Multimedia Tools and Applications (2024) 83:28125–28145
https://doi.org/10.1007/s11042-023-16579-0

HyNet: A novel hybrid deep learning approach for efficient interior design texture retrieval

Junming Chen[1] · Zichun Shao[1] · Caichun Cen[1] · Jiaqi Li[1]

Received: 3 April 2023 / Revised: 19 July 2023 / Accepted: 18 August 2023 /
Published online: 31 August 2023

Abstract
Interior designers are suffering from a lack of intelligent design methods. This study aims to enhance the accuracy and efficiency of retrieval textures for interior design, which is a crucial step toward intelligent design. Currently, interior designers rely on repetitive tasks to obtain textures from websites, which is ineffective as a interior design often requires hundreds of textures. To address this issue, this study proposes a hybrid deep learning approach, HyNet, which boosts retrieval efficiency by recommending similar textures instead of blindly searching. Additionally, a new indoor texture dataset is created to support the application of artificial intelligence in this field. The results demonstrate that the proposed method's ten recommended images achieve a high accuracy rate of 91.41%. This is a significant improvement in efficiency, which can facilitate the design industry's progression towards intelligence. Overall, this study offers a promising solution to the challenges facing interior designers, and it has the potential to significantly enhance the industry's productivity and innovation.

Keywords Interior design · Texture retrieval · Hybrid neural networks · Hand-crafted features · Deep learned features · Evaluation protocol

1 Introduction

Texture retrieval [1–5] is a significant challenge in the interior design process, as it still heavily relies on designers with extensive work experience in the industry. Designers often spend a considerable amount of time retrieving textures for their designs, and an automatic interior design texture retrieval system can effectively enhance the efficiency of texture retrieval, which in turn improves design efficiency and benefits society. The prevalence of artificial intelligence (AI) applications in various fields [1–3, 6–9], necessitates the use of AI methods

✉ Jiaqi Li
jiqli@must.edu.mo

Junming Chen
jmchen@must.edu.mo

[1] Faculty of Humanities and Arts, Macau University of Science and Technology, Avenida WaiLong, 999078 Taipa, Macau, China

to speed up the interior design process. For instance, extensive research has been conducted on person retrieval [10–13] and vehicle retrieval [14–22], and the application of AI has facilitated substantial growth in these industries. However, the interior texture retrieval community has been slow to progress due to a lack of adequate datasets and benchmarks. Therefore, the development of suitable datasets and comprehensive benchmarks is critical for the growth of the interior texture retrieval community.

Image retrieval [11, 15, 23–25] has advanced significantly in a variety of applications, including person retrieval [10–13, 26], vehicle retrieval [15–18, 20, 20–22], and furniture retrieval [27, 28]. For example, the "Market-1501" [10] dataset for person retrieval, the "VeRi-776" [22] dataset for vehicle retrieval, and the "DeepFurniture" [27] dataset for furniture retrieval. These datasets have significantly impacted the growth of the related areas. However, to our knowledge, few datasets suit interior texture retrieval, which stymies the advancement of deep learning in this sector.

Hand-crafted features and deep learned features are commonly used in image retrieval systems. Hand-crafted features can be divided into global features and local features [29, 30]. Common global features include color features [31, 32], texture features [23, 33], and edge features [34]. Common local features include Bag of Words (BoW) [35], Scale-invariant Feature Transform (SIFT) [36, 37], Speeded-up Robust Features (SURF) [31, 38], Local Binary Pattern (LBP) [15, 39], and Histogram of Oriented Gradients (HOG) [38, 40]. For example, Lowe [36] adopted SIFT features to extract image keypoints for image retrieval. Yan et al. [37] applied HOG features to person retrieval. Wang et al. [29] applied LBP features to vehicle retrieval. However, hand-crafted features are usually selected and designed according to a specific task, which requires the accumulated knowledge of professionals and a long time to design a reasonable feature extraction method [41].

Deep learned features are commonly employed in image retrieval to address the dependence of hand-crafted features on professionals. Deep learned features are usually extracted by Convolutional Neural Networks (CNNs), and the feature extraction methods include AlexNet [42], VGG [43], and ResNet [44]. For example, Yuan and Zhang [45] proposed using AlexNet for landscape and architecture retrieval. Ha et al. [46] proposed using VGG for interior space retrieval. Ayyachamy et al. [25] proposed using ResNet for medical image retrieval. Since the deep learning-based methods can automatically and effectively learn features from images, many studies have shown that using deep learning-based methods, CNNs can autonomously learn low-level to high-level features from a large number of images, improving image classification and retrieval close to human-level [47]. For example, HSGM [48] proposes a hierarchical similarity graph module to solve the conflict of backbone networks and mine discriminative features. However, the deep learning-based methods tend to ignore features such as texture features and edge features [49].

As shown in Fig. 1, we propose a comprehensive approach for improving interior texture retrieval performance. Firstly, a robust dataset called "Interior-134" is constructed, consisting of 26K textures categorized into 15 major categories and 134 subcategories. The dataset is well-annotated and publicly accessible, which can serve as a valuable resource for the development of image retrieval applications in the interior design sector. Secondly, the paper establishes a benchmark for both hand-crafted and deep learned feature methods on the "Interior-134" dataset. The proposed hybrid deep learning method, called HyNet, is introduced. HyNet combines the advantages of hand-crafted features and deep learned features, where the hand-crafted feature retrieval is achieved by enhancing VGG [43] with HOG [40]. Finally, the experimental results show that HyNet outperforms the best deep learned feature method, VGG, by 1.26% and 4.53% on Rank1 and mAP, respectively. Overall, this study

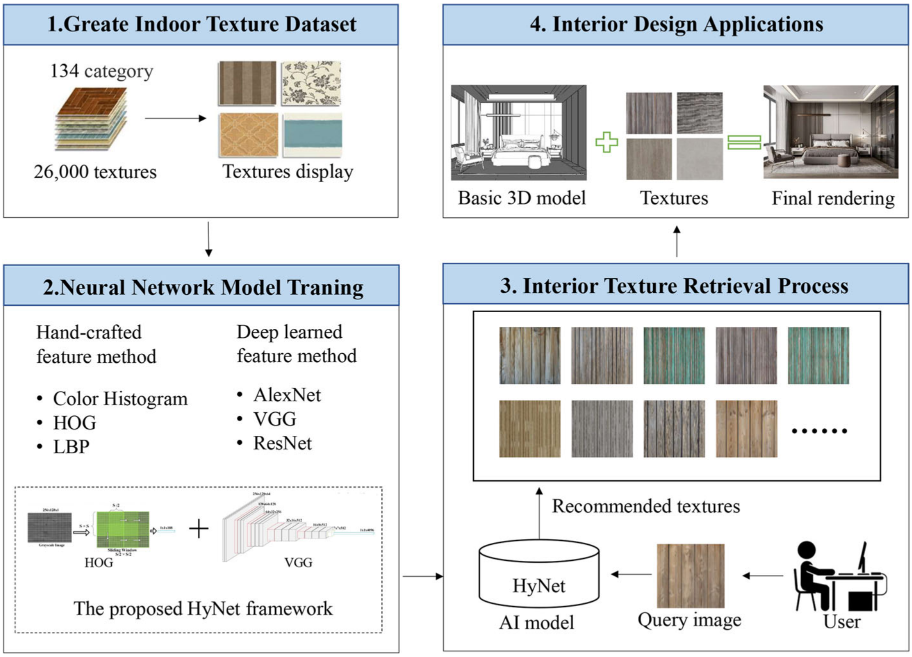

1.Greate Indoor Texture Dataset
134 category
26,000 textures
Textures display
4. Interior Design Applications
Basic 3D model
Textures
Final rendering
2.Neural Network Model Traning
Hand-crafted feature method
• Color Histogram
• HOG
• LBP
Deep learned feature method
• AlexNet
• VGG
• ResNet
HOG
VGG
The proposed HyNet framework
3. Interior Texture Retrieval Process
Recommended textures
HyNet
AI model
Query image
User

Fig. 1 Block diagram of the proposed research. This study first established an indoor texture dataset for training AI to enhance the ability of AI to retrieve textures and then improved the retrieval algorithm. Therefore, users can easily retrieve other suitable textures through one texture, speeding up design efficiency

provides a promising solution for interior texture retrieval and contributes to the development of more efficient and accurate image retrieval applications in the interior design sector.

