



# Image retrieval method based on deep learning semantic feature extraction and regularization softmax

Qinghai Wu<sup>1</sup>

Received: 11 January 2019 / Revised: 5 March 2019 / Accepted: 8 April 2019 /  
Published online: 29 May 2019  
© Springer Science+Business Media, LLC, part of Springer Nature 2019

## Abstract

In content-based image retrieval (CBIR), an image retrieval method combining deep learning semantic feature extraction and regularization Softmax is proposed for the “semantic gap” between the underlying visual features and high-level semantic features. First, the deep Boltzmann machine (DBM) and the convolutional neural network (CNN) in the deep learning method are combined to construct a convolution depth Boltzmann machine (C-DBM), which enables it to extract High-order semantic features of images, and robust to image scaling, affine and other transformations. Then, the Dropout regularized Softmax classifier is used to classify the image features. Finally, the image is retrieved according to the sort output. The experimental results show that the proposed method can extract semantic features effectively and has high retrieval accuracy. The classification precision rate in STL-10 image data set reaches 60.3%.

**Keywords** Image retrieval · Semantic features · Deep Boltzmann machine · Convolutional neural network · Dropout regularization

## 1 Introduction

With the rapid development of mobile Internet technology, users upload and share massive images every day. How to enable users to accurately find the information they need in the massive image data resources is an important research topic at present. Content-based image retrieval (CBIR) emerges as the times require. It can search for corresponding images [6, 10, 14, 18] that meet the query conditions in the image database according to the objective visual features such as pixel information, color, texture, shape, etc. contained in the image itself, and no need to manually annotate the image.

One of the most fundamental problems in CBIR is how to achieve effective expression of images. Because of this, the extraction and expression of features has been widely concerned.

---

✉ Qinghai Wu  
904201126@qq.com

<sup>1</sup> Jilin Agricultural Science and Technology University, Jilin 132101, China

The traditional content-based image retrieval technology is based on the underlying visual features of the image, and it has a huge “semantic gap” problem with human perception of images. Therefore, semantic-based image retrieval has become one of the key issues in the field of image retrieval [1, 16].

For the construction of image semantic features, a more common approach is to directly classify the underlying features of the image to obtain semantic features. For example, the literature [7] proposes to learn the semantics of images by learning the joint probability model of images and annotations. The main idea is to model the image feature vector and the text annotation with semantic information as a non-parametric Gaussian kernel model, and solve the “semantic gap” problem through the model. The disadvantage of this method is that the selection of visual features plays a decisive role and the robustness is poor.

In recent years, deep learning technology has been greatly developed. Deep learning technology is developed on the basis of drawing lessons from the principle of human brain visual mechanism. It is a process of iteration and abstraction layer by layer. The greatest advantage of in-depth learning is that it can learn image features independently, from bottom edge features to object structure features, and even more abstract features [3, 11]. Convolutional Neural Network (CNN), Deep Boltzmann Machine (DBM) and Automatic Encoders (AE) are classical deep learning algorithms. They have achieved good results in image classification tasks. For example, the literature [9] proposes a binary image retrieval method based on Deep Belief Network (DBN) and Softmax classifier. However, in the process of using the back propagation to correct the connection weight and offset of the network, the DBN algorithm is prone to problems such as small gradient, low learning rate and slow error convergence [12]. In [15], Stacked Auto encoder (SAE) is used to extract geometric features for image classification. However, SAE is easily affected by the imbalance of training data, and there are many parameters in a single SAE model. The classification effect is easy to change with the change of parameters and the robustness is poor [8, 20]. Therefore, some scholars have proposed a Stack Denoising Automatic Encoder (SDAE), which adds appropriate noise to the input of SAE and takes the noisy feature as the original complete feature. This method improves the generalization ability of the SAE model. For example, the literature [2] proposed a gesture image recognition method based on SDAE, which improved the performance of SAE to some extent. However, SDAE is a deep-structured neural network model that requires layer-by-layer training. Therefore, the amount of calculation is large, the training speed is slow, and it takes a long time to adjust to the optimal parameters.

In view of the semantic gap problem, this paper introduces a structure and training method of convolution depth Boltzmann machine (C-DBM) combined with CNN and DBM to construct a model from the underlying visual features to the advanced semantic features, which is a layer-by-layer iterative, layer-by-layer abstract deep network mapping model. The model aims to reduce the semantic gap and obtain image semantic features. Finally, according to the extracted semantic features, the image classification and retrieval are performed by the Dropout regularized softmax classifier. The experimental results demonstrate the effectiveness of the proposed method.

## 2 Convolutional neural network (CNN)

CNN is a kind of artificial neural network. It is affected by delayed neural network (TDNN) and uses weight sharing to reduce the size of network parameters. It consists of a convolutional layer and a pooled layer. The convolution layer performs a convolution operation, and the

linear filter inputs the signal to each local receptive field area in a sliding window manner to perform inner product operations, and generates a pair of input signal positions through a nonlinear activation function. The activation value is finally obtained, and finally the feature map is obtained [5, 17]. In addition, CNN usually combines the Softmax classifier to solve multi-classification problems. The CNN architecture for image retrieval is shown in Fig. 1.

