

Visual Model of Pattern Design Based on Deep Convolutional Neural Network

Jingjing Ye¹ and Jun Wang^{2*}

¹Architecture and Design College, Nanchang University
Nanchang, 330031, Jiangxi, China
[e-mail: 357966278@qq.com]

²Fashion Institute, Donghua University,
Shanghai, 200051, China
[e-mail: jingjingye81@163.com]

*Corresponding author: Jun Wang

*Received October 23, 2023; revised December 19, 2023; accepted February 4, 2024;
published February 29, 2024*

Abstract

The rapid development of neural network technology promotes the neural network model driven by big data to overcome the texture effect of complex objects. Due to the limitations in complex scenes, it is necessary to establish custom template matching and apply it to the research of many fields of computational vision technology. The dependence on high-quality small label sample database data is not very strong, and the machine learning system of deep feature connection to complete the task of texture effect inference and speculation is relatively poor. The style transfer algorithm based on neural network collects and preserves the data of patterns, extracts and modernizes their features. Through the algorithm model, it is easier to present the texture color of patterns and display them digitally. In this paper, according to the texture effect reasoning of custom template matching, the 3D visualization of the target is transformed into a 3D model. The high similarity between the scene to be inferred and the user-defined template is calculated by the user-defined template of the multi-dimensional external feature label. The convolutional neural network is adopted to optimize the external area of the object to improve the sampling quality and computational performance of the sample pyramid structure. The results indicate that the proposed algorithm can accurately capture the significant target, achieve more ablation noise, and improve the visualization results. The proposed deep convolutional neural network optimization algorithm has good rapidity, data accuracy and robustness. The proposed algorithm can adapt to the calculation of more task scenes, display the redundant vision-related information of image conversion, enhance the powerful computing power, and further improve the computational efficiency and accuracy of convolutional networks, which has a high research significance for the study of image information conversion.

Keywords: Image processing, Texture vision sampling, Convolutional neural networks, Image denoising, Feature clustering

1. Introduction

With the passage of time, deep neural network technology has made rapid progress. Through human thinking driven by big data, it makes full use of neural network models. Neural network model and strong language expression ability can effectively overcome the texture effect expression of complex objects. In view of the limitations of neural network model in complex scenes, the user-defined template matching method takes into account the basic working principle to simplify the calculation process, which facilitates the simplification research of computational vision technology in many fields[1]. Texture inference method based on reference and self-defined template matching is considered. The initial stage of the combined convolutional network is based on the visualization of known target objects. In the initial stage of the automatic detection system, the similarity dimension between the scene to be inferred and the user-defined template is calculated. The texture of the user-defined template with the lowest similarity is regarded as the feature vector of the object in the current scene. Finally, the inference is carried out.

The model-based single image denoising method adopts a lot of physical prior knowledge as the basis. Furthermore, the physical modeling requires a lot of comprehensive knowledge. There are many factors to consider. Therefore, it is difficult to cover all aspects. It has been proved by practice that its denoising effect highly depends on the reliability and accuracy of the established model, and it can not be used in a wide range in the real world, and its generalization performance still needs to be strengthened. With the emergence of the convolutional neural network variants, the research on image denoising algorithm based on the convolutional neural network is becoming more and more mature. A self-encoder network is constructed by a nonlocally enhanced dense block, in which the weight values of the nonlocal feature map change following four densely connected convolutional layers [2].

Reference [3] proposes a basic structure of multi-dimensional feature processing, which can learn the external features of different scales. These sample parameters are gated and cycled. Then, make full use of the relevant information of the sample parameters with different scales. The concrete method is different from the previous multi-scale concrete methods. It explores and verifies the way of external scale connection to further improve the transmission quality and performance of display images. Reference [4] proposes a structural framework called multi-scale progressive fusion neural network connection to eliminate the noise formed in dual-display images. For the noise formed by the similarity caused by different nodes, the global texture effect is captured by means of recursive calculation. The noise formed by the feature target can be structured by exploring the two-dimensional complementary and redundant relevant information in the space. The basic structure of the multi-scale three-dimensional image is constructed. The attention control mechanism is further introduced to guide the fusion of information related to the relevant data from different scales. An end-to-end solution for detail recovery was proposed in reference[5], showing that images can be converted into neural network connections. This particular approach introduces another parallel sample parameter with a global activation function that can cooperatively cancel the generated noise, and recover details lost due to noise cancellation. At present, the mainstream research method is only to strengthen the training of the database data that can be generated, but it is not very good for the one-sided quality and performance of the images displayed in the real world. Therefore, the real and credible research content of the neural network deep learning model is adopted, that is, human thinking changes from semi-supervised learning to displayed image domain, and learns to improve the physical process according to supervised execution. Reference [6] shows a

learning framework for convolutional neural networks, which is used to adaptively simulate the characterization of various noise degradations using self-supervised modules. It transfers the supervised noise reduction to the unsupervised case, and uses the noisy data to pair with the pseudo-labels generated by the target network to improve the robustness of image extraction. The proposed method effectively proves that the performance of the semi-supervised framework has been significantly improved with the improvement of the supervision mechanism, and at the same time, it also causes further thinking about the supervision mechanism.

