Diffusion, one of the most famous image-generative AI models, was released in 2022. This model generates images by processing input texts (*i. e.,* prompts), or a sample image. While the base Stable Diffusion model excels in various tasks, it encounters challenges in generating specific styles.

Rather than adjusting the prompt to generate specific styles, a more practical approach involves utilizing custom models that are fine-tuned models with images of the target sub-genre. These customized models, known as checkpoint models and LoRA models, offer enhanced performance. This paper focuses on custom LoRA models, demonstrating proficiency in generating specialized style images. Within one year, file-tuned LoRA models of sharing became common.

Specifically, on a website called "Civitai(注1)", many image-generative file-tuned LoRA models are shared. From anime to realistic style, they are fine-tuned with domain images and uploaded there. Users can freely download models and generate images of what they want.

Despite the proliferation of such models, subtle variations exist among those with similar styles, complicating user attempts to find a LoRA model that closely aligns with their desired output. When users want to find models that can be generated to style what they wish, they need to check the thumbnail images of each model to know whether this model can be generated to style what they want. This makes users waste a lot of time.

If thumbnail images of the model exist, the model can be found by the above methods. This method depends on existing thumbnail images. Also, adjusting hyperparameters (strength, guidance scale, etc.) of fine-tuned models when generating images can generate different style images, although using the same fine-tuned model. So that that model can generate the style the users want.

As described above, the fine-tuned image generation model can generate various styles of images by adjusting the hyperparameters and prompts. This means that even if a fine-tuned image generation model does not exist prior, if a model generated close to the style image that the user wants can be retrieved, an image in the desired style can be generated.

This paper proposes a method for obtaining an image-generative LoRA model using a single-style sample image. For example, when a user inputs a single sample image of Van Gogh, the system outputs models that can generate a painting style that is the same or close to the input image.

(注1)：Civitai: The Home of Open-Source Generative AI
https://civitai.com/

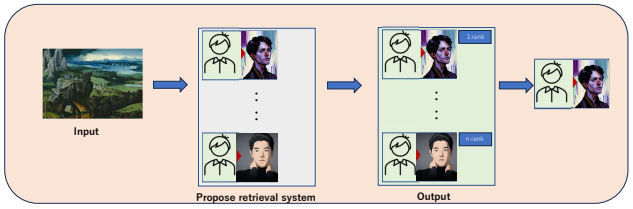

Figure 1 Overview of the retrieval system

Since multiple models may be output, model ranking is performed. The model that generates the closest match to the style of the input image is placed at the top of the list.

Figure 1 shows an overview of our system. It accepts a style image as its input, and outputs the ranking of models order by the possibility of generating a given image. To rank the models, we used a ResNet50-based classifier.

Our research aims to help users easily find fine-tuned image generative models without checking the thumbnail images of LoRA models. Moreover, it supports users in finding the alternative image generative model when the model they really want does not exist.

This system is necessary not only for users who want to generate an image, but also for artists and copyright holders. Artists and copyright holders may have to check if their original artworks were illegally used to train the model without agreement. When fine-tuning Stable Diffusion with specific domain images, it is necessary to use a lot of images in this domain. A fined-tuned model can only generate the style images used during fine-tuning. If the user is an artist with many well-known works, the painter may suspect their works are being used illegally. By using this system, the user can check for such illegal use. By inputting an image into the system, the system outputs a model that can generate a painting style similar to that of the image.

To find image-generative fine-tuned models, we used pre-trained ResNet50. To adapt pre-trained ResNet50 in our proposal, ResNet50 is fine-tuned with image classification tasks.

To create an image for fine-tuning ResNet 50, Stable Diffusion version 1.5, ControlNet - Canny, and LoRA models are used. Further, to boost ResNet50's performance with the given task, a small part of the original ResNet50 architecture is modified. By fine-tuning the pre-trained Resnet50 with the style-transformed image classification task, the fine-tuned Resnet50 obtains higher accuracy than the pre-trained ResNet50. Macro-recall, Macro-precision, and Macro-F1 scores are used to evaluate the image classification task.