The input content of the previous layer and the learning kernel (weight) constitute a single neuron in the convolutional layer. In the same layer, neurons with the same feature map share the same kernel, and the kernels of neurons with different feature maps are also different. The input and output expression of a neuron is:

$$u_j^l = f\left(\sum_{i=1}^N u_i^{l-1} * w_{ji} + b_j^l\right) \tag{1}$$

In the above formula,  $u_i^{l-1}$  is the input neuron of the  $l-1$  layer,  $N$  is the number of input neurons,  $b_j^l$  is the bias term, and  $f$  is the activation function.

For a given input feature map, the pooling operation of the pooling layer is to sample it. The sampling process does not change the number of input feature maps, but the feature map becomes smaller, expressed as:

$$u_j^l = f\left(\beta_j^l \text{down}(u_j^{l-1}) + b_j^l\right) \tag{2}$$

The CNN training process targets minimizing the error function. A classification problem with  $N$  training samples, the reconstruction error is expressed as follows:

$$E = \frac{1}{2} \sum_{n=1} \sum_{k=1} (s_k^n - y_k^n)^2 \tag{3}$$

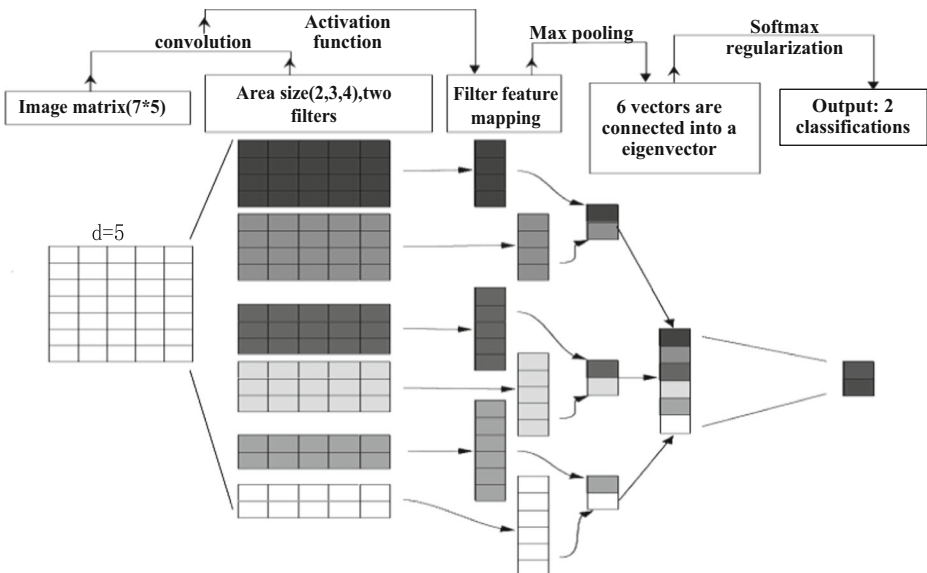


Fig. 1 CNN architecture for image retrieval

In the above formula,  $s_k^n$  is the  $n$  dimension of the  $k$  sample, and  $y_k^n$  is the output value of the  $n$  input sample at the  $k$  output layer. The sum of the errors on each neuron sample represents the error of the entire data set. The error for a single sample is expressed as:

$$E = \frac{1}{2} \sum_{k=1}^K (s_k^n - y_k^n)^2 \tag{4}$$

The training phase of CNN consists of two steps: forward propagation and back propagation. Forward propagation is the process by which a feature map passes from the current layer to the next layer through a predefined activation function with learning parameters (weights and offsets). For example, the output of layer  $l$  is defined as  $v^l = f(u^l)$ , where  $u^l = (W^l v^{l-1} + b^l)$ . During the process of back propagation, the weights  $W^l$  and  $b^l$  are updated by a stochastic gradient descent strategy. The data is then normalized so that they exhibit a normal distribution in the feature space, which speeds up the convolution.

### 3 Deep Boltzmann machine (DBM)

DBM is an undirected graph model [13], as shown in Fig.2. DBM can learn high-level representations from the data itself through unsupervised learning, and then fine-tune the model with a small amount of annotated data in a supervised learning manner. Unlike the Deep Trusted Network (DBN), the estimation derivation process of DBM includes top-down feed-back, which enables better transmission and processing of input data uncertainty and ambiguity [4].

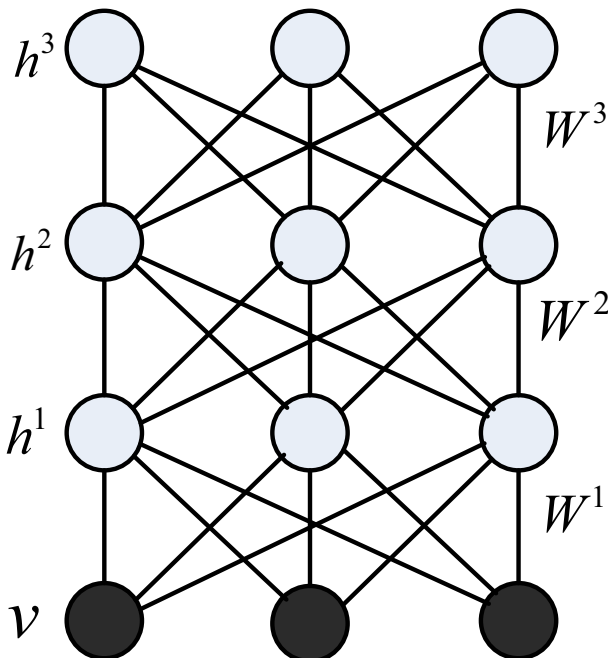


Fig. 2 Deep Boltzmann model

Considering a two hidden layer DBM and ignoring the deviation values in the visible and hidden layers, the energy of the model can be defined as:

$$E(v, h^1, h^2; \theta) = -v^T W^1 h^1 - h^{1T} W^2 h^2 \tag{5}$$

Where  $\theta = \{W^1, W^2\}$  is the model parameter,  $W^1$  represents the connection weight between the visible layer  $v$  and the first hidden layer  $h^1$ , and  $W^2$  is the connection weight of the first hidden layer  $h^1$  and the second hidden layer  $h^2$ . The probability of assigning to a visual layer state  $v$  at this time is:

$$p(v; \theta) = \frac{1}{Z(\theta)} \sum_{h^1, h^2} \exp(-E(v, h^1, h^2; \theta)) \tag{6}$$

The conditional probabilities of the visible layer unit, the first hidden layer unit, and the second hidden layer unit are respectively:

$$p(v_i = 1 | h^1) = \sigma \left( \sum_j W_{ji}^1 h_j \right) \tag{7}$$

$$p(h_j^1 = 1 | v, h^2) = \sigma \left( \sum_i W_{ji}^1 v_i + \sum_m W_{mj}^2 h_m^2 \right) \tag{8}$$

$$p(h_m^2 = 1 | h^1) = \sigma \left( \sum_j W_{jm}^2 h_j^1 \right) \tag{9}$$

The only top layer in the DBN is the Restricted Boltzmann Machine (RBM). In contrast, the adjacent two layers in the DBM form an RBM. For RBM, after training it, the entire model can be expressed as:

$$p(v; \theta) = \sum_{h^1} p(h^1; W^1) p(v | h^1; W^1) \tag{10}$$

Where  $p(h^1; W^1) = \sum_v p(h^1, v; W^1)$  is an implicit prior to  $h^1$ , which is defined by the parameter.

The second layer of RBM can be obtained by replacing  $p(h^1; W^1)$  with  $p(h^1; W^2) = \sum_{h^2} p(h^1, h^2; W^2)$ . It should be noted here that the newly added hidden layer does not change the probability distribution of the model. This way, using only one RBM replacement for the topmost hidden layer at a time, you can increase the depth of the model without changing the underlying probability. Replacing  $p(h^1; W^1)$  with the second RBM is equivalent to improving the model distribution of  $p(h^1; W^1)$ . It is therefore feasible to use the upper and lower RBMs and then average them to derive the  $p(h^1; W^1, W^2)$ . Using  $W^1$  bottom-up and  $W^2$  top-down propagation is equivalent to calculating double input data for  $v$ , because from the perspective of the graph model,  $h^2$  is a value that depends on  $v$ .

The structure of DBM determines that its training process is different from DBN. The training algorithm of DBM mainly adjusts the weights to compensate for the problem that there is only bottom-up and no top-down signal flow. The DBM training

process is more complicated than the DBN. Since the DBM belongs to the undirected graph model, the middle layer is directly connected to the upper and lower layers, so the mean field algorithm is usually used.

### 4 Image semantic feature extraction combined with CNN and DBM

The semantic features of the image are hierarchical, and the low-level image features have lower abstraction and higher correlation with the image data content itself. High-level image features have a high degree of abstraction and are less correlated with the content of the image data itself. Therefore, it is considered to establish a hierarchical learning model, which uses unsupervised learning to learn image data, so as to obtain the characteristics of the data itself. As the level increases, the learned features become high-order features of the input data. In this paper, the deep learning model is used to learn the semantic features of the image, and the abstraction of the model is improved by the increase of the depth, so as to establish the semantic hierarchical structure, and then use the classification model to implement semantic-based classification of the image.

CNN has good adaptability to image stretching, affine and other changes, but it ignores high-order statistical features in the image. DBM has good properties in capturing high-order dependencies in images, but it is slower when applied to larger images, and is more sensitive to external changes, and lacks the ability to capture local invariance. Therefore, this paper considers combining these two models to obtain a deep learning model with local invariance and ability to learn high-order statistical features.

A Convolutional-Deep Boltzmann Machine (C-DBM) can be obtained by using CRBM instead of each RBM in the DBM. Using C-DBM as the semantic feature extraction module of the image semantic classification model, the C-DBM-based image semantic classification model shown in Fig. 3 can be obtained.