The residual generated by the shallow image block is used to guide the deep image block, and the negative residual is predicted from coarse to fine, and finally the outputs of different blocks are fused. It is an important attempt to realize multi-scale feature fusion in the field of single image denoising, and the deep features are guided by shallow features to realize the denoising operation from coarse to fine. In this paper, it uses the deep neural network modeling to obtain the external features in the design and production of decorative patterns, and adopts the external features as sample-free supervised execution to complete the frame reduction of semi-supervised execution structure. Increased training for connections subject to noise factors is conducted. With the help of quantitative analysis of experiments, it is finally proved that the specific method proposed in this paper has better quality and performance than other fully supervised execution or semi-supervised specific methods; and it is more effective to introduce external feature data to show the custom module of image neural network model, and to deal with the quality and performance of 3D pattern images to obtain more design effects. It is relatively convenient to build an encoder/decoder architecture, design special standardization related parameters, and require conditions to activate each unit to achieve normalization processing.

2. Relevant work

2.1 Visual feature modeling

The basic work of the transfer machine learning algorithm is to select the computer operation to transfer the style type graph, and the feature definition presents different style types on the basic knowledge. The style type conversion of the artistic images is to establish style type feature clustering by referring to the combination of feature definition and style type map, which belongs to style type feature clustering. Sensory motivation is a deep learning model that observes pre-reinforcement training. Features can show adjustment information related to natural language in images [7]. Sensory motivation is the type of art architectural patterns, which is defined as the visual texture effect. Different patterns representing visual texture effects are described in a vector space, and feature clusters with the same probability distribution in perceptual ability in the space are the same visual texture effects [8].

Assuming that the visual texture effect in the space is uniform, the style and type transfer in the space is shown in Fig. 1.

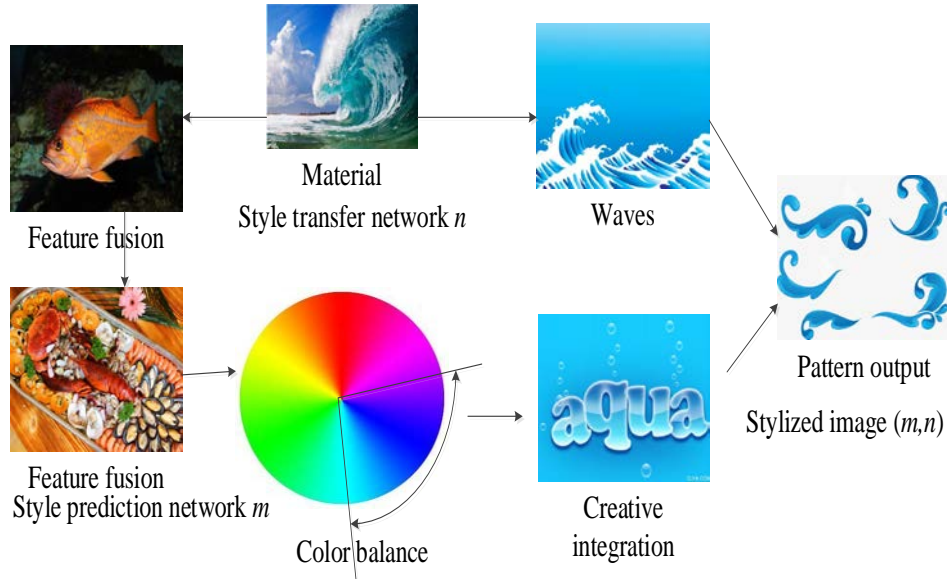


Fig. 1. Conversion of the style type texture model

One is the specific content and style type, and the other is the Hilbert multiple relation, which is used to measure the relative strength of style type [9]. The lowest and middle external features are connected to act as the low-level and high-level feature activation ρ of the deep learning model.

$$\rho = L_{(m,n)} + v_t^i (s.t \cdot R)^2 \quad (1)$$

In Equation 1, L is the connection activation of the first layer vector space, v_t^i is the entropy of all t units in the i layer, and $s.t \cdot R$ is the Gram vector space associated with the i layer [10].

2.2 Convolutional neural network

The basic structure of the convolutional neural network includes an input layer, multiple hidden layers and an output layer. Through the back propagation algorithm, the training data is input into the neural network, and the parameters of each convolution kernel are adjusted according to the error between the prediction result and the real label as to continuously optimize the accuracy of the network. In the testing/reasoning stage, the data is input into the convolutional neural network, and the result is obtained through the forward propagation algorithm.

The convolution kernel is an important part of the convolutional neural network, which slides on the whole input rain image like a window and performs convolution operation to extract the rain line characteristics and background information of the image. The convolutional neural generation tool is shown in Fig. 2.