The main contributions of this paper are as follows:

1. We propose the HyNet approach which optimally fuses features learned from both RGB and grayscale images to improve interior texture retrieval accuracy.
2. We introduce the “Interior-134” dataset, which is the first publicly accessible dataset in interior texture retrieval and is labeled to support further research in this field.
3. We establishe a more comprehensive benchmark by comparing hand-crafted features and deep learned features.
4. Extensive experiments on the “Interior-134” dataset demonstrate that the proposed method outperforms other state-of-the-art methods in terms of retrieval accuracy.

The remainder of the study is arranged as follows: Section 2 introduces the related work. Section 3 covers the dataset and methods. Section 4 shows the experimental results and analysis. Section 5 addresses the limitations of this study and concludes this paper.

2 Related work

2.1 Application of hand-crafted features methods in image retrieval

Common hand-crafted feature methods include HOG [40], LBP [39], and SIFT [36]. HOG [40] establishes feature descriptors by setting different detection window sizes and calculating the gradient direction [15, 38, 40], which can be used for person retrieval. After scaling the same image to obtain a large number of images, SIFT [36] compares the differences between these images to determine the key points and then calculates the gradient histogram around the key points before generating a vector with unique features, which can be used for image retrieval [36, 37, 48]. LBP [39] is an efficient nonparametric texture description method that obtains the local features of an image by describing the grayscale relationship between each pixel. Since it describes the relationship between pixels and the grayscale spatial distribution of the surrounding pixels, it is not affected by light, and its improved version has rotational invariance [39, 50]. Wang et al. [15] applied LBP in vehicle locating. In addition, Shen et al. [23] proposed a large benchmark for fabric image retrieval, including hand-crafted feature methods and deep learned feature methods.

2.2 Application of deep learned features in image retrieval

AlexNet [42], VGG [43], and ResNet [44] are examples of common deep learned feature methods. AlexNet [42] is a deep neural network with five Convolutional (Conv) Layers and three Fully Connected (FC) Layers. First applied to image classification and retrieval, AlexNet [42] greatly surpassed hand-crafted feature methods in terms of performance. The modular 3 × 3 network structure of VGG [43] makes modular combinatorial networks possible while expanding the neural network hierarchy. ResNet [44] provides a residual structure that overcomes the neural network degradation problem as the number of network layers grows, enabling deeper neural network layers and improved performance. EMRN [20] proposes a multi-resolution feature dimension uniform module to fix dimensional features from images of varying resolutions. HPGN [18] adopts a backbone network and a pyramidal graph network for vehicle retrieval.

2.3 Image retrieval related datasets

The development of image retrieval relies on the relevant datasets. In terms of person retrieval, Tsinghua University constructed the “Market-1501” [10] dataset, which includes 1,501 persons captured using six cameras, averaging eighteen training images per person. Beijing University of Posts and Telecommunications acquired and made available the “VeRi-776” [22] vehicle dataset, which comprises 776 cars with over 50k images shot by twenty cameras across a square kilometer in twenty-four hours. Liu et al. [27] constructed a freely available “DeepFurniture” dataset, which comprises 24k interior images and 170k furniture instances. With their construction and public availability, these datasets greatly assisted studies in their respective domains. However, the absence of suitable datasets and benchmarks has hindered research in the interior texture retrieval community. Hence, we aim to establish an interior texture dataset as well as its benchmark.

3 Datasets and methods

3.1 Datasets Description

Appropriate interior texture resources with the color, texture, and resolution satisfying user requirements are scarce. In this connection, a new fine-grained interior texture dataset, Interior-134, with a two-level category structure is established in this paper. First, high-resolution textures are collected from the Internet and manually screened and annotated by professional designers. The screening process involves removing textures with ambiguous semantic expressions and verifying the annotations of the remaining textures. The basis for the secondary category texture classification is the subdivision of textures by professional design texture websites. The processed dataset comprises 26k interior textures, with 15 primary categories and 134 secondary categories, as shown in Fig. 2. One primary category, such as Stone, has twenty secondary category directories. However, the number of secondary category directories may vary for other primary categories. For example, the Leather primary category has just two secondary category directories. Other than that, some primary categories include thousands of textures, such as Stone, Decorative Painting, and Natural Ground, while other categories, such as Leather and Mosaic, have only tens or hundreds of textures.

This dataset is suitable for fine-grained interior texture retrieval in that its included textures are divided into primary and secondary categories. Since the textures in different secondary categories are similar, making neural networks understand why seemingly similar textures are classified into different categories is one of the challenges in the study. The similarity of secondary textures under different primary categories can be observed from a selection of secondary directory textures shown in Figure 3. For example, the Woven Wood is similar to the Plaid Fabric Pattern, and the European Classic Style Wallpaper is similar to the Patterned Fabric Pattern. In the meantime, many secondary category textures in the same primary category also share some similarities. For example, the Wooden Flooring is similar to the Woodgrain Panels, and the Plaid Fabric Pattern is similar to the Striped Fabric Pattern. Such is the challenge of the texture retrieval task.

In addition, Fig. 4 shows the texture distribution in all secondary categories, where the number of textures in each secondary category is also unbalanced. The number of textures in the largest secondary category of the dataset exceeds 2k, while that of the smallest secondary category is only three. The uneven number of textures in different secondary directories and the lack of data in some directories are both research challenges.

3.2 Datasets division

The dataset is divided into training, validation, and test sets. First, each secondary category is split evenly into total training and test sets, and those that cannot be divided evenly are also separated into training sets. The total training sets are then divided into training sets

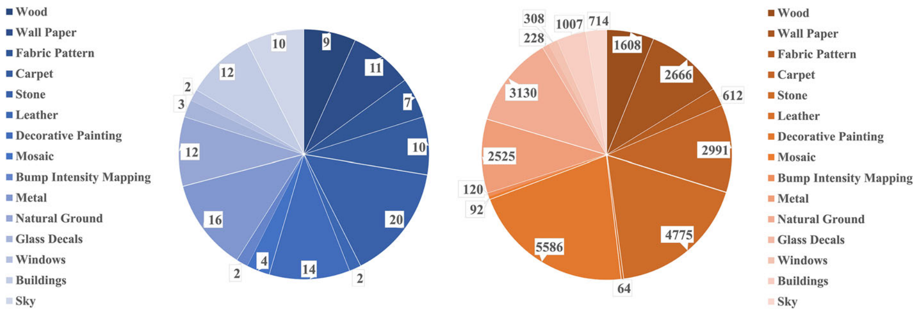


Fig. 2 Distribution of categories and quantities in Interior-134. The established dataset has 134 categories and more than 24k textures

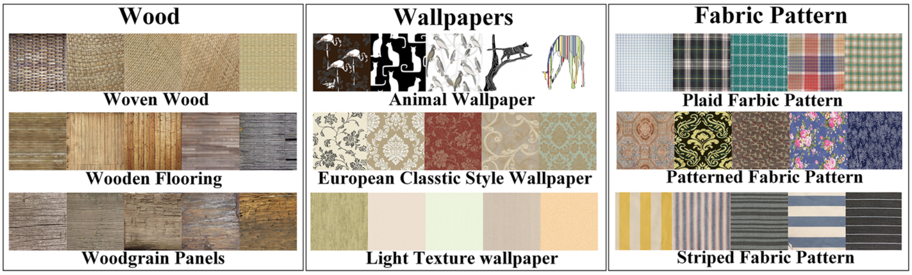


Fig. 3 Partial texture display of the Interior-134 dataset

and validation sets, with the first of every ten textures in each secondary category going into validation sets and the remainder going into training sets. We further divide the test sets into query and gallery sets in the same way as above. The final overview of the dataset is shown in Table 1.

3.3 Evaluation protocol

Cumulative Matching Characteristics (CMC) [42] and mean Average Precision (mAP) [10] are used to comprehensively evaluate the retrieval method performance in this paper. CMC indicators focus on retrieving target category images within a certain number of times. The commonly used measurement standards are Rank 1, Rank 5, Rank 10, and Rank 20. Rank 1 indicates the probability that the first retrieved image is in the target category. Rank 5 indicates the probability that more than one of the first five retrieved images is in the target category, and Rank 10 and Rank 20 may be deduced by analogy [42]. The mAP metric measures the ease with which all target category images are retrieved, and a higher mAP indicates a better retrieval. Since this method considers accuracy and recall, it can provide a more comprehensive evaluation [10].