The training process of C-DBM involves only the first three layers, and there is no participation in the classification layer. Modifying multiple convolution kernels at the input

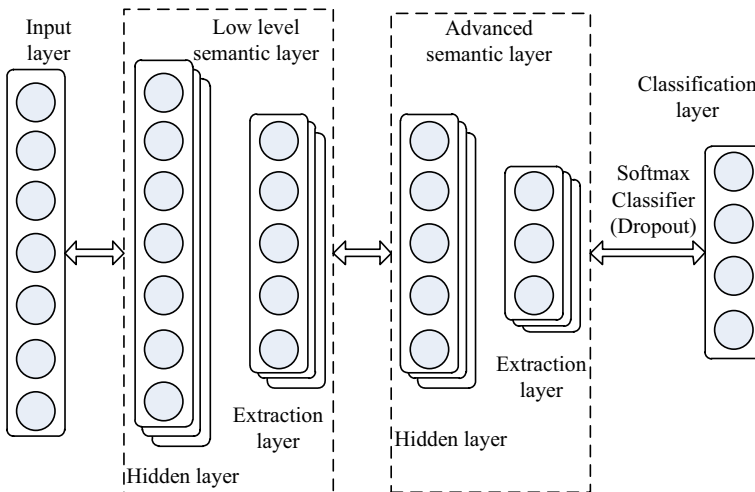


Fig. 3 Image semantic classification model based on C-DBM

layer and then performing convolutional mapping yields an implicit layer containing low-level semantics. On this basis, the extraction operation is performed, and finally the low-level semantic unit is obtained. The double convolution map here and the DBM weight doubling principle are the same, and both are to compensate for the loss caused by the top-down input. Then, based on this, the high-level semantics are extracted. The basic process is similar to DBM. The whole process is completely unsupervised learning, and the extracted features can be seen as a more essential description of the data content itself.

The training algorithm of C-DBM is the same as the training algorithm of DBM. In the process of bottom-up training propagation, the weights need to be doubled to compensate for the loss of probability caused by the top-down signal. When the modified C-DBM is training, the input formulas of the hidden sub-layers of the first layer and the second layer are as follows:

$$I(h_{i,j}^{1,k}) = (2W_k^1 \cdot V + b_k^1)_{i,j} \tag{11}$$

$$I(h_{i,j}^{2,k}) = \left( \sum_l^{n_1} W_{kl}^2 \cdot p^{1,l} + b_k^2 \right)_{i,j} \tag{12}$$

Here  $h_{i,j}^{1,k}$  denotes the unit in the  $k$ sublayer of the first hidden layer directly connected to the visible layer,  $W_k^1$  denotes the  $k$  convolution kernel in the first CRBM, and  $b_k^1$  denotes the deviation on the  $k$ sublayerof the first hidden layer.  $h_{i,j}^{2,k}$  denotes a unit in the  $k$ sublayerof the second hidden layer,  $n_1$  denotes the convolution kernel number of the first hidden layer,  $h^{1,l}$  denotes the  $l$ sublayer in the first hidden layer, and  $W_{kl}^2$  denotes the second hidden layer. The  $k$  convolution kernel connected to the sublayer $l$  of the first hidden layer, and  $b_k^2$  represents the deviation on the  $k$  implicit sublayer.

All the hidden layer and the abstraction layer unit are binary. In order to obtain the binary state of each binary unit, the posterior probability of the hidden layer and the extracted layer must be obtained in the training process. During the training process of C-DBM,the posterior probability formulas of the hidden layer unit and the abstract layer unit are as follows:

$$p(h_{i,j}^k = 1|v) = \frac{\exp(I(h_{i,j}^k))}{1 + \sum_{(i',j') \in B_\alpha} \exp(I(h_{i',j'}^k))} \tag{13}$$

$$p(h_\alpha^k = 1|v) = \frac{1}{1 + \sum_{(i',j') \in B_\alpha} \exp(I(h_{i',j'}^k))} \tag{14}$$

After the initial training, the parameters of the model are initialized to a better position. Based on this, the C-DBM is trained in Mean Field (MF) to fully train the model. It should be noted that this process is also conducted in an unsupervised manner.

Since the C-DBM algorithm used in this paper uses two hidden layers C-DBM, the first hidden layer, that is, the low-level semantic layer simultaneously receives input information

from the visible layer and the advanced semantic layer. The posterior probability formula of the hidden layer in the mean field training is modified to:

$$I(h_{i,j}^{1,k}) = (W_{kl}^1 \cdot v + b_k^1)_{i,j} \tag{15}$$

$$I(p_{i,j}^{1,k}) = \left( \sum_{l=1}^{n_l} \tilde{W}_{lk}^1 h^{2,l} + b_k^1 \right)_{i,j} \tag{16}$$

$$p(h_{i,j}^1 = 1 | v, h') = \frac{\exp(I(h_{i,j}^1) + I(p_\alpha^1))}{1 + \sum_{(i',j') \in B_\alpha} \exp(I(h_{i',j'}^1) + I(p_\alpha^1))} \tag{17}$$

$$p(p_\alpha^1 = 1 | v, h') = \frac{1}{1 + \sum_{(i',j') \in B_\alpha} \exp(I(h_{i',j'}^1) + I(p_\alpha^1))} \tag{18}$$

In the above formula,  $\tilde{W}_{lk}^1$  represents the left and right and vertical flip operations on the convolution kernel between the  $k$ sublayer in the first hidden layer and the  $l$ sublayer in the second hidden layer, which is.