Our paper is structured as follows. In this section, we describe the background and objectives of this study. Section 2 describes related studies and the position of these studies in related fields. Section 3 describes the methodology proposed in this study, and section 4 describes the evaluation method of the proposed methodology. Section 5 discusses the results obtained through experiments. Finally, we summarize this study and discuss future prospects.

2 Related Work

This study aims to help users find image-generative fine-tuned LoRA models more easily by inputting a sample image. This study relates to the image-generative AI field. Also, this study relates to the classification of images and related to the content-based image retrieval field.

2.1 Image Generative AI

In recent years, the field of artificial intelligence has witnessed remarkable advancements, especially in the domain of computer vision. One of the most intriguing developments is the emergence of generative AI image models, which have revolutionized how we perceive and interact with visual data. These sophisticated algorithms are at the forefront of the AI revolution, enabling machines to generate realistic and creative images.

First, we explain the concept of generative AI image models and their significance in the world of generative AI development. Generative AI refers to a class of artificial intelligence models that can generate data resembling authentic examples from the dataset they were trained on. In the context of images, generative AI aims to create new images that resemble the distribution of images in the training dataset, exhibiting creativity and imagination. The stable Diffusion model is a type of generative AI. It is used not only to generate images, but also to generate training data for supervised machine learning [1].

Model Agnostic Zero-Shot Classification is a method to classify real images by learning with synthetic images. Using the Stable Diffusion model improves the quality of the training dataset and solves problems relating to large-scale vision-language models [2].

Class incremental learning aims to learn new classes without forgetting previously learned classes incrementally. Several research works have shown how additional data can be used by incremental models to help mitigate catastrophic forgetting. A Stable Diffusion model [3]can generate synthetic samples belonging to the same classes as the previously encountered images.

Computer-aided surgical systems can provide surgeons with supportive information to improve procedure execution and overall outcomes. The Stable Diffusion model [4] of image-to-image task can reduce the gap between synthetic and real data. This makes computer-assisted surgical systems more accurately annotate data.

2.2 Image Classification by CNNs

CNN-based architecture is used for object detection and pattern recognition. Object detection is highly attended to in computer vision, but classification image style is not so attended to. Hence, there are few prior works that use CNNs to classify image styles.

CNN-based VGG-19 is adopted to extract object features and texture features [5]. Using a deep convolutional net to extract image features [6]. Although models were trained for object detection, these models can be used to classify aesthetics and styles.

CNNs are adapted to identify artists [7]. It has been discovered that CNNs can learn illustration styles from images. An ensemble of convolutional neural networks is adapted to learn the surface of image data, utilizing a CNN-based multiple neural network architecture [8]. The output of the CNN allows for the prediction of attribute images.

2.3 Content-Based Image Retrieval

In recent years, content-based image retrieval has become common in a variety of fields, such as product search and similar image retrieval. With the development of e-commerce sites, customers can search for and purchase just about anything on e-commerce. However, too much information hinders the search for information. Images [9] can be used to search for information easily.

Content-based image retrieval (CBIR) is a widely used technique for retrieving images from huge and unlabeled image databases. To help users more easily retrieve images [10], deep learning frameworks based on Convolutional Neural Networks (CNN) for feature extraction and Support Vector Machine (SVM) for classification are adopted. In this paper, instead of using SVM, pre-trained Resnet50 of the fully connected layer is used for classification. Fully connected layers of weights are initialized with ImageNet weights that obtain better results.

By retraining the CNN models with classification or similarity learning objective on the new domain, [11] found that the retrieval performance could be boosted significantly, which is much better than the improvements made by similarity learning.

Jin *et al.* [12] propose a method for searching images that share the same spatial semantics and enjoy visual consistency as the query image in complex scenes. They adopted CNNs to extract image features.

Adopting deep learning architecture to extract content from query images is common in the content-based image retrieval field. Instead of using text to search content, using images offers more information for the retrieval system, introducing the closest results that users expect.