The CMC is given by using (2):

$$AccK = \begin{cases} 1 = \text{The first K query results have the same category.} \\ 0 = \text{The first K search results do not have the same category.} \end{cases} \quad (1)$$

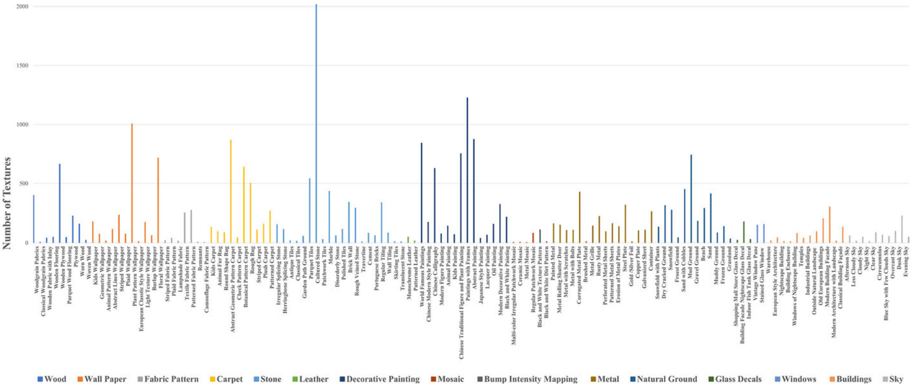


Fig. 4 Distribution of secondary label's number in Interior-134

Table 1 Splitting of interior-134

Training Set	Validation Set	Test Set	
		Gallery	Query
10,606	1213	13,001	1478

$$CMC = \frac{\sum_{1}^{N_q} AccK}{N_q} \tag{2}$$

where

N_q Total number of query

The mAP is given by using (6):

$$precision = \frac{TP}{TP + FP} \tag{3}$$

$$recall = \frac{TP}{TP + FN} \tag{4}$$

$$AP = \frac{1}{N_t} \sum_{i} precision_i, i \in \Omega \tag{5}$$

$$mAP = \frac{\sum_{i=1}^{N_q} AP_i}{N_q} \tag{6}$$

where

FN False Negative
FP False Positive
TN True Negative
TP True Positive
N_t Number of all images of the same category as the query image
Ω The set of the kth query result of the same category as query
N_q Total number of query

As shown in Fig. 5, if the desired image has two correct values, the first five search results, as an example, can be selected for analysis:

- **(a)** The retrieval of the first and fourth images is correct because the correct result is obtained for the first image. Thus, Rank 1 and Rank 5 in CMC are both 1, and the AP value is
$$\frac{1 \div \frac{2}{4}}{2} = 0.75.$$
- **(b)** Retrieving the first and second images is correct, Rank 1 and Rank 5 in CMC are both 1, and the AP value is also 1.
- **(c)** Retrieving the first and fifth images is correct, Rank 1 and Rank 5 in CMC are both 1, and the AP value is
$$\frac{1 \div \frac{2}{5}}{2} = 0.7.$$

Therefore, CMC only considers the position order of the first correct image retrieval, while mAP is concerned with the efficiency of finding all the correct category images.

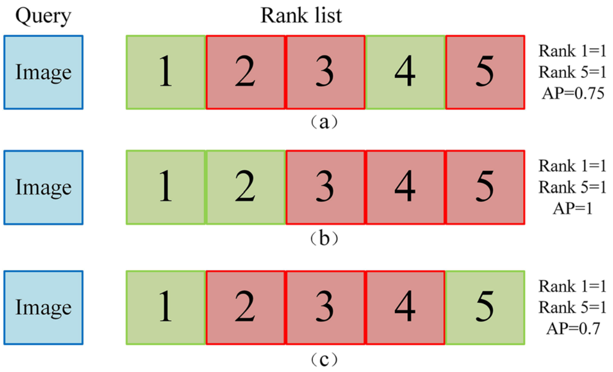


Fig. 5 Comparison of CMC and AP indicators

3.4 Method

Based on the above research, we propose a hybrid network fusion of hand-crafted features and deep learned features named HyNet. HyNet extracts hand-crafted features via HOG [40] and deep learned features via VGG [43] before fusing them in the back-end.

3.4.1 HOG features extraction

In terms of the selection of the hand-crafted features extraction method, this paper selects the HOG [40] method. The HOG [40] method was first proposed in 2005 and works by creating a histogram of the gradient direction distribution in an image and accumulating these gradient sizes, and the accumulated results are normalized to extract features. The idea of the HOG [40] method is that stronger gradients contribute more to the size of their respective angular histograms while minimizing the effect of weak and randomly oriented gradients caused by noise. The HOG [40] features extraction steps are displayed in Table 2.

We selected two textures, one from Woven Wood and the other from Plant Pattern Wallpaper, and employed the Histogram of Oriented Gradients (HOG) method to extract their features. The visualized results of these features are shown in Fig. 6.

Upon examining the images, we can observe that the HOG features effectively highlight the underlying patterns within the textures. For instance, in the first texture from Woven Wood, the visualization of HOG features a prominent horizontal texture direction. On the other hand, in the second texture, the visualization of HOG features reveals a dominant

Table 2 HOG features extraction steps

Step 1. Set a detection window covering the entire image.

Step 2. Calculate the gradient size and direction of each pixel in the detection window.

Step 3. Set $M \times N$ size cells.

Step 4. Calculate the gradient corresponding to each cell and divide it in the corresponding box.

Step 5. Adjacent cells are combined into blocks.

Step 6. Standardize on each block.

Step 7. All normalized bars are combined into HOG feature descriptors.

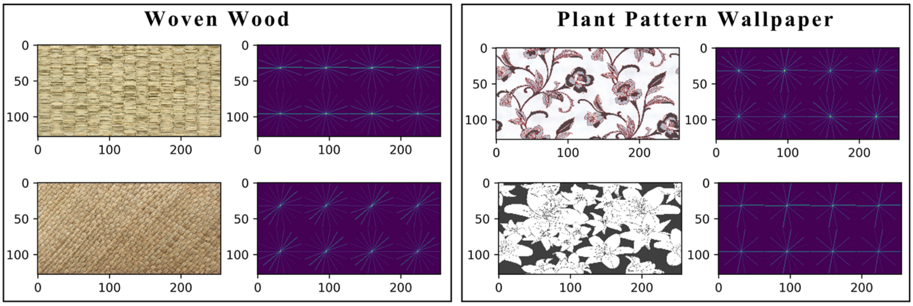


Fig. 6 Visual display of texture features extracted by the HOG method

diagonal texture direction. As for the textures in the Plant Pattern Wallpaper, where no apparent texture direction is evident, the HOG method extracts diverse feature intensities for each feature block, further showcasing the HOG method's capability to capture intricate texture features.

3.4.2 VGG features extraction

VGG [43], the first-place winner in the 2014 ImageNet challenge, is selected for deep learned feature extraction. VGG [43] proved that increasing the network depth affected its final performance, arguing that the VGG network using the 3 × 3 convolution instead of the larger convolutional kernels (11 × 11, 7 × 7, 5 × 5) in AlexNet [42] can achieve better results. By enabling VGG [43] parameter reduction while increasing the network depth, this method ensures that more complex models are learned. VGG [43] also provides the concept of modularity, i.e., combining modules to create neural networks, thus making neural network creation easier. The high-performing VGG [43] commonly comprises 16 layers, including some pooling layers between many Conv Layers to compress the image size while increasing the number of channels to prevent information loss. The final three layers are FC Layers with dimensions of 4096, 4096, and 1000. A Rectified Linear Unit (RELU) function is also added to each hidden layer to increase nonlinearity and accelerate the convergence speed. The RELU function is widely used in subsequent research. Table 3 shows the VGG [43] configurations.

3.4.3 HyNet

Numerous studies have proven that fusing the features extracted by different methods, for example, fusing hand-crafted features such as color and texture, can achieve better performance, and Bag of Words (BoW) can be used to enhance accuracy [31, 32, 34, 38]. Another example is the fusion of deep learned features and hand-crafted features in person retrieval [11, 33, 51].

In this paper, we also adopted a fusion model to increase the accuracy of texture retrieval. In hand-crafted features, we first evaluated the performance of HOG [40], LBP [39], and SIFT [36], and the results revealed that HOG [40] performed the best. In deep learned features, we evaluated AlexNet [42], VGG [43], and ResNet [44], and the results revealed that VGG [43] performed the best. Therefore, VGG [43] and HOG [40] are selected for back-end fusion in this study, and Fig. 7 shows the final network configurations.