Equations (17) and (18) adopt a method called “probability type maximum extraction”, which embodies the probability sampling of the context information composed from the visible layer and the second hidden layer, and realizes Standardization and comprehensive inference of context information flow.  $(i',j') \in B_\alpha$  represents a probability calculated from the context and all convolutional layer units within the extraction region that are outputted with  $\alpha$ . Since the second hidden layer and the pooled layer have no top-down information flow, they are the same as the formula that propagates from low to upward during training, and will not be described here.

The process of image semantic classification algorithm based on C-DBM is mainly divided into three parts. (1) First, layer-by-layer training and mean field training are performed on the first three layers of the network. (2) Then take the output of the C-DBM module as the extracted semantic feature. (3) Supervised training of the Softmax classifier according to the extracted features, thereby completing the training process of the entire network. The learning process of layer-by-layer training and mean field training adopts the unsupervised learning method to learn the semantic features in the image content without the participation of category information.

### 5 Softmax classifier with dropout regularization

After obtaining the image semantic features, it is necessary to perform image classification and recognition by a classifier. The end of a traditional CNN network typically uses a fully connected softmax classifier. Due to the large number of parameters of the neural network, over-fitting is likely to occur in practice. This paper introduces the Dropout algorithm [19] on the classifier side, which can effectively prevent the model from over-fitting and make the model have better generalization ability.



The connection method of the Dropout algorithm is to randomly set the original input data to a certain proportion of  $\rho$  to 0, and only other elements that are not set to 0 participate in the operation and connection.

Suppose there is a neural network with a hidden layer of the L layer whose input-output relationship is as shown in Eqs. (19) and (20):

$$z_i^{l+1} = w_i^{l+1}y^l + b_i^{l+1} \quad (19)$$

$$y_i^{l+1} = f(z_i^{l+1}) \quad (20)$$

Where  $z$  is the input vector,  $y$  is the output vector,  $w$  is the weight,  $b$  is the offset vector, and  $f$  is the activation function, which is used to limit the amplitude.

When adding to Dropout, the input-output relationship of the feed forward neural network is as shown in Eqs. (21)–(23), where the Bernoulli random variable  $r_i^l$  obeys the Bernoulli distribution with probability  $\rho$ :

$$\tilde{y}^l = r^l y^l \quad (21)$$

$$z_i^{l+1} = w_i^{l+1}\tilde{y}^l + b_i^{l+1} \quad (22)$$

$$y_i^{l+1} = f(z_i^{l+1}) \quad (23)$$

For the sake of simplicity of description, we assume that each update parameter takes only one sample. The specific process is as follows: First, the input vector is set to 0 according to the ratio  $\rho$ , and the element without 0 is involved in the operation and optimization of the classifier; Then accept the input vector of the second sample. At this time, the elements participating in the training are also selected according to the random setting of 0 until all the samples have been learned once. Since each time a sample is entered, the way to set 0 is random, so the network weight parameters are different for each update. In the final prediction process, the parameters of the entire network are multiplied by  $1-\rho$  to obtain the final classifier network parameters. Because the parameters of each update are different, the Dropout algorithm can be regarded as a combination of neural networks into multiple models, which can effectively prevent over-fitting and improve the prediction accuracy of the model [17].

For a fully connected layer network, the effect of Dropout on all hidden layers is better than that on only one hidden layer, and the probability should be chosen appropriately. Too extreme will lead to poor results. Through many experiments, the best probability should be chosen to be 0.5. As shown in Fig. 4, when  $\rho=0.5$ , half of the neurons in the hidden layer of the Dropout neural network are set to 0, which means that it does not work during the training. It is restored when the probability is not set to 0 during the next training.

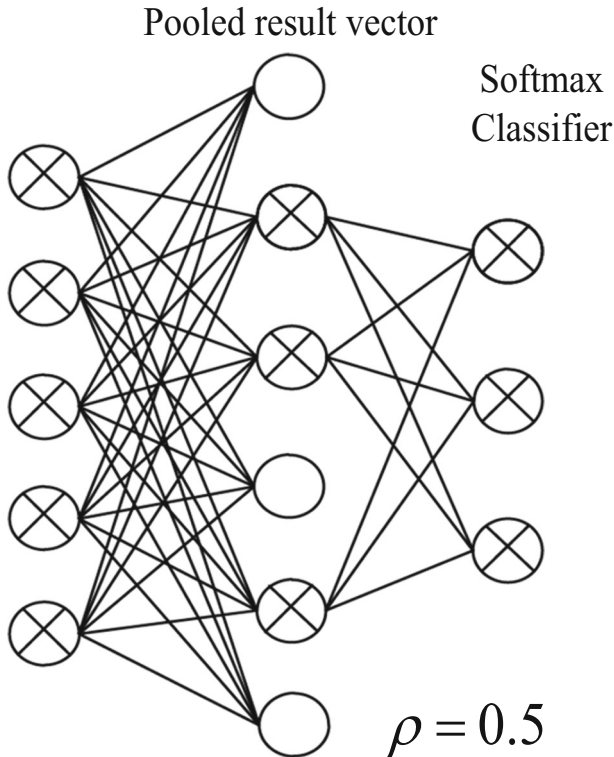


Fig. 4 Schematic diagram of neural network connection with Dropout  $\rho = 0.5$

## 6 Experimental results and analysis

### 6.1 Experimental setup

The STL-10 dataset is a public image collection from Stanford University that studies image recognition datasets for unsupervised feature learning and deep learning algorithms. The STL-10 includes ten types of color images with an image resolution of  $96 \times 96$ , covering airplanes, birds, cars, cats, deer, dogs, horses, monkeys, boats, trucks, and more. Each type of image includes 500 training images and 800 test images. In the data set, in addition to the above-mentioned tagged categories, it also includes other types of images, such as animals (bears, rabbits, etc.), and cars (trains, buses). A partial image of the STL-10 data set is shown in Fig. 5. From this, it can be seen that the difference among the images in one type is very large.