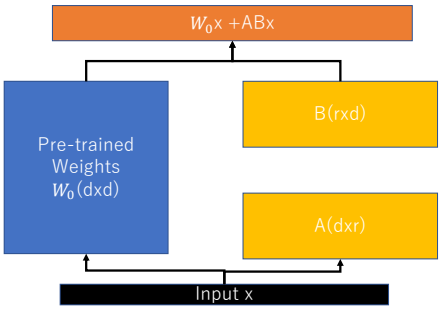

Figure 2 Low Rank Adaptation

3 Classification to Identify Model Capable to Generate Exemplified Styled Image

This section describes the proposed method. We explain the method in two sections: The image generation process, and Resnet50 fine-tuning task.

3.1 Image Generation Models

This sub-section describes the image generation process by fine-tuning image-generative AI models. Latent stable diffusion models [13], ControlNet-Canny [14], and Stable Diffusion fine-tuned by LoRA [15]are implemented. Latent stable diffusion models are exploited based on diffusion models. By training on latent space and introducing cross-attention layers into diffusion models. Latent stable diffusion models are more powerful and flexible generators for general conditioning inputs such as prompt or bounding boxes and high-resolution synthesis.

When enabling a pre-trained large model into the specific task, it is necessary to add a task-specific head or update the weights of the pre-trained large model through backpropagation during the training process. This process demands a lot of GPU memory. If small-scale data is used for full fine-tuning, the fine-tuned model can be catastrophic forgetting. This caused harmful to infer phase.

In Figure 2, instead of fine-tuning all the weights of the base model, the LoRA (low-rank adaptation) method decomposes the weights of the base model into matrix A and matrix B. The new A and B matrix weights are only updated during retraining; the weights of the base model are fixed. The LoRA method is being introduced for the first time through the fine-tuning of a large-scale language model. However, it can be applied to any dense layers in deep learning models. When fine-tuning Stable Diffusion models, the cross-attention layers in Stable Diffusion models are trained by LoRA. The cross-attention layers are the part of the model where the image and the prompt meet.

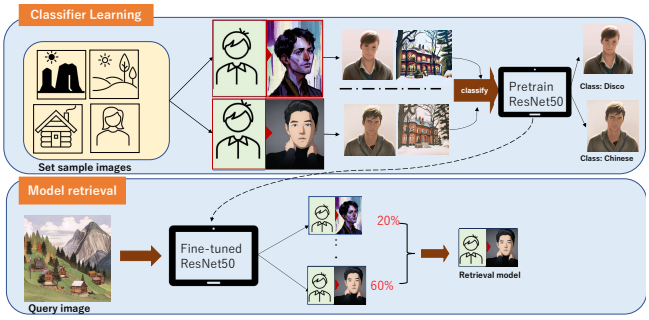

Figure 3 Image Classification and Model Retrieval Process

Using specific domain images and adopting LoRA for fine-tuning, fined-tuned Stable Diffusion models with specific styles are created; we call it the xx LoRA model. When inferring or generating, the weight of the fine-tuned LoRA model and the weight of the base model (Stable Diffusion model) are merged to implement the task. For instance, the anime LoRA model, the Chinese Ink LoRA model, etc.

With the advent of text-to-image Stable Diffusion models [13], we can create visually stunning images by inputting prompts. This paper proposes a method to classify the LoRA model style through a generated image. For this reason, image-to-image Stable Diffusion models are adopted to transfer base images into LoRA-style images. However, generating new images by image-to-image task without a prompt is still challenging.

The reason is related to the diffusion model mechanism, which uses image reconstruction. First, diffusion models noise input images by Markov chain, then reconstructing the image from the noised image via the denoising process. This means diffusion models can not retain anything about inputted images. This causes generating images through image-to-image to be stuck. ControlNet is adapted to address this challenge in the image-to-image task. The neural network architecture of ControlNet is described in Figure 4. We adopted ControlNet with Stable Diffusion model version 1.5 as the base model.

The ControlNet architecture based on the Stable Diffusion model version 1.5 works as below. ControlNet copies 12 encoder blocks and one middle block weight of the Stable Diffusion model and uses them as trainable parameters. As illustrated in Figure 4,

- x is the input features map,
- c is the internal vector calculated based on x, ControlNet of input,
- y is the output feature map.