Specifically, HyNet is built by the parallel configurations of VGG [43] and the HOG [40]. For VGG [43], we compress RGB textures of different sizes to a resolution of 256 × 128 × 3

Table 3 VGG network configurations

Name	Input channel	Input dimension	Output channel	Output dimension	Subwindows
Original image	3	256 × 128	3	256 × 128	–
Conv	3	256 × 128	64	256 × 128	3 × 3
Conv	64	256 × 128	64	256 × 128	3 × 3
Maxpool	64	256 × 128	64	128 × 64	2 × 2
Conv	64	128 × 64	128	128 × 64	3 × 3
Conv	128	128 × 64	128	128 × 64	3 × 3
Maxpool	128	128 × 64	128	64 × 32	2 × 2
Conv	128	64 × 32	256	64 × 32	3 × 3
Conv	256	64 × 32	256	64 × 32	3 × 3
Conv	256	64 × 32	256	64 × 32	1 × 1
Maxpool	256	64 × 32	256	32 × 16	2 × 2
Conv	256	32 × 16	512	32 × 16	3 × 3
Conv	512	32 × 16	512	32 × 16	3 × 3
Conv	512	32 × 16	512	32 × 16	1 × 1
Maxpool	512	32 × 16	512	16 × 8	2 × 2
Conv	512	16 × 8	512	16 × 8	3 × 3
Conv	512	16 × 8	512	16 × 8	3 × 3
Conv	512	16 × 8	512	16 × 8	1 × 1
Adaptivepool	512	16 × 8	512	7 × 7	2 × 2
FC	512	7 × 7	4096	1 × 1	–

The Convolutional Layer is abbreviated as “Conv.” The Fully Connected Layer is abbreviated as “FC.” The ReLU activation function is not shown in this table

Table 4 Comparing the difference in feature concat before or after fully connected layers

Methods	Features dimension	mAP(%)	Rank 1	Rank 10	Rank 20
Before FC-Flatten	25196	25.95	69.12	90.37	94.12
Before FC-GAP	620	25.41	67.86	89.45	94.01
After FC: HyNet	**4204**	**27.16**	**72.87**	**91.41**	**94.79**

Bold font indicates that the result is the most important conclusion in this table

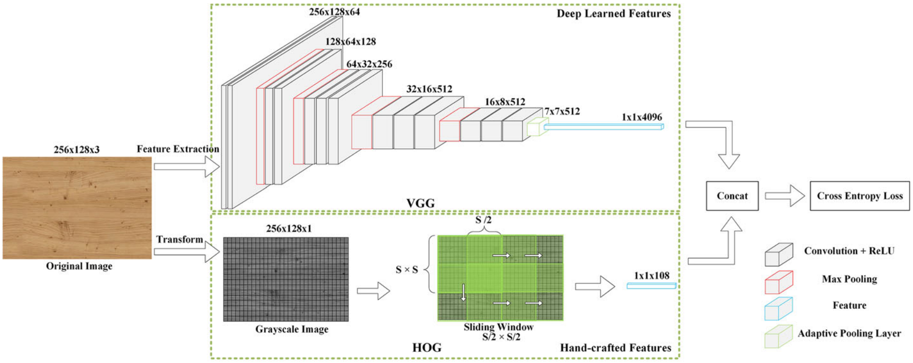


Fig. 7 Our Proposed HyNet framework. VGG and HOG for features fusion in the back-end

($Height \times Width \times Channel$) and input them into the VGG [43] network for feature extraction. First, the texture passes through the first and second Conv layers, where the filter in this convolution is 3 × 3. The texture dimensions change from 256 × 128 × 3 to 256 × 128 × 64. Immediately after that, the texture passes through a pooling layer with a filter size of 3 × 3 and a stride of two, where the resulting texture dimensions will be converted from 256 × 128 × 64 to 128 × 64 × 128. The next third to thirteenth layers constitute a process similar to that described earlier, where the output texture will be reduced to 16 × 8 × 512.

There was originally a pooling layer between the thirteenth and fourteenth layers. The size of our input texture is not the 224 × 224 × 3 (Height × Width × Channel) resolution of the default VGG [43]. To make the final output feature dimension consistent with the default VGG [43], we replaced the default last pooling layer with an adaptive pooling layer so that the output dimension can be fixed at 7 × 7 × 512.

The fourteenth FC layer is first flattened to 25088-dimensional features and then mapped to 4096-dimensional features, i.e., the fused features. For HOG [40], we convert the color texture into a grayscale texture with a resolution of 256 × 128 × 3 and input it into HOG [40]. The block size of 128 × 128 is selected for HOG [40]. Using a sliding window with a cell size of 64 × 64 for features extraction, the 108-dimensional features are obtained. Finally, we fuse the 4096-dimensional features extracted by VGG [43] with the 108-dimensional features extracted by HOG [40] to form the proposed HyNet.

There are two sample ways to fuse before FC. Before FC-Flatten means using flattened features for direct fusion, and Before FC-GAP means using flattened and performing global average pooling (GAP) for fusion. The results obtained by Before FC-Flatten are better than Before FC-GAP, but the dimension is too large. Our method (Hynet) is more modest in terms of dimensionality. More importantly, HyNet outperforms Before FC-Flatten and Before FC-GAP by 1.21% and 1.75% on mAP metrics, respectively. Therefore, we choose to fuse after FC (Table 4).

HyNet adopts Cross-Entropy as the loss function. During model training, the loss function adjusts the weights corresponding to each neuron of the neural network to create a machine learning model with better performance. Thus, the loss function is important. The cross-entropy loss function is usually used in classification tasks. The neural network training first performs forward propagation and outputs the probabilities of the classification categories. The loss function evaluates how these outputs differ from the correct classification to calculate the error. Then, the error is calculated and propagated backward to adjust the value of each neuron. The process will continue to loop until the error gradually decreases to finally produce a model with a low error. The standard categorical Cross-Entropy loss function [52] is expressed with (7):

$$J_{ce} = -\frac{1}{M} \sum_{k=1}^{K} \sum_{m=1}^{M} y_m^k \times \log\left(h_\theta\left(x_m, k\right)\right) \tag{7}$$

where

M	Number of training examples
K	Number of classes
y_m^k	Target label for training example m for class k
x	Input for training example m
h_θ	Model with neural network weights θ

4 Experimental results and analysis

4.1 Implementation details

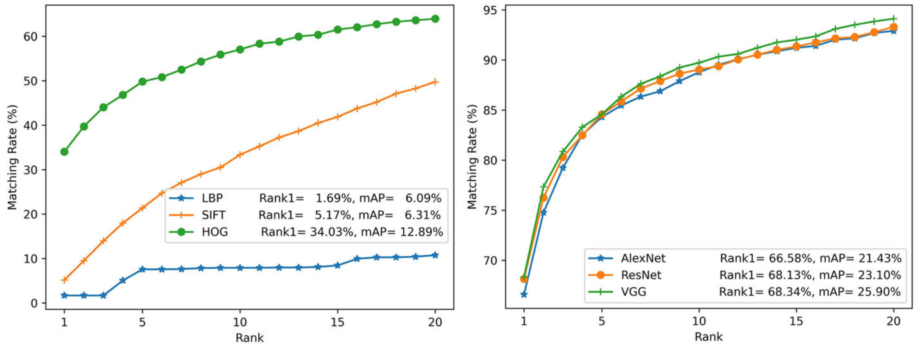
The proposed network is implemented in PyTorch on a platform running Windows 10 with 48 GB of RAM and an RTX5000 GPU, which has 16 GB of video memory. With multiprocess computation, the proposed HyNet took 10 hours to train the model, and the Average feature extraction time (AFE) is 13 microseconds per image. The benchmark of hand-crafted feature methods and deep learned feature methods is established. In addition, the performance improvement by fusing hand-crafted feature methods and deep learned feature methods in the back end is evaluated. The following parameters are used in the comparison process. The batch size is set to 64, the epoch is set to 60, the dropout is set to 0.5, the random horizontal flip is set to 0.5, the learning rate is set to 0.05 and programmed to decay once the number of iterations proceeds to 2/3 to 1/10 of its initial value. Cross-Entropy is selected as the loss function, and the optimizer adopts Stochastic Gradient Descent (SGD) and scales all input textures to a resolution of $256 \times 128 \times 3$ (Height × Width × Channel). Note that post-processing such as re-ranking [12, 13] and multi-query fusion [53] is not conducted.