The input layer of the C-DBM model is set to a size of  $32 \times 32 \times 3$  (that is, the input can be regarded as three mapping layers of size  $32 \times 32$ ). The convolutional layer in the first hidden layer contains 6 feature maps, the convolution kernel size is  $5 \times 5$ , and the extraction layer has an extraction area size of  $2 \times 2$ . The convolutional layer of the second hidden layer contains 8 feature maps, the size of the convolution kernel is  $7 \times 7$ , and the size of the extraction region of the extraction layer is  $2 \times 2$ . Finally, the output units of the model are combined into a one-dimensional vector. The activation function of the model uses the sigmoid function. It should be pointed out that, as mentioned above, the perceptual domain of visual cortical cells becomes

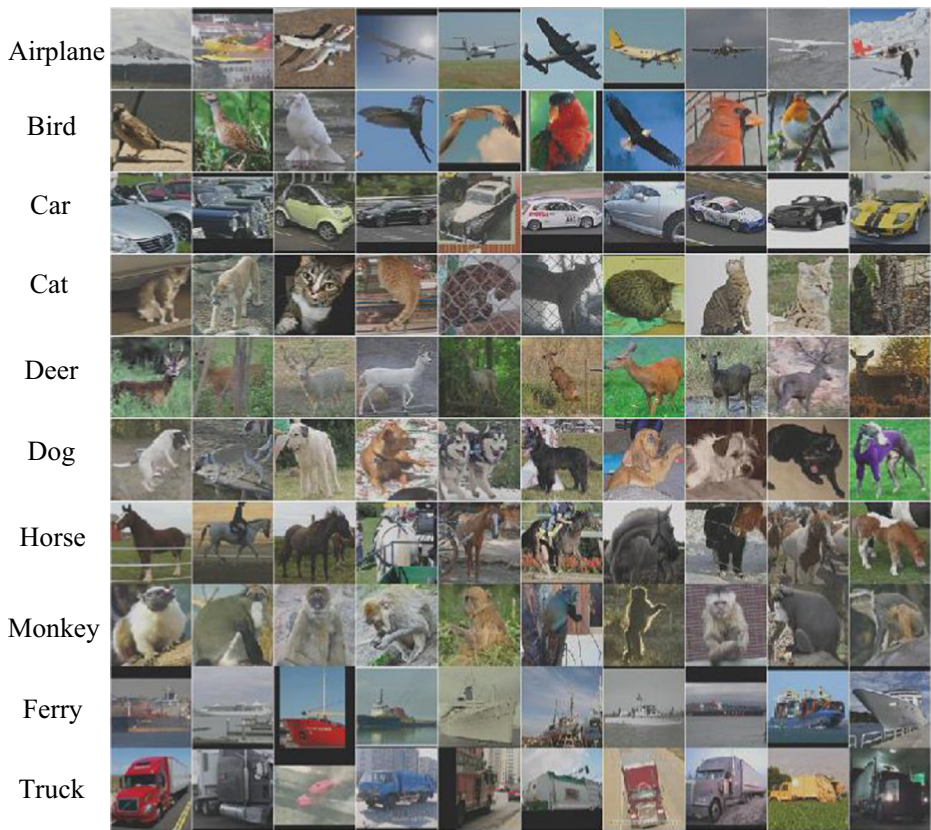


Fig. 5 STL-10 image dataset part of the picture

larger as the level increases. Therefore, the second layer here uses a larger convolution kernel, which is more in line with the principle of biological vision, which is also the place where the model of this paper is different from the conventional convolutional neural network.

All modules of the system are completed on the Matlab2014a GUI platform. The experimental platform is a Windows 7 64-bit system PC, CPU is Intel Core i5–6400, 2.7GHz clock, and memory is 8G.

## 6.2 Performance indicators

The evaluation indicators of the retrieval system are: precision rate, recall rate, and F value. The recall rate reflects the comprehensiveness of the search within the number of limited return images; the precision rate reflects the retrieval accuracy of the system. These two indicators are a contradiction. In many cases, it is difficult to meet the requirements of two indicators at the same time. The general retrieval system only needs to be able to reach an optimal balance point. The F value is such an indicator for comprehensive assessment of accuracy and recall.

$$\text{Precision Rate} = \frac{\text{Number of images related to search results}}{\text{Total number of images retrieved}}$$

$$\text{Recall Rate} = \frac{\text{Number of images related to search results}}{\text{Total number of related images in the image library}}$$

$$F = \frac{2 \times \text{precision rate} \times \text{recall rate}}{\text{precision rate} + \text{recall rate}}$$

### 6.3 Performance analysis

First, verify the feasibility of the image retrieval system. In the image retrieval module, the user inputs an image himself and returns the retrieved semantic similar image. Enter an image of a pair of aircraft and horses respectively, and retrieve the first four images of the output as shown in Fig. 6. As you can see, these images retrieved are very similar to the input image and belong to the same category. This indicates that deep learning can extract robust semantic features from complex images and has good linear reparability. Moreover, it is very robust to factors such as color, illumination and background.