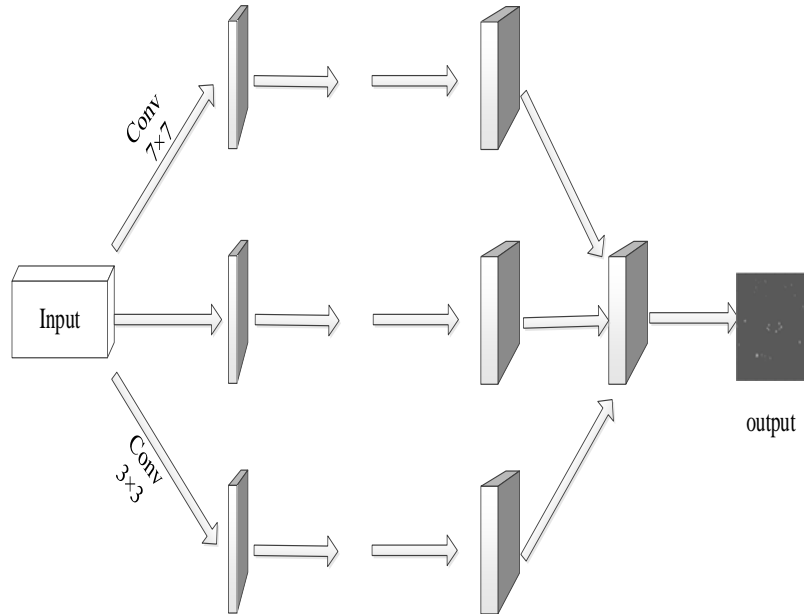


Fig. 2. Convolutional neural network architecture

According to the basic knowledge of SHM, the internal structure of the kernel generation tool for the hyper-convolutional networks further promotes the design of the hierarchical saliency model. This vertical distance from the environment adapts to different samples. Inspired by the supernet connection, with the help of the unified HKG neuron λ , the convolution kernel group is generated centrally for all HSM neurons γ [11]. The shared convolutional nerve is generated by a custom module τ . Each convolutional neuron corresponds to a saliency level θ . These shared hyper-convolutional network kernels will be parsed into different groups of convolutional network kernels through the lexical layer and then passed to the cascaded SHM custom module [12].

$$\lambda = \sum_{i=1}^n \alpha_{(x,y,z)}^1 + \alpha_{(x,y,z)}^2 + \dots + \alpha_{(x,y,z)}^n \quad (2)$$

In Formula 2, α is the neuron with unit 1, and (x, y, z) are the spatial positions of the pattern coordinates respectively.

3. Visual sensory semantic modeling

The Gram vector space is a parallelogram symmetric matrix used to directly measure the underlying structure of a layer of average relative data in the space required to activate the filtering device. It does not exclude any type of expression of learning art architectural patterns, and the focus of the work is to iteratively update feature clustering to synthesize visual texture effects or transfer unique artistic styles to feature clustering [13].

3.1 Visual semantic feature clustering

Converting from actual content feature clustering to a non-art rendered version is done. The target of reinforcement training is the combination of style type loss and specific content

loss, which is obtained by connecting the output of the vector space with the substitution of $v_i^j (s.t \cdot R)^2$ in the basic formula 1 [14].

The concatenated output is replaced with the combination of the loss due to the style type and the loss due to the specific content κ formed in Equation 1.

$$\kappa = \begin{cases} \varpi(m-1)^2 \\ \varpi(n-1)^2 \end{cases} \quad (3)$$

In Equation 3, ϖ is the centralized training of the relevant parameters of the style-type transmission connection, which is to minimize the specific content considered by the selected image pixels [15].

In this paper, the VGG19 model is chosen to extract the feature map of the image. The module proposed by VGG (Visual Geometry Group) contains 19 hidden layers, including 16 convolution layers and 3 fully connected layers. The overall structure is very simple. The whole network uses a convolution kernel with a size of 2×2 . Three 3×3 convolution kernels with a step size of 1 are equivalent to one convolution kernel with step 7×7 . The size of the final output feature map is 4, that is, the effect after convolution is the same, which can increase the depth of the network under the condition of the same receptive field. It can also improve the training effect of the neural network to a certain extent and reduce the number of parameters. The feature vector is shown in Fig. 3.

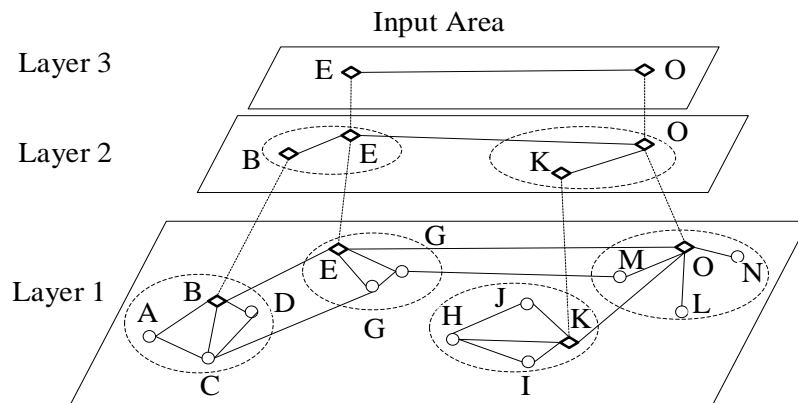


Fig. 3. Deep convolution feature vector

The perception of two convolution kernels with a step size of 1 is also equal to the perception of one convolution kernel, which is the input area of neurons in the sensory neural network.

The connections thus formed do not artistically further accelerate the rendering movement of feature clusters, learning separate connections for each type of the artistic architectural pattern style. The artistic architectural pattern style type has the common visual texture effect, natural color and natural language to analyze the scene displayed by the image [16]. Building a style type transport link that represents sharing between many pixels will give a fairly rich vocabulary to represent the semantics of architectural patterns. The general form is to construct an image decoding architecture design style conversion.

3.2 Image decoding normalization processing

The general approach is to establish a convenient encoder/decoder architecture style conversion network, but to specialize the specific normalization parameters of each architectural pattern style. This procedure is called conditional instance normalization, and it is recommended to normalize the activation η of each unit [17].

$$\eta = \sqrt{\kappa_{(m,n)} \cdot \rho_i^i} \quad (4)$$

In Equation 4, η is the mean and standard deviation of the entire spatial axis in the activation map. Constitute a linear transformation specifying the learned mean and the learned standard deviation of the unit [18].