The trainable parameters are connected to the base model with zero convolution layers, denoted $Z(\cdot;\cdot)$. Specifically, $Z(\cdot;\cdot)$ is a 1 x 1 convolution layer with both weight and bias initialized to zeros. To build up a ControlNet, two instances

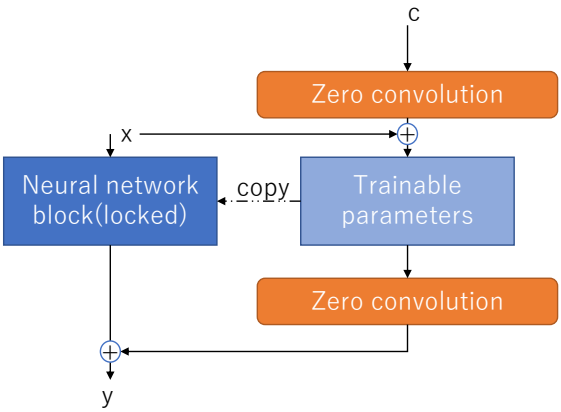

Figure 4 ControlNet neural network

of zero convolutions with parameters Θ z1 and Θ z2, respectively, are used. Although convolution layers of weight and bias are initialized to zeros, ControlNet still learns and updates parameters. We demonstrate the following formula:

$$y = Wx + b, \tag{1}$$

$$\frac{\partial y}{\partial w} = x, \tag{2}$$

$$\frac{\partial y}{\partial x} = W, \tag{3}$$

$$\frac{\partial y}{\partial b} = 1, \tag{4}$$

$$\tag{5}$$

where y denotes output, W denotes weight, x denotes input, b denotes bias. If W and b are zero and x is not zero, y is different from zero. This means that when images are input, the weight and bias of ControlNet are grown up. When training ControlNet, the weight and bias of the stable diffusion model are locked, and only trainable weights are updated by gradient descent. ControlNet offers various conditional controls in the Stable Diffusion model. In our experiment, we used the ControlNet-Canny condition. Via ControlNet-Canny conditional control, we can extract the outline of the subject in each base image, then transfer object style with the Stable Diffusion model and LoRA models.

3.2 Classifier Using Fine-tuned Resnet50

In the context of the classification problem, shallow layers would [16] extract low-level features that are almost the same as the input image. In deeper layers, more abstract features were extracted. The abstract features make the classification possible and easier. Hence, more layers are added to network architecture in the hope that the network can learn more features from input images and boost classification accuracy. However, when more layers are adapted, a degradation problem is caused. This problem makes training errors and test/validation errors go up [17]. By introducing a residual learning framework into CNN, Resnet50 [17] is easier to

optimize and can gain higher accuracy. The deep residual learning framework of ResNet50 is shown in Figure 5. As shown in Figure 5, the x identity map feature is added to network weight before activating network weight forward. Instead of output $F(x)$, the output changes into $F(x) + x$. This makes deep layers learn shallow layers of features and decreases errors in training and test/validation.

In this paper, Resnet50 is adopted to classify LoRA style via transferred images generated from fine-tuned image generative LoRA models. To enhance classification accuracy, pre-train Resnet50 is modified and fine-tuned.

3.3 Classification-based Model Search

In this section, we explain more clearly about our method. First, we collected sample images. The sample image denotes which image we use to transform its style. Sample images are variously selected. Next, we prepared the fine-tuned image generative LoRA models. As mentioned in the previous section, to transfer the style of base images, image-to-image Stable Diffusion model version 1.5, ControlNet canny model, and LoRA models are adopted. At first, the subject in each base image of the outline is extracted and transformed into a vector. We call it a conditional control vector. Then, the Stable Diffusion model version 1.5 and the Canny model of ControlNet are implemented. In the stage of applying the LoRA models style, weight in u-net layers and text encoder layers of LoRA [15] are set with 1. Hence, enabling the LoRAs feature is clearly reflected in the transferred image. The base image and conditional control vector are used when transforming the base image style without any text input. In addition, hyperparameters such as inference steps, guidance scales, etc... are adjusted so that the transformed image better reflects LoRA features.