4.2 Comparison with state-of-the-art methods

The experiment was carried out under the secondary classification because it is more difficult than the primary classification. In terms of hand-crafted feature methods, we evaluated the performance of LBP [39], SIFT [36], and HOG [40]. As displayed in Figure 8, HOG [40], presented as the green line, outperformed LBP [39] and SIFT [36]. HOG [40] has the highest Rank 1 (i.e., 34.03%) among the three hand-crafted feature methods, which keeps mounting up gently to 63.60% in the following Rank 1 to Rank 20 section, reaching a final mAP of 12.89%. LBP [39] and SIFT [36] are presented as blue and orange lines, respectively. Initially, the CMC of these two methods is low. Then, the CMC of SIFT [36] surged from Rank 1 (i.e., 5.17%) to Rank 20 (i.e., 49.76%) while that of LBP [39] was lower at Rank 1 (i.e., 1.69%). The matching rate growth of LBP [39] was rather slight, finally reaching 10.75% at Rank 20. The final mAP of LBP [39] and SIFT [36] is 6.09% and 6.31%, respectively. In conclusion, HOG [40] has the highest CMC and mAP all along, making it the best-performing hand-crafted feature method among these three.

The reason for the better performance of HOG [40] is the texture dataset nature of Interior-134, where the textures are repeated and distributed. The design idea of LBP [39] is to extract the image edge contour, and the design idea of SIFT [36] is to extract the key points of the image. These two methods are not well suited for interior texture feature extraction, while HOG [40] extracting the gradient features and the corresponding intensity is more suitable for the constructed dataset.

During the selection of deep learned feature methods, we evaluated AlexNet [42], ResNet [44], and VGG [43]. As depicted in Fig. 8, AlexNet [42], ResNet [44], and VGG [43] are presented as the blue, orange, and green lines. The overall performance and matching rate growth trends of these three methods are similar, with the matching rate first growing rapidly from Rank 1 to Rank 5 before gradually slowing down from Rank 6 to Rank 20. Specifically, VGG [43] achieved the highest mAP (i.e., 25.90%), Rank 1 (i.e., 68.34%), Rank 10 (i.e., 89.72%), and Rank 20 (i.e., 94.11%). VGG [43] outperforms ResNet [44] and AlexNet by 0.21% and 1.76% in terms of mAP and 2.80% and 4.47% in terms of Rank 1. Thus, VGG has a clear lead on mAP.



(a) The CMC comparison of hand-crafted features configured

(b) The CMC comparison of deep learned features configured

Fig. 8 The CMC comparison in Interior-134

Based on the above results, we fused the best performing hand-crafted feature method (i.e., HOG [40]) with the best performing deep learned feature method (i.e., VGG [43]) and achieved a performance surpassing all previously evaluated methods. Table 5 shows the performance comparison of the proposed HyNet with multiple state-of-the-art methods in Interior-134. The proposed HyNet achieves the highest mAP (i.e., 27.16%), Rank 1 (i.e., 72.87%), Rank 10 (i.e., 91.41%), and Rank 20 (i.e., 94.79%). As shown in Table 5, deep learned feature methods are mostly superior to hand-crafted feature methods. For example, the Rank 1 of VGG [43] is obviously higher (i.e., 38.84%) than the best hand-crafted feature method HOG [40], which also has a 13.01% lead on mAP.

The advantage of hand-crafted feature methods is to extract shallow features such as texture and color, which are not good at deep learned feature methods. The advantage of deep learned methods lies in extracting semantic features, in which deep learned methods are more robust than hand-crafted feature methods. Therefore, the fusion of hand-crafted and deep learned features can often improve retrieval performance (e.g., HyNet).

Moreover, the proposed HyNet outperforms both hand-crafted feature methods and deep learned feature methods. For example, HyNet defeats the best hand-crafted feature method HOG [40] and the best deep learned feature method VGG [43] by 14.27% and 1.26% in terms of mAP and 38.84% and 4.53% in terms of Rank 1. These results suggest that HyNet

Table 5 Performance comparison of the proposed HyNet method and multiple state-of-the-art methods in Interior-134

Type	Methods	mAP(%)	Rank 1	Rank 10	Rank 20
hand-crafted Features	LBP [39]	6.09	1.69	7.92	10.76
	SIFT [36]	6.31	5.17	33.36	49.76
	HOG [40]	12.89	34.03	57.04	63.94
Deep learned Features	AlexNet [42]	21.43	66.58	88.77	92.90
	ResNet [44]	23.10	68.13	89.04	93.30
	VGG [43]	25.90	68.34	89.72	94.11
Our Method	**HyNet**	**27.16**	**72.87**	**91.41**	**94.79**

Bold font indicates that the result is the most important conclusion in this table

can combine the strengths of hand-crafted feature methods and deep learned feature methods to extract salient and discriminative features.

The best performing method in hand-crafted features (i.e., HOG) achieved mAP of 12.89% and Rank 1 of 34.03%. The best performing method in deep learned features (i.e., VGG) respectively has 25.90% and 68.34% in term of mAP and Rank 1. The mAP of the fused feature model (i.e., HyNet) is 27.16%, and Rank 1 is 72.87%. These results show that HyNet can fuse hand-crafted features and deep learning features such that they complement each other.

4.3 Ablation experiments and analysis

4.3.1 Fixed VGG Fusion of Different Hand-crafted Methods in Interior-134

Table 6 shows a performance comparison between VGG [43] and the fusion of different hand-crafted feature methods in Interior-134. The fusion of VGG [43] with SIFT [36] or LBP [39] is not effective compared with VGG [43]. The Rank 1 of the fusion of VGG with SIFT or LBP decreased by 64.08% and 66.65% compared to VGG, respectively, and the mAP is reduced by 20.57% and 19.8%, respectively, compared with VGG [43]. The proposed HyNet outperforms VGG [43], gaining a 4.53% improvement on Rank 1 and a 1.26% improvement on mAP compared with VGG [43], indicating the effectiveness of the proposed HyNet.

4.3.2 Performance comparison of hynet with different configurations in interior-134

Table 7 shows the effect of choosing different cell sizes for the hand-crafted feature method (i.e., HOG [40]) on the performance of HyNet. When the cell sizes are 8 × 8, 16 × 16, 32 × 32, 64 × 64, respectively, the corresponding output feature dimensions of HyNet are 20,836, 7,876, 4,852, and 4,204. Thus, the fusion method works best when the cell is 64 × 64. At this point, the mAP is 27.16%, the Rank 1 is 72.87%, the Rank 10 is 91.41%, and the Rank 20 is 94.79%. Therefore, the cell size of 64 × 64 is selected. It can be observed that a larger cell size allowed us to obtain a smaller proportion of hand-crafted features, which could be fused with the deep learned features to achieve better accuracy. Figure 9 visualizes the above results and shows that CMC keeps increasing as the cell size increases and has an optimal performance when the cell is 64 × 64.

4.3.3 Performance comparison of hynet with fusion methods and basic methods in interior-134

Figure 10 shows the performance of the proposed HyNet compared with the two fusion methods (i.e., HOG+ResNet, HOG+VGG_2048) and basic methods (i.e., ResNet, VGG_2048,

Table 6 Fixed VGG fusion of different hand-crafted methods in Interior-134

Methods	mAP(%)	Rank 1	Rank 10	Rank 20
VGG [43]	25.90	68.34	89.72	94.11
VGG [43] + SIFT [36]	5.33	4.26	30.45	47.43
VGG [43] + LBP [39]	6.10	1.69	7.92	10.76
HyNet	**27.16**	**72.87**	**91.41**	**94.79**

Bold font indicates that the result is the most important conclusion in this table

Table 7 Performance comparison of HyNet with different configurations in Interior-134

Methods	Cell size	Features dimension	mAP(%)	Rank 1	Rank 10	Rank 20
HyNet	8 × 8	20836	13.07	35.18	58.93	64.82
	16 × 16	7876	15.29	49.32	72.40	77.94
	32 × 32	4852	20.17	70.57	88.77	91.68
	64 × 64	**4204**	**27.16**	**72.87**	**91.41**	**94.79**

Bold font indicates that the result is the most important conclusion in this table

VGG_4096) in Interior-134. The basic methods have a lower CMC compared to the fusion methods, indicating the improved performance of fusion methods. For example, HyNet outperformed the best performing 4096-dimensions VGG [43] (i.e., VGG_4096) among the basic methods by 4.53% on Rank 1 and 1.26% on mAP.