The concatenation constitutes a multi-dimensional embedding vector, and the style transition network is represented as spanning all feature nodes. The research of reference [1] shows that such a network makes the response faster and saves time. Such a network provides a fast stylization of artistic styles. The embedding space is rich and smooth enough to allow users to combine architectural patterns by interpolating the learning embedding vectors of styles [19]. Although this is an important step forward, this y -type network is still limited compared to the original optimization-based techniques, as the network is limited to working on explicitly trained styles. The goal of this work is to extend this model to train $y \geq i, i = 1, 2, \dots, n$ styles and stylize unseen architectural patterns that have never been observed before [20]. The latter goal is particularly important because the extent to which the network generalizes about unseen architectural motif styles measures the extent to which the network (and the embedded space) represents the true breadth and diversity of all architectural motif styles [21].

4. Convolutional network feature vector model

The design of the neural layer saliency flattening model is conducted. The design of the saliency flattening model neuron is guided by the prior computational knowledge. Decoding that enables deep saliency detection occurs in regions where auto-adaptive saliency cannot be detected [22]. Reasonable saliency balance score is the key to the design of the neural saliency balance model. The custom module shares how to classify the saliency pyramid structure to classify the saliency level of each region with reference to the main features of the displayed image in the region where each camera pixel is located [23]. The specific process of classifier learning saliency is additionally controlled by prior computational knowledge. The ability of the corresponding learning target of the predicted sub-saliency subnet mask map is the saliency predicted by the composite function combined with prior computer knowledge [24].

4.1 Hierarchical neuron

With sufficient feature interaction, Meta Knowledge is embedded into the output hyperconvolution kernel. Each hyperconvolution kernel actually corresponds to a particular level of saliency in the respective SHM. Traditional work mainly uses Transformer to predict task-specific elements (detection box and segmentation map), while the algorithm in this paper aims to generate convolution kernel parameters for saliency decoder and improve the neural network capacity and reasoning flexibility of the overall framework. Therefore, a way of generating convolution kernels is proposed to model the significant differences between different samples.

Under the guidance of additional computational knowledge, under the help of various level model design methods guided by prior computer knowledge, the creation of models will show more targeted and predictable results. The implementation details of the salient hierarchical model design neuron are shown in Fig. 4.

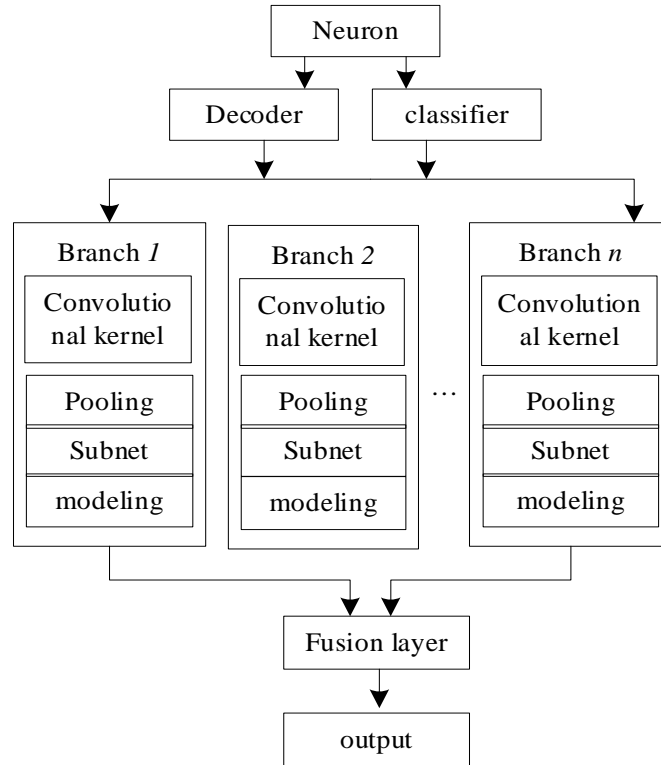


Fig. 4. Hierarchical modeling module

The design and fabrication of neurons is to be able to achieve a significant layering model design of the neural layer. The whole process can be roughly divided into two parts: the design of the special saliency external characteristics and the hierarchical saliency model after disassembly.

4.2 Coding optimization algorithm

In order to obtain the convolutional nerve, a number of custom modules τ are selected in HKG to serve as the custom modules for reaction generation. In addition, time can be queried between multiple learnable dimensional layers and flattened display image patches to construct an overwhelming attention focus control mechanism. The custom module τ consists of video file custom modules stacked in σ layers [25]. Each thin basic structure of τ constructs an interactive operation between a learnable saliency query space vector and a flat extrinsic feature block of the displayed image [26].

$$\tau = \begin{cases} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & \cdots & y_n \\ \vdots & \vdots & \cdots & \vdots \\ z_1 & z_2 & \cdots & z_n \end{cases} \quad (5)$$

The l th σ -video file layer can be specifically formalized as:

$$\sigma = \sum_{i,j}^t \lambda(\alpha, \beta)^2 \kappa - \tau \nu \quad (6)$$

In Equations 5 and 6, κ represents the self-attention concentration operator, and ν is the cross-attention concentration operator, which is a three-layer structure perceptron. (α, β) represents the external characteristics of the flattened display image after selecting the common standard position coding method [27].