After transforming the style of base images, the fine-tuning phase is implemented. We fine-tune pre-trained ResNet50 with image classification tasks. Data is transformed style base images, and the label is the name of LoRA models. In our experiment, the training dataset is very tiny compared with the ImageNet dataset used for training ResNet50 [17]. For this reason, only small parts of pre-trained ResNet50 are returned, instead of returning the entire model. As mentioned in [18], deeper CNN layers can extract more abstract features of the image. These features represent a unique style of image. ResNet50 architecture has four major residual layers, which are assigned to extract the abstract features of each image. Moreover, each layer has a different amount of bottleneck [17]. The third layer of Resnet50 has the most number of bottlenecks, with six bottlenecks. Based on the above mentioned, weights between layer 3 and the last fully connected layer are set backpropagation. The primary role of the CNNs is feature extraction. Our main purpose is image classification, thus more emphasis on fully connected layers. One more fully connected layer is added.

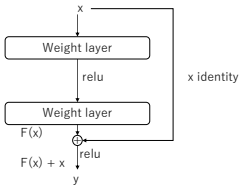

Figure 5 Residual block

4 Evaluation

We evaluate the proposed retrieval system through three factors. First, we tried to clarify whether the designed fine-tuning is effective or not. Second, we evaluated what kind of dataset can be used to obtain good performance for the proposed system. Finally, we evaluated how many style-transformed images should be included in each training data class to ensure sufficient learning.

4.1 Fine-tuning result

This section introduces the evaluation method for the proposed method. The base images are downloaded from the Internet. We call this the base image set as set B. The set B is processed to include as many different types of images as possible. Images of people, scenery, food, plants, animals, etc., are included in the set B. Each element in the set B is unique. The set B is transformed style by group models, which are the stable diffusion, canny, and LoRA models. With each image in the B set creates one image in the F set. After transforming the style, we create a transformed style image set called the F set. The size of B set and F set are the same, both set include 126 images. Such as, $(\forall x \in B | \exists x' \in F, |B| = |F|)$, where

- x : base image,
- B : base image set,
- x': transformed style image, and
- F: transformed style image set.

In the fine-tuning phase, 80 percent of the F set is trained, and 20 percent of the set is used for validation.

Table 1 shows the list of models that we used in our experiment. As the search target, 12 LoRA models are prepared. All these LoRa models are downloaded from Civitai. In the experiment, 15 epochs and a 0.001 learning rate are used. The number of nodes in the output of the fully connected layer is modified to match the number of LoRA models we wish to classify.

When returning, early stopping with five patients and the learning schedule CosineAnnealingLR are utilized to obtain the best performance fine-tuned ResNet50 model.

As shown in Figure 6, training and validating results are

Table 1 Models used in experiment

Model name
Anime Screencap Style LoRA (注2)
Anime Game Backgrounds(注3)
xiaorenshu(注4)
Chinese ink painting(注5)
Disco Elysium / Aleksander Rostov / Oil paints & Abstraction / Style(注6)
Fairy in Clouds(注7)
XSarchitectural-11Fantasyarchitecture(注8)
Phantasmal Luminous: The Radiance of Rainbow Dispersion(注9)
8bitdiffuser 64x — a perfect pixel art model(注10)
pop art(注11)
Texture illustration(注12)
Realistic Dating attire(注13)

plotted. When fine-tuning pre-trained ResNet50 with 12 LoRA models, the best result was obtained in the 6th epoch. In the loss plot, the learning loss decreases gradually as the number of learning epochs increases. Nevertheless, the validation loss decreases from the first to the third epoch, but becomes very unstable from the fourth epoch to the end. The same phenomenon is observed in the accuracy plot.

Then, to evaluate the effect of fine-tuning, we compare the classification performance of the fine-tuned ResNet50 with the classification results of the pre-trained ResNet50. Here, we use Accuracy and Macro-F1. In each experiment, shown in 2, the same set of validation data is used. Also, the number(100, 26), in second and third column, denoting amount of images included in each class.

With pre-trained ResNet50, the output of a fully connected layer is modified from 1000 to 12 to adapt with our task. When inferring, weights of pre-trained ResNet50 are frozen. The detailed result is shown in 2. The our ResNet50 performs better than the pre-trained ResNet50 in the three measures of Accuracy, Macro-F1 and MRR. We calculated the Macro-F_i score as

$$Macro - F1 = \frac{1}{N}\sum_{i=0}^{N}(\frac{2TPi}{2TPi + FPi + FNi}) \qquad (6)$$

where

- N: Amount of models(N = 12),
- TP: True positive,
- FN: False negative, and
- FP: False positive.