The method is compared with several existing methods, which are the retrieval methods based on deep belief network (DBN) and Softmax classifier proposed in [9], and the image classification method based on stack denoising automatic encoder (SDAE) proposed in [2], both of which are classical methods. After these methods are trained multiple times on the STL-10 image dataset, the retrieval accuracy and recall rate obtained on each type of image are shown in Table 1. It can be seen that compared with SDAE and DBN algorithms, the C-DBM method proposed in this paper exceeds SDAE and DBN in terms of average classification accuracy and average recall rate. In addition, the average F value of the DBN method is 0.5653; the average F value of the SDAE method is 0.6036; the average F value of the method is the highest, reaching 0.6315. This also illustrates the effectiveness of the method in this paper.

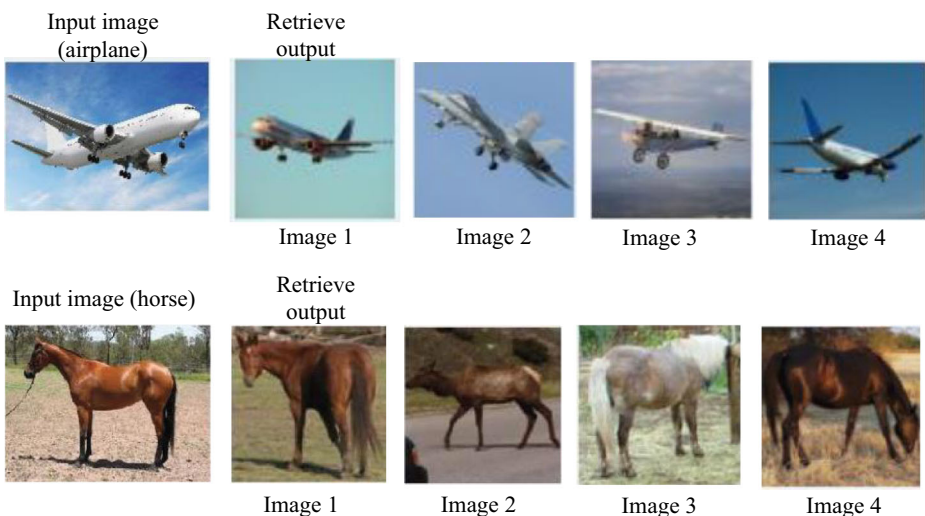


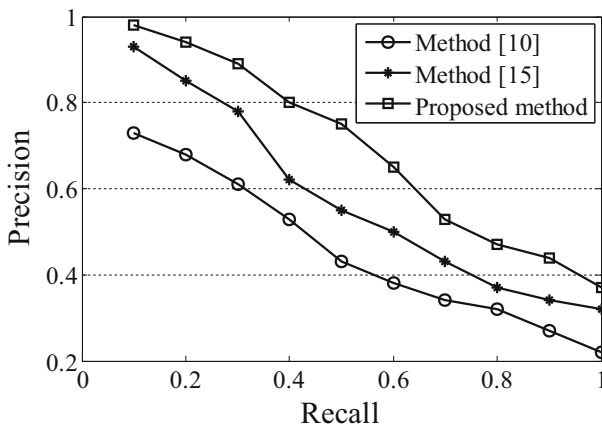
Fig. 6 Retrieve the first 4 most similar images returned

**Table 1** Retrieval performance of three methods on each type of image

Image type	Precision rate (%)			Recall rate (%)		
	Literature [9]	Literature [2]	Method of this paper	Literature [9]	Literature [2]	Method of this paper
Airplane	61.24	58.29	66.28	60.62	57.32	65.15
Bird	57.32	65.34	64.85	56.74	64.23	63.75
Car	52.27	64.25	59.78	51.74	63.16	58.78
Cat	61.98	66.31	65.31	61.36	65.18	64.20
Deer	56.54	51.34	58.92	55.97	50.51	57.94
Dog	58.37	59.82	64.31	57.78	58.82	63.22
Horse	49.58	57.48	70.64	49.08	56.53	69.42
Monkey	52.92	55.35	65.25	52.39	54.44	64.14
Ferry	62.39	58.83	62.18	61.76	57.85	61.13
Truck	55.64	66.67	59.46	55.08	65.53	58.47
Average	56.82	60.37	63.69	56.25	60.36	62.62

SDAE has better learning ability than SAE and can be used to classify image semantic features. By using convolutional mapping of local data and then learning the statistical features in the data, this has higher performance. However, C-DBM can better learn high-order semantic features in images and has better performance in applying to image semantic classification. This shows that compared with the classical deep learning model, the semantic model used in this paper extracts the semantic features in the image better. The learned semantic features are more suitable for semantic classification tasks than the traditional semantic features.