QL means the output of the third T layer. Since then, by taking the output of the external feature QL virtual image using the shared MLP custom module, it has become a super bootstrap for each saliency pyramid structure:

$$\mu_{i,j}^t = \sum_{i,j} \theta^n (\phi_i^t + \phi_j^t) - \sigma^t \quad (7)$$

In Equation 7, μ is a feature vector matrix, and θ is a convolutional nerve prepared for a significance level n . Through the pooling converter, the sampling sample output can be realized to form a sample adaptive convolutional nerve.

It is shared among all SHM modules. For resolving the shared convolutional neuron (Sn) into different convolutional kernel groups, K pooling layers of different neuron levels are used as mapping functions in HKG:

$$\delta_{i,j}^t = \sqrt{\lambda(\alpha, \beta)^n} - \theta^n (\phi_i^t + \phi_j^t) \quad (8)$$

Algorithm code

Input: Original input [x,y,z]

Output: vector matrix μ

class DeformConv2D(nn.Module):

def __init__(self, inc, outc, kernel_size=3, padding=1, bias=None):

super(DeformConv2D, self).__init__()

self.kernel_size = kernel_size #

self.padding = padding #

self.zero_padding = nn.ZeroPad2d(padding) #

self.conv_kernel = nn.Conv2d(inc, outc, kernel_size=kernel_size,

stride=kernel_size, bias=bias)

if self.padding:

x = self.zero_padding(x)

p

p = self._get_p(offset, dtype)# (b, 2N, h, w)

p = p.contiguous().permute(0, 2, 3, 1) # (b,h,w,2N)

q_lt = Variable(p.data, requires_grad=False).floor() #floor

q_rb = q_lt + 1 #

5. The experimental analysis

To verify the external feature processing effect of the 3D image proposed in this paper, it will be tested on the public deep learning model of LINEMOD and the deep learning model that can be synthesized in the design and production of decorative patterns. The final comparison and specific analysis will be made with other least squares methods that use the

same contour. Further compare and test the actual effect of each scheme. Refer to the 3D image sampling model combined with visual guidance. The effectiveness of image processing for the external features of decorative patterns is also discussed [28].

5.1 Experimental protocol setting

In this paper, three evaluation rules are selected: external feature, ADD (3D average distance standard of model points) and reasoning analysis. Two different criteria were used in the task to verify the standardization of the proposed method and the accuracy of the data [29]. The general standard of the 2D virtual image and the 3D average straight-line distance of the 3D image are the general standard. A common criterion for a heavy virtual image can be realized by means of a 3D model for a decorative pattern and a 3D image of an appearance, a corresponding rigid transformation of the 3D image, and a transformation of the virtual image of the 3D image points onto a 2D display image. If the average linear distance of the camera pixels of all corresponding points does not exceed 5 pixels, the inference of the external feature 3D image is considered to be wrong. The method mentioned in this paper is a one-stage specific method, which does not require post-processing to obtain more accurate three-dimensional maps of external features.

Establish connection parameters with that neural network according to the deep learning model. Under the deep learning framework PyTorch, the algorithm is optimized. The strategy of the momentum stochastic gradient descent is adopted to clean up the relevant parameters. When training, the batch modification size of the deep learning model is different. Set it to 10. Add them all up. Then, conduct 6 rounds of training. The learning rate of a point is 0.01, and the learning rate decays by a factor of 1. For the deep learning model, 85% of the image data displayed in the data set is used for the deep learning model. 15% of the image database data displayed is used for the neural network model test. **Table 1** shows the hardware and software information required for the test.

Table 1. Configuration of the testing environment

Configuration	Parameters
CPU	Intel i7-13700K 16 cores 24 threads 5.4Ghz
RAM	16G DDR5
Hard disk	1T SSD
Rendering accelerator	CUDA 10.0, CUDNN 7.6.2
Deep learning framework	Py Torch 1.2.0

5.2 Comparison of the scheme

(1) GAN can be taken to synthesize deep learning models for intensive training. Combining the deep learning model with and without external features, it becomes more accurate in controlling the details of external features of pattern details compared with the original complete solution [30].

(2) The QRSTRU loss function connection proton custom module consists of two parts, followed by remapping and residual, to represent the identity mapping, with a shallower residual layer of input [31]. Partially connected inputs are used for cross-layer flow of database data. Basic formula in the whole process:

$$ST_i = W_i + Q_i, 0 \leq i \leq 1 \quad (9)$$

In Equation 9, ST_i is the connection map before the point summation, W_i is the output, and Q_i is the connection map after the summation. The connection is the concrete form of the identity map.

This operation does not cause the parameters associated with the connection to change as the number of layers increases over time, but does not increase the amount of additional computation. Loss function concatenation is the concatenation of multiple loss function blocks by means of skip concatenation according to a test solution outside the concatenation basis.

5.3 Analysis of the experimental results

(1) Accuracy comparison of general standard for duplicate virtual image

The external features of GAN (Generative Adversarial Networks) and QRSTRU before and after post-processing are listed, and the 3D graph reasoning is compared with the specific methods proposed in this paper. **Table 2** is a comparison of high precision rates under the double virtual image general standard and the ADD general standard.