Via Accuracy and Macro-F1, we can evaluate classification performance of ResNet50. Via MRR, we can evaluate capability retrieval model of proposed system.

In addition, the more images included in each class, the better the system's learning performance. However, once a certain amount is reached, the change becomes small. Specifically, even if 70 or more images are included in each class, no significant change in learning accuracy occurs. In conclusion, 70 images in each class are sufficient to train the proposed retrieval system.

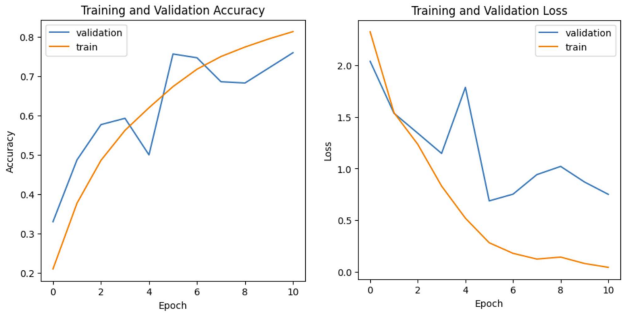

Figure 6 12 models of classification result

4.2 Experiment Model Retrieval

We prepare more than one sample image dataset to evaluate the importance of training data variety. In the above dataset, we call it a variety dataset, which includes a variety of images such as humans, animals, landscapes, *etc.*. The new one, we call it a specialized domain image dataset that contains only human face images. Each image in the domain image dataset is processed like an image in the variety dataset.

The result is shown in table 2 Fine-tuning the pre-trained ResNet50 using the same number of data, the retrieval system outperforms when using diverse data. In conclusion, diverse data images are important for building the proposed retrieval system.

A fine-tuned ResNet50 model with 12 models is also used to find LoRA models. Found models are ranked based on the output of the second fully connected layer of ResNet50. Images are used in the experiment, not included in the training or validating sets.

To enable us to evaluate objectively, these images are downloaded from the site called Civitai, and they are thumbnail images of the original LoRA models created by authors.

Table 2 Evaluation Results

	Train data	Validation data	Accuracy	Macro-F1	MRR
Fine-tuned ResNet50	100 variety images	26 variety images	79.50	74.37	60.92
Pretrained ResNet50		26 variety images	8.30	Nan	
Fine-tuned ResNet50	70 variety images	26 variety images	76.90	70.17	62.15
Fine-tuned ResNet50	50 variety images	26 variety images	67.90	65.50	55.09
Fine-tuned ResNet50	30 variety images	26 variety images	62.20	60.48	60.42
Fine-tuned ResNet50	70 human face images	26 variety images	50.30	48.33	47.92

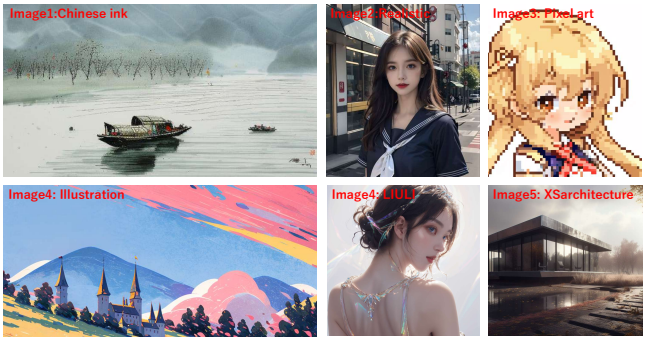

Figure 7 Experimental images

Figure 8 Image generation sample

Experimental images are shown in Figure 7. The red letters in the top left of each image are the names and correct labels of each image. A correct label indicates that the image is generated from that LoRA model.

Finding model results by a sample image is shown in Table 3. Via our method, among the six sample images, three sample images can accurately retrieve the original LoRA model. When the LoRA model is retrieved by feeding the experimental images into the proposed system, the correct results are included within the top six.