According to the basic model comparison shown in Fig. 10, VGG_4096 beats ResNet [44] by 0.21% and 2.80% on mAP and Rank 1, respectively. In addition, since the dimensions of VGG_4096 do not match that of ResNet [44], the performance comparison was conducted with VGG [43] reduced to the same dimension as ResNet. Therefore, we trained an additional 2048-dimensions VGG (i.e., VGG_2048) to compare with ResNet [44]. As can be observed from Fig. 10, VGG_2048 has decreased mAP and Rank 1 by 2.19% and 7.92%, respectively, compared to VGG_4096, which indicates that the dimension decrease of VGG has a significant effect on its performance. Compared with ResNet, VGG_2048 has a 0.61% higher mAP but a 7.71% lower Rank 1, indicating the importance of maintaining the default dimensions of VGG.

We also compared HyNet with other fusion methods due to its excellent performance. As can be observed from Figure 10, HyNet beats the ResNet [44] + HOG [40] fusion method by 3.30% and 0.62% on mAP and Rank 1, respectively. Similarly, HyNet outperforms the HOG + VGG_2048 method in terms of mAP and Rank 1 by 0.96% and 3.38%, respectively.

As shown in Table Table 8, VGG_2048 is an additionally trained VGG model with 2048 dimension feature. The training process changes the VGG dimension from 4096 to 2048 for

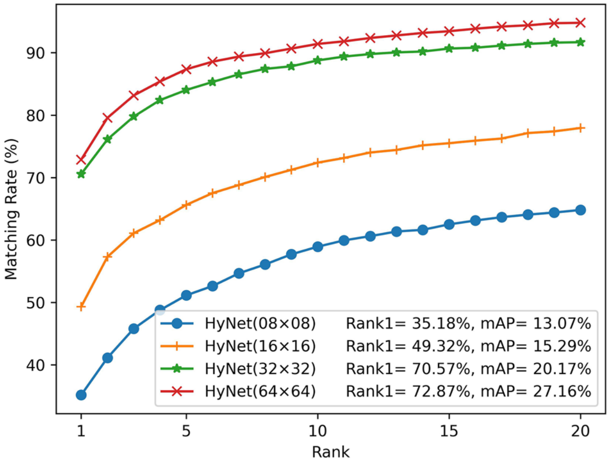


Fig. 9 Performance comparison of HyNet with different configurations in Interior-134

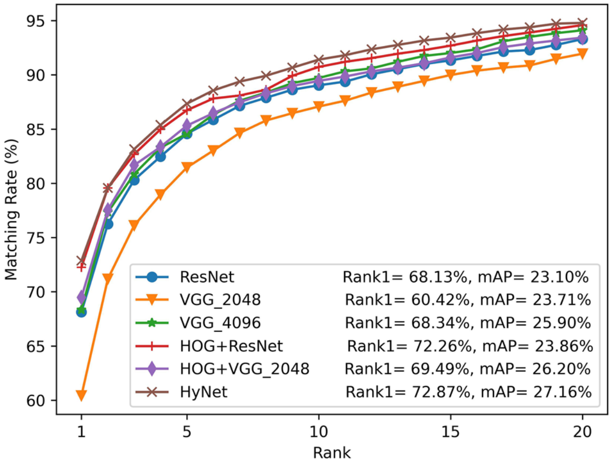


Fig. 10 Performance comparison of HyNet method with fusion and basic methods in Interior-134.The 2048-dimensions and 4096-dimensions VGG method are denoted as "VGG_2048" and "VGG_4096"

a fair comparison with the 2048-dimensional ResNet. We can see that in the same feature dimension, the mAP of HOG [40] + VGG_2048 is 2.34% higher than that of HOG [40] + ResNet [44], indicating that the overall performance of HOG [40] +VGG_2048 is better. It can be seen that HyNet achieved the highest mAP (i.e.,27.16%), Rank 1 (i.e.,72.87%), Rank 10 (i.e.,91.41%), and Rank 20 (i.e.,94.79%) compared with the other two fusion models. HyNet outperformed the HOG [40] + ResNet [44] method and HOG [40] + VGG_2048 method in mAP by 3.30% and 0.62%, respectively, while beating these two methods in terms of Rank 1 by 0.96% and 3.38%, respectively. Therefore, HyNet is superior.

4.4 Qualitative evaluation

We randomly selected a query texture and evaluated the differences in the retrieval by the three methods (HOG [40], VGG [43], and HyNet). The database offered the top ten textures that most closely resemble the query texture. Figure 11 shows an application example of indoor texture retrieval, and Fig. 12 shows an extended example of 3D modeling for texture retrieval.

Figure 13 shows the retrieval outcomes of the three methods. The query texture category is Venture Scattering. According to the retrieval results, the texture categories retrieved by HOG [40] are the Culture Stone, Rough Rug, Striped Wallpaper, and Venture Scattering. Only one of the ten retrieved textures is correct. The categories retrieved by VGG [43] are

Table 8 Performance comparison of HyNet with fusion methods in Interior-134

Methods	Features dimension	mAP(%)	Rank 1	Rank 10	Rank 20
HOG [40] + ResNet [44]	2156	23.86	72.25	90.73	94.58
HOG [40] + VGG_2048	2156	26.20	69.49	89.45	93.44
HyNet	**4204**	**27.16**	**72.87**	**91.41**	**94.79**

Bold font indicates that the result is the most important conclusion in this table

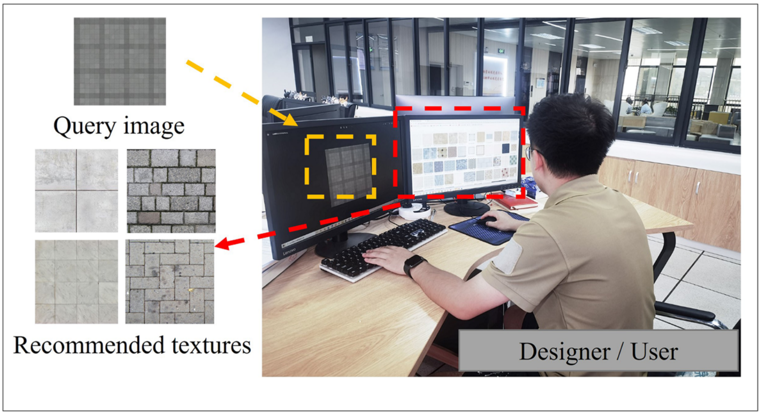


Fig. 11 application example of indoor texture retrieval

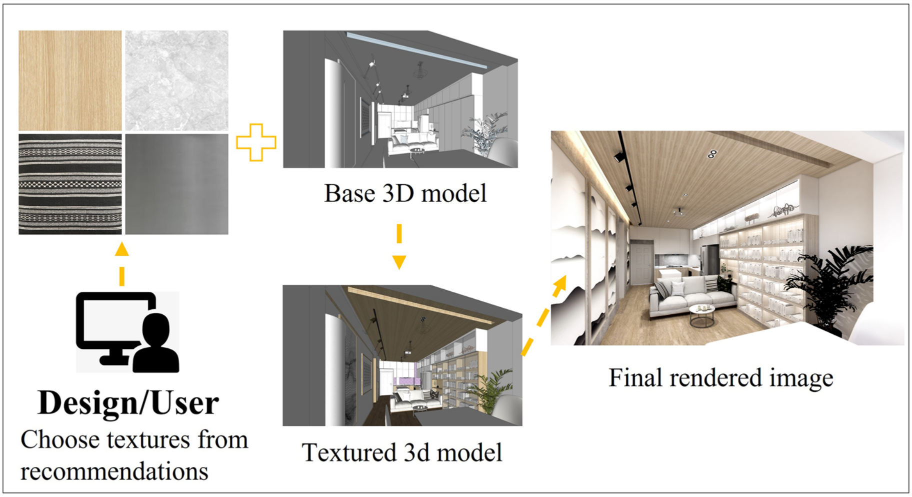


Fig. 12 An extended example of 3D modeling for texture retrieval

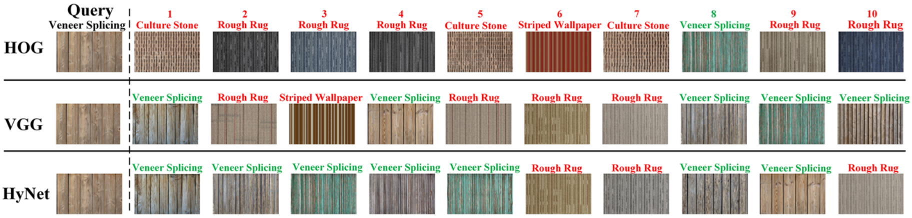


Fig. 13 Performance comparison of HyNet with fusion and basic methods in Interior-134.The 2048-dimensions and 4096-dimensions VGG methods are denoted as "VGG_2048" and "VGG_4096"

Rough Rug, Striped Wallpaper, and Veneer Splicing. Five of the ten retrieved textures are correct. The categories retrieved by HyNet are the Rough Rug and Veneer Splicing, and seven of the ten retrieved textures are correct. Ranking from HOG [40] to HyNet, the type and number of category classification mistakes are reduced, which is an intuitive proof of the superior performance of HyNet.