Table 2. Accuracy comparison of the double virtual image

Material	Comparison of the methods		
	GAN	QRSTRU	The algorithm in this paper
Pattern texture	85.54%	88.82%	98.21%
Pattern structure	84.43%	90.26%	98.69%
Pattern color	89.59%	91.81%	99.27%

Under the general standard of reprojection, compared with the specific methods before and after GAN post-processing, the specific methods proposed in this paper have a significant improvement in accuracy.

(2) The ADD general standard accuracy comparison

Under the ADD general criteria, the results of the specific methods proposed in this paper compared with the GAN and QRSTRU specific methods without post-processing are shown in **Table 3**.

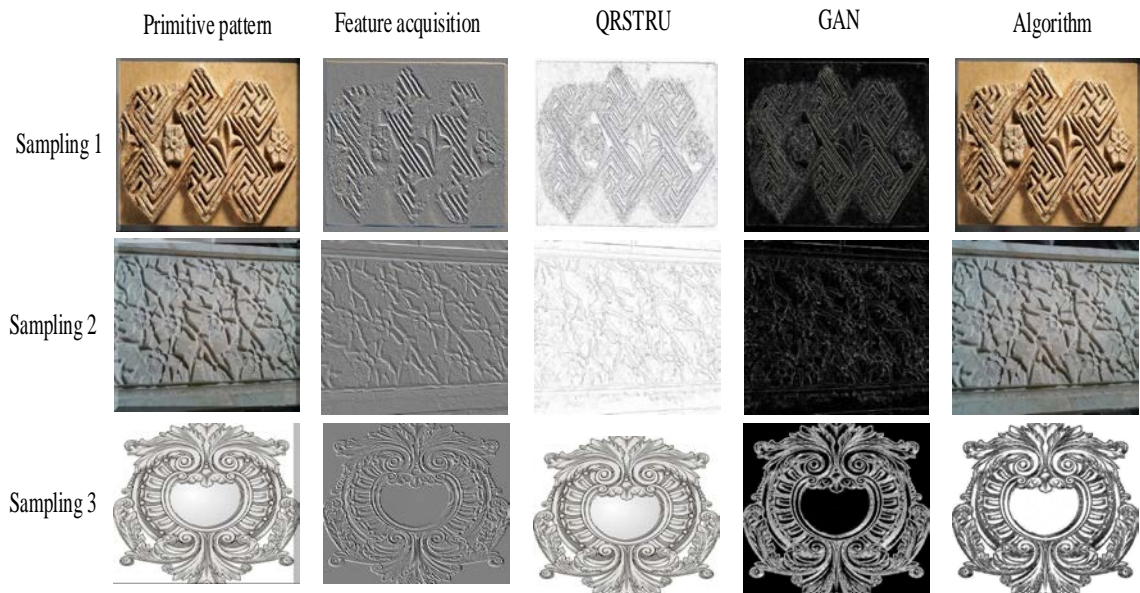
Table 3. ADD standard accuracy comparison

Material	Comparison of the methods		
	GAN	QRSTRU	The algorithm in this paper
Pattern coordinates	73.81%	82.57%	96.57%
Pattern features	82.58%	83.47%	97.07%
Pattern association	83.57%	86.67%	98.61%
Pattern calculation time/ms	0.64	0.59	0.34

The comparison results indicate that the algorithm in this paper has higher accuracy. After processing optimization, the average accuracy of the specific method in this paper is higher than that of GAN and QRSTRU, and it has higher computational efficiency, which can effectively reduce the occupancy rate of system resources. To further improve the efficiency of the least squares method, the external characteristics of the processing of three-dimensional images in this paper is the best solution.

(3) Comparison of output results

The comparison of the final output with other methods is shown in [Fig. 5](#).

**Fig. 5.** Comparative analysis of output results

The running efficiency of the method in this paper is further improved to 20 fps under the hardware and software configuration, which can meet the specific requirements of real-time automatic external feature 3D map reasoning and inference. It should take approximately 0.5ms to load the data and convert it to data. If such an effect is achieved, it is

necessary to further verify how to process the 3D images of the external features mentioned in this paper, and the effect in the multi-objective style interior decoration design scene is also verified. Intensive training and stability testing of deep 3D images will be performed on the synthetic deep learning models. The database data set contains different patterns, color senses, and processing techniques. The distribution discrete value f_1 in the database data set is shown in Fig. 6.