4. 3 Image generation

This section uses the retrieved LoRA models from the previous section to generate images. Following the result of Table 2, the search result is the most undesirable when using image 2 to find the LoRA model that generated image 2. The desirable search result is located in the sixth position. In the experiment of image generation, the first and sixth models are used to generate images and compare them. Figure 9 shows the four images in our experiment, such as

Depending on each LoRA model, the generated image does not exactly resemble the base image. However, they were converted to their respective painting styles. Indeed, the style of the image transformed by the Realistic LoRA model is more similar to the style of the reference style image than the style of the image transformed by the XSarchitecture LoRA model. However, when users want to generate images, they want the generated images to follow their desired style, and the generated images are beautiful. In this experiment, in the two generated images, we are not able to say that the image transformed by the Realistic LoRA model is more aesthetic than the image transformed by the XSarchitecture LoRA model.

5 Disccusion

Through this experiment, it is certainly possible for the proposed method to search for fine-tuned image-generative LoRA models. However, to retrieve the LoRA model accurately, the number of LoRA models fed in ResNet50 is limited. Furthermore, since the models are retrieved through style-transformed images, amount of models that can be retrieved may depend on the process of image style transformation. There are many hyper-parameters that need adjusting during the process of transforming the style of the image. Thus, when generating more images, the image generation process is more complicated and not sturdy.

Moreover, the number of fine-tuned LoRA models gradually increases, and in this proposal, ResNet50 will need to return when new fine-tuned LoRA models are released. This is inefficient. ResNet50 is specialized to extract image style features. Although the original ResNet50 is modified for best performance on a given task, its functionality is still limited when a number of LoRA models are added to ResNet50. We have confirmed that the capability of ResNet50 is not appropriate for our research. We need to investigate more effective methods in the future.

One of effective method is the Siamese Network, employing a loss function called contractive loss or triplet loss during

Table 3 Result of retrieving model

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th
Image1	**Chinese ink**	XSarchitecture	Illustration	Anime	Pop art	Realistic	Pixel art	LIULI	Cloudy
Image2	XSarchitecture	Anime	cloudy	Illustration	Chinese ink	**Realistic**	Pixel art	LIULI	Anime background
Image3	**Pixel art**	Pop art	Illustration	Anime background	Realistic	Chinese style	XSarchitecture	Anime	LIULI
Image4	LIULI	**Illustration**	Anime background	Cloudy	Pop art	Disco	XSarchitecture	Anime	Realistic
Image5	Cloudy	XSarchitecture	Illustration	**LIULI**	Anime	Pop art	Anime background	Realistic	Chinese ink
Image6	**XSarchitecture**	Cloudy	Anime	Illustration	Chinese ink	Realistic	Pixel art	LIULI	Pop art

training. The Siamese Network takes two or three images as input values and calculates the distance between them. Based on distance, determine if they are the same image or not. When applying to our research, we assume that one image is the style the user wants and the other is a thumbnail image of the original LoRA model. The above process makes it possible to eliminate the work of image style transformation.

6 Conclusion

This paper proposes a method for retrieving fine-tuned image-generating LoRA models. Through our method, it is possible to output a LoRA model that can generate an image style similar to the style of the reference image. A dataset of various base images is prepared to implement the proposed method. The style of the sample images is transformed by using Stable Diffusion version 1.5, ControlNet-Canny, and fine-tuned image generation LoRA models. The style-transformed images are used as fine-tuning data for the pre-trained ResNet50. Several parts of the original ResNet50 architecture are modified to adapt it to our task. Experiments are conducted to compare pre-trained ResNet50 and fine-tuned ResNet50 and search fine-tuned image generation LoRA models.

Acknowledgement

This work was supported by JSPS KAKENHI Grants Numbers 21H03775, 21H03774, and 22H03905.

References

[1] Gabriele Valvano, Antonino Agostino, Giovanni De Magistris, Antonino Graziano, and Giacomo Veneri. Controllable image synthesis of industrial data using stable diffusion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 5354–5363, 2024.

[2] Jordan Shipard, Arnold Wiliem, Kien Nguyen Thanh, Wei Xiang, and Clinton Fookes. Diversity is definitely needed: Improving model-agnostic zero-shot classification via stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 769–778, 2023.