On the STL-10 image dataset, set the interval of recall rate to 0.1, and calculate the precision rate based on the recall rate. The precision rate - recall rate curve (PR curve) is shown in Fig. 7 (the horizontal axis is the recall rate and the vertical axis is the precision rate). It can be seen from the figure that the precision rate of the proposed method is better than other methods when the recall rate is the same, indicating that the algorithm can improve the precision rate of the search.



**Fig. 7** Precision rate - recall rate curve

## 7 Conclusion

In order to solve the “semantic gap” between image features in traditional image retrieval, an image retrieval method combining deep learning semantic feature extraction and regularization Softmax is proposed. Combine DBM with CNN to construct a Convolutional Depth Boltzmann Machine (C-DBM) to extract effective semantic features. The Dropout regularized Softmax classifier is used to classify and identify image features. The experimental results on the STL-10 image dataset show that the proposed method can extract semantic features effectively and has high retrieval accuracy.

In the future work, a simplified DBM model is combined with CNN to ensure the retrieval accuracy and reduce the computational complexity.

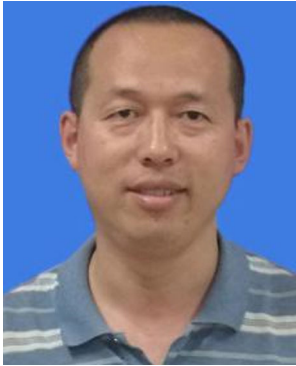
**Acknowledgments** This work was financially supported by project of Jilin province science and technology development plan, 20180623004TC.

## References

1. Alzu Bi A, Amira A, Ramzan N (2015) Semantic content-based image retrieval: a comprehensive study[J]. *Journal of Visual Communication & Image Representation* 32:20–54
2. Budiman A, Fanany MI, Basaruddin C (2014) Stacked Denoising Autoencoder for feature representation learning in pose-based action recognition[C]// *Consum Electron*
3. Chan TH, Jia K, Gao S et al (2014) PCANet: a simple deep learning baseline for image classification[J]. *IEEE Trans Image Process* 24(12):5017–5032
4. Feng F, Li R, Wang X (2015) Deep correspondence restricted Boltzmann machine for cross-modal retrieval[J]. *Neurocomputing* 154(1):50–60
5. Fu R, Li B, Gao Y, et al. (2017) Content-based image retrieval based on CNN and SVM[C]// *IEEE International Conference on Computer & Communications* 2017.
6. Guo JM, Prasetyo H (2015) Content-based image retrieval using features extracted from halftoning-based block truncation coding[J]. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society* 24(3):1010–1024
7. Lavrenko V, Manmatha R, Jeon J (2003) A model for learning the semantics of pictures[C]// *2003 Advances in Neural Information Processing Systems (NIPS 2003)*. Vancouver: NIPS foundation 1:2
8. Liang J, Liu R (2016) Stacked denoising autoencoder and dropout together to prevent overfitting in deep neural network[C]// *International Congress on Image & Signal Processing*, IEEE 213–216
9. Liao B, Xu J, Lv J et al (2015) An image retrieval method for binary images based on DBN and Softmax classifier[J]. *IETE Tech Rev* 32(4):10
10. Liu Y, Zhang D, Lu G, Ma WY (2007) A survey of content-based image retrieval with high-level semantics[J]. *Pattern Recogn* 40(1):262–282
11. Ma X, Jie G, Wang H (2015) Hyperspectral image classification via contextual deep learning[J]. *Eurasip Journal on Image & Video Processing* 32(1):20–28
12. Niu J, Bu X, Li Z, et al. (2014) An improved bilinear deep belief network algorithm for image classification[C]// *Tenth international conference on Computational Intelligence & Security*
13. Salakhutdinov R (2012) Multimodal learning with deep Boltzmann machines[J]. *J Mach Learn Res* 15(8): 1967–2006
14. Smeulders AWM, Worring M, Santini S, Gupta A, Jain R (2000) Content-based image retrieval at the end of the early[J]. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 22(12):1349–1380
15. Tang XS, Hao K, Hui W et al (2017) Using line segments to train multi-stream stacked autoencoders for image classification[J]. *Pattern Recogn Lett* 94:55–61
16. Vogel J, Schiele B (2007) Semantic modeling of natural scenes for content-based image retrieval[J]. *Int J Comput Vis* 72(2):133–157
17. Wei Y, Yang K, Yao H et al (2016) Exploiting the complementary strengths of multi-layer CNN features for image retrieval[J]. *Neurocomputing* 237:235–241

18. Xia Z, Wang X, Zhang L et al (2017) A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing[J]. *IEEE Transactions on Information Forensics & Security* 11(11): 2594–2608
19. Yang J, Yang G (2018) Modified convolutional neural network based on dropout and the stochastic gradient descent optimizer[J]. *Algorithms* 11(3):28
20. Yu J, Di H, Wei Z (2017) Unsupervised image segmentation via stacked Denoising auto-encoder and hierarchical patch indexing[J]. *Signal Process* (143):346–353

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Qinghai Wu**, was born on April of 1978, Heilongjiang, and now works in Jilin Agricultural Science and Technology University as a lecturer. His main research interest is software engineering.