5 Conclusion

We presents a new fine-grained interior texture dataset “Interior-134” to aid the development of texture retrieval in the interior design sector. Subsequently, the dataset accuracy benchmark is established using hand-crafted features and deep learned features.

HyNet combines both hand-crafted features and deep learning features to enhance retrieval accuracy by recommending similar textures, sparing designers from aimless searches through texture databases. HyNet significantly improves texture retrieval performance at the algorithm level, achieving an impressive accuracy rate of 91.41% in the top 10 recommended textures. This advancement empowers users to swiftly retrieve their desired dataset of textures or texture websites, significantly boosting the efficiency of acquiring suitable textures.

HyNet exemplifies the design industry’s shift toward intelligence-driven practices by showcasing the feasibility of image retrieval techniques in interior design. HyNet offers a promising solution to streamline the texture searching process, enhancing overall productivity and effectiveness in design endeavors.

The following conclusions are drawn:

1. Proposing fusion model method named HyNet to improve retrieval accuracy.
2. A public dataset of multi-variety fine-grained indoor textures is established for researchers to use.
3. The performance of hand-crafted and deep learned methods are compared, and baselines are established for others to reach.
4. Experiments have shown that the retrieval performance of this method is better than that of the basic hand-crafted and deep learned features methods.

Future work:

1. The dataset categories in this paper are established on manual annotation. The automatic creation of artificial intelligence-based category annotations could be considered in the future.
2. In this paper, the feature fusion of poposed HyNet occurs in the back end, and future research can try front-end feature fusion.
3. Research has shown that the proposed method is effective, but the difficulty lies in collecting the dataset. If there is no texture that the user wants in the dataset, the retrieval will be ineffective. Future research could attempt to use GAN to control the generation of datasets. So that users can retrieve textures and generate the textures they want by secondary processing based on the retrieval results.
4. The retrieval results can be filtered using an aesthetic evaluation method to make the retrieved textures similar and beautiful.
5. The application fields of this method can be expanded, such as building location retrieval, clothing retrieval, fault retrieval, and medical retrieval.

Acknowledgements This research work were funded by the project FRG-23-021-FA and granted by the Research Fund of Macao University of Science and Technology (FRG-MUST) and National Social Science

Foundation of China Key Project of Art Science "Research on Chinese Animation Creation and a Theoretical Innovation under the Construction of National Cultural Image" (Grant No. 20AC003).

Data Availability The datasets generated during the current study are available from the corresponding author on reasonable request.

Declarations

Conflicts of interest The authors declare no conflicts of interest.

References

1. Hong, SA, Huu, QN, Viet, DC, Thuy, QDT, Quoc, TN (2023) Improving image retrieval effectiveness via sparse discriminant analysis. Multimedia Tools and Applications, pp 1–24. https://doi.org/10.1007/s11042-023-14748-9
2. Wang H, Qu H, Xu J, Wang J, Wei Y, Zhang Z (2022) Texture image retrieval based on fusion of local and global features. Multimedia Tools and Applications 81(10):14081–14104. https://doi.org/10.1007/s11042-022-12449-3
3. Kale M, Dash J, Mukhopadhyay S (2022) Efficient image retrieval system for textural images using fuzzy class membership. Multimedia Tools and Applications 81(26):37263–37297. https://doi.org/10.1007/s11042-022-13529-0
4. Zhuo, W, He, Z, Zheng, M, Hu, B, Wang, R (2021) Research on personalized image retrieval technology of video stream big data management model. Multimedia Tools and Applications, pp 1–18. https://doi.org/10.1007/s11042-020-10499-z
5. Majhi M, Pal AK (2021) An image retrieval scheme based on block level hybrid dct-svd fused features. Multimedia Tools and Applications 80:7271–7312. https://doi.org/10.1007/s11042-020-10005-5
6. Arun K, Govindan V (2018) A hybrid deep learning architecture for latent topic-based image retrieval. Data Science and Engineering 3(2):166–195. https://doi.org/10.1007/s41019-018-0063-7
7. Desai P, Pujari J, Sujatha C, Kamble A, Kambli A (2021) Hybrid approach for content-based image retrieval using vgg16 layered architecture and svm: An application of deep learning. SN Computer Science 2(3):1–9. https://doi.org/10.1007/s42979-021-00529-4
8. Qiao, C., Shen, F, Wang, X, Wang, R, Cao, F, Zhao, S, Li, C (2022) A novel multi-frequency coordinated module for sar ship detection. In: 2022 IEEE 34th international conference on tools with artificial intelligence (ICTAI), pp 804–811. IEEE
9. Devulapalli, S, Potti, A, Krishnan, R, Khan, MS (2021) Experimental evaluation of unsupervised image retrieval application using hybrid feature extraction by integrating deep learning and handcrafted techniques. Materials Today: Proceedings.https://doi.org/10.1016/j.matpr.2021.04.326
10. Zheng, L, Shen, L, Tian, L, Wang, S, Wang, J, Tian, Q (2015) Scalable person re-identification: A benchmark. In: Proceedings of the IEEE international conference on computer vision, pp 1116–1124. https://doi.org/10.1109/ICCV.2015.133
11. Jiang M, Li Z, Chen J (2019) Person re-identification using color features and cnn features. In, (2019) IEEE 4Th international conference on image, vision and computing (ICIVC), pp 460–462. IEEE. https://doi.org/10.1109/ICIVC47709.2019.8980977
12. Zhong, Z, Zheng, L, Cao, D, Li, S (2017) Re-ranking person re-identification with k-reciprocal encoding. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1318–1327 .https://doi.org/10.48550/arXiv.1701.08398