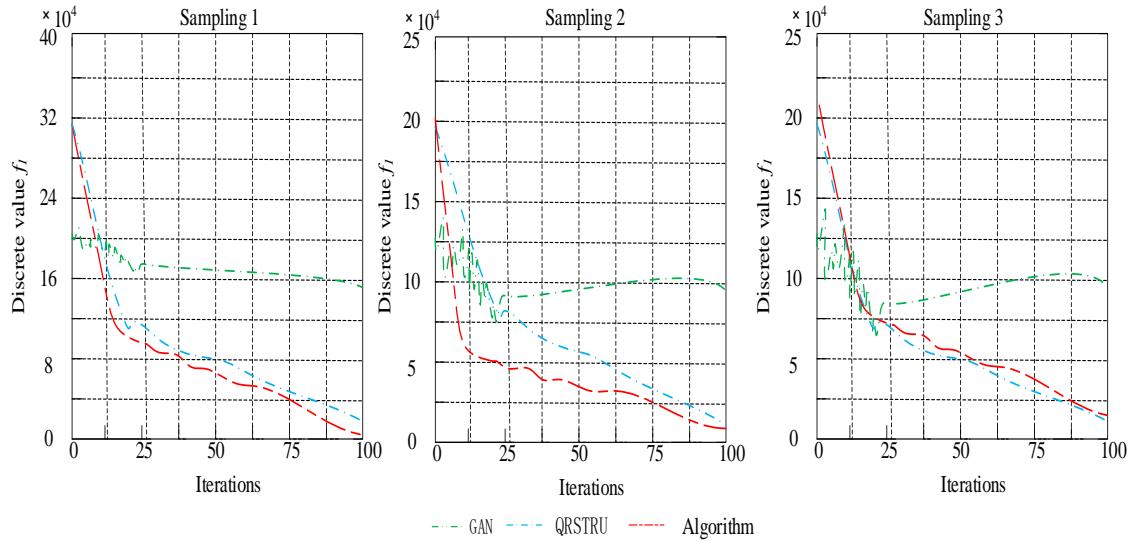


Fig. 6. Comparison of discrete effect of data set

85% of the database data in the synthetic database data set can be divided into the training set, and 15% of the database data can be divided into the training set and enter the test set.

5.4 Discussion

The convolutional neural network denoising method proposed in this paper uses prior knowledge to realize the input of multi-scale images, and introduces high-frequency residual maps to provide input images with rich edge information and clear details. Each layer (scale) sub-network line fuses multi-scale information to complete the preliminary feature fusion, realize feature enhancement, extraction and detail preservation, which is to obtain a complete and accurate residual map. Finally, it obtains a clean background. According to the characteristics of sub-networks in different layers (scales), the method uses a combined loss function to realize the convergence of network training with constraints. Compared with other fully supervised learning methods, this method has a better visual experience. This specific method shows that the noise reduction effect in the image is more detailed. A part of the details used as the background are better retained, and it is easier to obtain a very clear and clean pattern background image.

6 Conclusion

In this paper, in order to solve the problem of the region of the image displayed by texture effect, it is proposed to be able to transform the displayed image into a spatial field. In deep learning models with realism and different sizes, display images can change in quality and performance and computation. For the task of significant target software detection, the optimization of the convolutional neural network model is adopted. The design of the leveled dynamic model for significant target software detection is used to further improve the adaptive quality of samples to obtain a better image conversion effect.

In the next step, we will study the texture effect of the image display, and convert the display image into the common tasks after the intermediate digital image processing, such as target detection, semantic segmentation, computer vision technology and other final tasks. The combined display image can be transformed to cope with the rather complex digital image processing potential in the design and production of the architectural decorative patterns.

Acknowledgement

The study was supported by The Ministry of Education's Industry School Cooperation Collaborative Education Project "Reform and Research on the Course Construction of 'Basic Pattern' under the Background of New Liberal Arts"(220903711081839).

References

- [1] ZHANG M, XU S, PIAO Y, et al., "Exploring Spatial Correlation for Light Field Saliency Detection:Expansion From a Single View," *IEEE Transactions on Image Processing*, 31, 6152-6163, 2022. [Article\(CrossRefLink\)](#)
- [2] PIAO Y, RONG Z, ZHANG M, et al., "Exploit and replace: An asymmetrical two-stream architecture for versatile light field saliency detection," in *Proc. of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 11865-11873, 2020. [Article\(CrossRefLink\)](#)
- [3] Liang C, Wu Y, Zhou T, et al., "Rethinking Cross-modal Interaction from a Top-down Perspective for Referring Video Object Segmentation," 2021. [Article\(CrossRefLink\)](#)
- [4] CHENG H K, SCHWING A G., "XMem: Long-Term Video Object Segmentation with an Atkinson-Shiffrin Memory Model," in *Proc. of European Conference on Computer Vision*, 640-658, 2022. [Article\(CrossRefLink\)](#)
- [5] PANG Y, ZHAO X, ZHANG L, et al., "Multi-scale interactive network for salient object detection," in *Proc. of IEEE Conf. Comput. Vis. Pattern Recognition 2020*, 9413-9422, 2020. [Article\(CrossRefLink\)](#).
- [6] ZHANG M, FEI S X, LIU J, et al., "Asymmetric Two-Stream Architecture for Accurate RGB-D Saliency Detection," in *Proc. of Eur. Conf. Comput. Vis.*, pp. 374-390, 2020. [Article\(CrossRefLink\)](#).
- [7] WEI J, WANG S, HUANG Q., "F3net: Fusion, feedback and focus for salient object detection," in *Proc. of AAAI*, 12321-12328, 2020. [Article\(CrossRefLink\)](#)
- [8] CARION N, MASSA F, SYNNAEVE G, et al., "End-to-end object detection with transformers," in *Proc. of Eur. Conf. Comput. Vis.*, 213-229, 2020. [Article\(CrossRefLink\)](#)
- [9] WEI J, WANG S, WU Z, et al., "Label decoupling framework for salient object detection," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13022-13031, 2020. [Article\(CrossRefLink\)](#)