[3] Quentin Jodelet, Xin Liu, Yin Jun Phua, and Tsuyoshi Murata. Class-incremental learning using diffusion model for distillation and replay. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3425–3433, 2023.

[4] Joanna Kaleta, DiegoDall ' Alba, Szymon Płotka, Przemysław Korzeniowski. Minimal data requirement for realistic endoscopic image generation with stable diffusion. *International Journal of Computer Assisted Radiology and Surgery*, pp. 1–9, 2023.

[5] Tiancheng Sun, Yulong Wang, Jian Yang, and Xiaolin Hu. Convolution neural networks with two pathways for image style recognition. *IEEE Transactions on Image Processing*, Vol. 26, No. 9, pp. 4102–4113, 2017.

[6] Sergey Karayev, Matthew Trentacoste, Helen Han, Aseem Agarwala, Trevor Darrell, Aaron Hertzmann, and Holger Winnemoeller. Recognizing image style. *Proc. Brit. Mach. Vis.Conf. (BMVC)*, 2014.

[7] Kazuma Kondo and Tatsuhito Hasegawa. Cnn-based criteria for classifying artists by illustration style. IVSP '20, p. 93–98, New York, NY, USA, 2020. Association for Computing Machinery.

[8] Fang Ji, Michael S McMaster, Samuel Schwab, Gundeep Singh, Lauryn Nicole Smith, Shishir Adhikari, MarcioO ' Dwyer, Farah Sayed, Antony Ingrisano, Dean Yoder ほか. Discerning the painter ' s hand: machine learning on surface topography. *Heritage Science*, Vol. 9, No. 1, pp. 1–11, 2021.

[9] Farhan Ullah, Bofeng Zhang, and Rehan Ullah Khan. Image-based service recommendation system: A jpeg-coefficient rfs approach. *IEEE Access*, Vol. 8, pp. 3308–3318, 2020.

[10] Ouhda Mohamed, El Asnaoui Khalid, Ouanan Mohammed, and Aksasse Brahim. Content-based image retrieval using convolutional neural networks. In *Lecture Notes in Real-Time Intelligent Systems*, pp. 463–476. Springer, 2019.

[11] Ji Wan, Dayong Wang, Steven Chu Hong Hoi, Pengcheng Wu, Jianke Zhu, Yongdong Zhang, and Jintao Li. Deep learning for content-based image retrieval: A comprehensive study. In *Proceedings of the 22nd ACM International Conference on Multimedia*, MM '14, p. 157–166, New York, NY, USA, 2014. Association for Computing Machinery.

[12] Jin Ma, Shanmin Pang, Bo Yang, Jihua Zhu, and Yaochen Li. Spatial-content image search in complex scenes. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2503–2511, 2020.

[13] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10674–10685. IEEE, 2022.

[14] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3836–3847, 2023.

[15] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[16] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5188–5196, 2015.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE*

Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.

[18] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

(注2)：civitai: https://civitai.com/api/download/models/60568?type=Model&format=SafeTensor
(注3)：civitai: https://civitai.com/api/download/models/19065?type=Model&format=SafeTensor&size=full&fp=fp16
(注4)：civitai: https://civitai.com/api/download/models/25661?type=Model&format=SafeTensor&size=full&fp=fp16
(注5)：civitai: https://civitai.com/api/download/models/42314?type=Model&format=SafeTensor
(注6)：civitai: https://civitai.com/api/download/models/169568?type=Model&format=SafeTensor
(注7)：civitai: https://civitai.com/api/download/models/269819?type=Model&format=SafeTensor
(注8)：civitai: https://civitai.com/api/download/models/30572
(注9)：civitai: https://civitai.com/api/download/models/180175
(注10)：civitai: https://civitai.com/api/download/models/231819?type=Model&format=SafeTensor
(注11)：civitai: https://civitai.com/api/download/models/188417?type=Model&format=SafeTensor
(注12)：civitai: https://civitai.com/api/download/models/113534?type=Model&format=SafeTensor
(注13)：civitai: https://civitai.com/api/download/models/228470?type=Model&format=SafeTensor