13. Zhu B, Xu T, Zheng B, Zhang Q, Sun Y, Liu A, Mao Z, Yan C (2021) Evolution of icts-empowered-identification: A general re-ranking method for person re-identification. Pattern Recogn Lett 150:94–100. https://doi.org/10.1016/j.patrec.2021.06.031
14. Shen, F, Xie, Y, Zhu, J, Zhu, X, Zeng, H (2023) Git: Graph interactive transformer for vehicle re-identification. IEEE Trans Image Process **32**:1039–1051. https://doi.org/10.48550/arXiv.2107.05475
15. Wang Z, Zhan J, Duan C, Guan X, Yang K (2022) Vehicle detection in severe weather based on pseudo-visual search and hog-lbp feature fusion. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering 236(7):1607–1618. https://doi.org/10.1177/09544070211036311
16. Li, M, Wei, M, Ren, J, He, X, Shen, F (2022) Enhancing pary features via contrastive attention module for vehicle re-identification. In: Conference on international conference on image processing. IEEE. https://doi.org/10.1109/ICIP46576.2022.9897943
17. Deng J, Hao Y, Khokhar MS, Kumar R, Cai J, Kumar J, Aftab MU et al. (2021) Trends in vehicle re-identification past, present, and future: A comprehensive review. Mathematics 9(24):3162. https://doi.org/10.3390/math9243162
18. Shen F, Zhu J, Zhu X, Xie Y, Huang J (2021) Exploring spatial significance via hybrid pyramidal graph network for vehicle re-identification. IEEE Transactions on Intelligent Transportation Systems. https://doi.org/10.1109/TITS.2021.3086142
19. Shen, F, Du, X, Zhang, L, Tang, J (2023) Triplet contrastive learning for unsupervised vehicle re-identification. https://doi.org/10.48550/arXiv.2301.09498
20. Shen F, Zhu J, Zhu X, Huang J, Zeng H, Lei Z, Cai C (2021) An efficient multi-resolution network for vehicle re-identification. IEEE Internet of Things Journal. https://doi.org/10.1109/JIOT.2021.3119525
21. Krause, J, Stark, M, Deng, J, Fei-Fei, L (2013) 3d object representations for fine-grained categorization. In: 4th International IEEE Workshop on 3D representation and recognition (3dRR-13), Sydney, Australia. https://doi.org/10.1109/ICCVW.2013.77
22. Liu, X, Liu, W, Mei, T, Ma, H (2016) A deep learning-based approach to progressive vehicle re-identification for urban surveillance. In: European conference on computer vision, pp 869–884. Springer. https://doi.org/10.1007/978-3-319-46475-6_53
23. Shen, F, Lin, L, Wei, M, Liu, J, Zhu, J, Zeng, H, Cai, C, Zheng, L (2019) A large benchmark for fabric image retrieval. In: 2019 IEEE 4th International conference on image, vision and computing (ICIVC), pp. 247–251. IEEE. https://doi.org/10.1109/ICIVC47709.2019.8981065
24. Gharaei NY, Dadkhah C, Daryoush L (2021) Content-based clothing recommender system using deep neural network. In, (2021) 26th International computer conference, computer society of Iran (CSICC), pp 1–6. IEEE. https://doi.org/10.1109/CSICC52343.2021.9420544
25. Ayyachamy, S, Alex, V, Khened, M, Krishnamurthi, G (2019) Medical image retrieval using resnet-18. In: Medical Imaging 2019: imaging informatics for healthcare, research, and applications, vol 10954, pp 1095410. International Society for Optics and Photonics. https://doi.org/10.1117/12.2515588
26. Shen, F, Shu, X, Du, X, Tang, J (2023) Pedestrian-specific bipartite-aware similarity learning for text-based person retrieval. In: Proceedings of the 31th ACM international conference on multimedia
27. Liu, B, Zhang, J, Zhang, X, Zhang, W, Yu, C, Zhou, Y (2019) Furnishing your room by what you see: An end-to-end furniture set retrieval framework with rich annotated benchmark dataset. https://doi.org/10.48550/arXiv.1911.09299
28. Fu H, Jia R, Gao L, Gong M, Zhao B, Maybank S, Tao D (2021) 3d-future: 3d furniture shape with texture. Int J Confl Manag 129(12):3313–3337. https://doi.org/10.1007/s11263-021-01534-z
29. Yang H, Shi P, He S, Pan D, Ying Z, Lei L (2019) A comprehensive survey on image aesthetic quality assessment. In, (2019) IEEE/ACIS 18th international conference on computer and information science (ICIS), pp 294–299. IEEE. https://doi.org/10.1109/ICIS46139.2019.8940355
30. Deng Y, Loy CC, Tang X (2017) Image aesthetic assessment: An experimental survey. IEEE Signal Proc Mag 34(4):80–106. https://doi.org/10.1109/MSP.2017.2696576
31. Elnemr HA (2016) Combining surf and mser along with color features for image retrieval system based on bag of visual words. J Comput Sci 12(4):213–222. https://doi.org/10.3844/jcssp.2016.213.222
32. Zenggang X, Zhiwen T, Xiaowen C, Xue-min Z, Kaibin Z, Conghuan Y (2021) Research on image retrieval algorithm based on combination of color and shape features. Journal of Signal Processing Systems 93(2):139–146. https://doi.org/10.1007/s11265-019-01508-y
33. Wu, S, Chen, Y-C, Li, X, Wu, A-C, You, J-J, Zheng, W-S (2016) An enhanced deep feature representation for person re-identification. In: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), pp 1–8 IEEE. https://doi.org/10.1109/WACV.2016.7477681
34. Ahmed KT, Ummesafi S, Iqbal A (2019) Content based image retrieval using image features information fusion. Information Fusion 51:76–99. https://doi.org/10.1016/j.inffus.2018.11.004

35. Ayadi W, Elhamzi W, Charfi I, Atri M (2019) A hybrid feature extraction approach for brain mri classification based on bag-of-words. Biomedical Signal Processing and Control 48:144–152. https://doi.org/10.1016/j.bspc.2018.10.010
36. Lowe, DG (1999) Object recognition from local scale-invariant features. In: Proceedings of the Seventh IEEE international conference on computer vision, vol 2, pp 1150–1157. IEEE. https://doi.org/10.1109/ICCV.1999.790410
37. Yan, K, Wang, Y, Liang, D, Huang, T, Tian, Y (2016) Cnn vs. sift for image retrieval: Alternative or complementary. In: Proceedings of the 24th ACM international conference on multimedia, pp 407–411. https://doi.org/10.1145/2964284.2967252
38. Mehmood Z, Abbas F, Mahmood T, Javid MA, Rehman A, Nawaz T (2018) Content-based image retrieval based on visual words fusion versus features fusion of local and global features. Arab J Sci Eng 43(12):7265–7284. https://doi.org/10.1007/s13369-018-3062-0
39. Ojala, T, Pietikainen, M, Harwood, D (1994) Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In: Proceedings of 12th international conference on pattern recognition, vol 1, pp 582–585. IEEE. https://doi.org/10.1109/ICPR.1994.576366
40. Zhu Q, Yeh M-C, Cheng K-T, Avidan S (2006) Fast human detection using a cascade of histograms of oriented gradients. In, (2006) IEEE computer society conference on computer vision and pattern recognition (CVPR'06), vol 2, pp 1491–1498. IEEE. https://doi.org/10.1109/CVPR.2006.119
41. O'Mahony, N, Campbell, S, Carvalho, A, Harapanahalli, S, Hernandez, G.V, Krpalkova, L, Riordan, D, Walsh, J (2019) Deep learning vs. traditional computer vision. In: Science and Information conference, pp 128–144. Springer. https://doi.org/10.1007/978-3-17795-9_10
42. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems 25. https://doi.org/10.1145/3065386
43. Simonyan, K, Zisserman, A (2014) Very deep convolutional networks for large-scale image recognition. https://doi.org/10.48550/arXiv.1409.1556
44. He, K, Zhang, X, Ren, S, Sun, J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778. https://doi.org/10.1109/CVPR.2016.90
45. Yuan, Z-W, Zhang, J (2016) Feature extraction and image retrieval based on alexnet. In: Eighth international conference on digital image processing (ICDIP 2016), vol 10033, pp 100330. International Society for Optics and Photonics. https://doi.org/10.1117/12.2243849
46. Ha I, Kim H, Park S, Kim H (2018) Image retrieval using bim and features from pretrained vgg network for indoor localization. Build Environ 140:23–31. https://doi.org/10.1016/j.buildenv.2018.05.026
47. Alzu'bi A, Amira A, Ramzan N (2017) Content-based image retrieval with compact deep convolutional features. Neurocomputing 249:95–105. https://doi.org/10.1016/j.neucom.2017.03.072
48. Shen F, Peng X, Wang L, Hao X, Shu M, Wang Y (2022) Hsgm: A hierarchical similarity graph module for object re-identification. In, (2022) IEEE international conference on multimedia and expo (ICME), pp 1–6. IEEE. https://doi.org/10.1109/ICME52920.2022.9859883
49. Georgiou T, Liu Y, Chen W, Lew M (2020) A survey of traditional and deep learning-based feature descriptors for high dimensional data in computer vision. International Journal of Multimedia Information Retrieval 9(3):135–170. https://doi.org/10.1007/s13735-019-00183-w
50. Li, M, Wei, M, He, X, Shen, F (2022) Enhancing part features via contrastive attention module for vehicle re-identification. In: 2022 IEEE International Conference on Image Processing (ICIP), pp 1816–1820. IEEE
51. Liu Y, Peng Y, Lim K, Ling N (2019) A novel image retrieval algorithm based on transfer learning and fusion features. World Wide Web 22(3):1313–1324. https://doi.org/10.1007/s11280-018-0585-y
52. Ho Y, Wookey S (2019) The real-world-weight cross-entropy loss function: Modeling the costs of mislabeling. IEEE Access 8:4806–4813. https://doi.org/10.1109/ACCESS.2019.2962617
53. Lee J, Qian G, Beach A (2021) A sample weighting and score aggregation method for multi-query object matching. In, (2021) 17th IEEE international conference on advanced video and signal based surveillance (AVSS), pp 1–8. IEEE. https://doi.org/10.1109/AVSS52988.2021.9663848