- [10] LIU J J, HOU Q, CHENG M M, "Dynamic feature integration for simultaneous detection of salient object, edge, and skeleton," *IEEE Trans. Image Process.*, 29, 8652-8667, 2020. [Article\(CrossRefLink\)](#)
- [11] Zhou H, Chen P, Yang L, et al., "Activation to Saliency: Forming High-Quality Labels for Completely Unsupervised Saliency Object Detection," 2021. [Article\(CrossRefLink\)](#)
- [12] WU Z, SU L, HUANG Q., "Decomposition and Completion Network for Saliency Object Detection," *IEEE Transactions on Image Processing*, 30, 6226-6239, 2021. [Article\(CrossRefLink\)](#)
- [13] LIU T, YUAN Z, SUN J, et al., "Learning to detect a salient object," *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(2), 353-367, 2011. [Article\(CrossRefLink\)](#)
- [14] LIU Z, WANG Y, TU Z, et al., "Tri Trans Net: RGB-D Saliency Object Detection with a Triplet Transformer Embedding Network," in *Proc. of ACM Int. Conf. Multimedia*, 4481-4490, 2021. [Article\(CrossRefLink\)](#)
- [15] LIU N, ZHANG N, WAN K, et al., "Visual saliency transformer," in *Proc. of Int. Conf. Comput. Vis.*, 4722-4732, 2021. [Article\(CrossRefLink\)](#)
- [16] Radzikowski Kacper, Wang Le, Yoshie Osamu, Nowak Robert, "Accent modification for speech recognition of non-native speakers using neural style transfer," *EURASIP Journal on Audio, Speech, and Music Processing*, 2021(1), 2021. [ArticleCross\(RefLink\)](#)
- [17] Wan Xiang, Zhang Xiangyu, Liu Lilan, et al., "An Improved VGG19 Transfer Learning Strip Steel Surface Defect Recognition Deep Neural Network Based on Few Samples and Imbalanced Datasets," *Applied Sciences*, 11(6), 2021. [ArticleCross\(RefLink\)](#)
- [18] Ziqi Lu, "Digital Image Art Style Transfer Algorithm and Simulation Based on Deep Learning Model," *Scientific Programming*, vol. 2022, Article ID 8409459, 9 pages, 2022. [ArticleCross\(RefLink\)](#)
- [19] CHEN Q, LIU Z, ZHANG Y, et al., "RGB-D saliency object detection via 3D convolutional neural networks," in *Proc. of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 2, 1063-1071, 2021. [Article\(CrossRefLink\)](#)
- [20] ZHANG J, FAN D P, DAI Y, et al., "RGB-D saliency detection via cascaded mutual information minimization," in *Proc. of the IEEE/CVF International Conference on Computer Vision*, 4338-4347, 2021. [Article\(CrossRefLink\)](#)
- [21] Chen T, Hu X, Xiao J, et al., "BPFNet: Boundary-aware progressive feature integration network for saliency object detection," *Neurocomputing*, 451(8), 152-166, 2021. [Article\(CrossRefLink\)](#)
- [22] Yeh M C, Hsu C F, Lu C J., "Fast saliency object detection through efficient subwindow search," *Pattern Recognition Letters*, 46(sep.1), 60-66, 2014. [Article\(CrossRefLink\)](#)
- [23] LI A, ZHANG J, LV Y, et al., "Uncertainty-aware joint saliency object and camouflaged object detection," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10071-10081, 2021. [Article\(CrossRefLink\)](#)
- [24] Amini A, Periyasamy A S, Behnke S., "YOLO Pose: Transformer-based Multi-Object 6D Pose Estimation using Keypoint Regression," 2022. [Article\(CrossRefLink\)](#)
- [25] Zoph B, Cubuk E D, Ghiasi G, et al., "Learning data augmentation strategies for object detection," in *Proc. of European conference on computer vision*, 566-583, 2020. [Article\(CrossRefLink\)](#)
- [26] Ge Z, Liu S, Wang F, et al., "YOLOX: Exceeding YOLO Series in 2021," *arXiv preprint arXiv:2107.08430*, 2021. [Article\(CrossRefLink\)](#)
- [27] Iwata T, Ghahramani Z., "Improving Output Uncertainty Estimation and Generalization in Deep Learning via Neural Network Gaussian Processes," *arXiv e-prints*, 2017. [Article\(CrossRefLink\)](#)
- [28] Wang X, Chen J, Jiang K, et al., "Single Image De-raining Via Clique Recursive Feedback Mechanism," *Neurocomputing*, vol. 417, pp. 142-154, 2020. [Article\(CrossRefLink\)](#)
- [29] Jin X, Chen Z, Li W., "AI-GAN: Asynchronous Interactive Generative Adversarial Network for Single Image Rain Removal," *Pattern Recognition*, 100, 107143, 2019. [Article\(CrossRefLink\)](#)
- [30] Wei Y, Zhang Z, Wang Y, et al., "Semi-Derainage: A New Semi-Supervised Single Image Deraining Network," in *Proc. of the IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, 1-9, 2021. [Article\(CrossRefLink\)](#)

- [31] Su Z, Zhang Y, Zhang X P, et al., “Non-local Channel Aggregation Network for Single Image Rain Removal,” *Neurocomputing*, 469(Jan.16), 261–272, 2021. [Article\(CrossRefLink\)](#)



Jingjing Ye, whose birthplace is in Yugan, Jiangxi province, P.R. China, was born in 1981. She received her Master’s degree from Jiangnan University, P.R. China. Now, she works in the Department of the Architecture and Design College, at Nanchang University. Her research interests mainly include art and design. Furthermore, she is also interested in the computer sciences.



Jun Wang is a man who has a Doctor of Philosophy. He works at the Fashion Institute, Donghua University in China. He is now an associate professor. His main research direction is modern design art theory. He is also an expert in computers and has made some progress in them.