

RESEARCH

Open Access



Classification methods of a small sample target object in the sky based on the higher layer visualizing feature and transfer learning deep networks

Yu Chen^{1*} , Hongbing Meng^{2*}, Xinling Wen¹, Pengge Ma¹, Yuxin Qin¹, Zhengxiang Ma¹ and Zhaoyu Liu¹

Abstract

The effective classification methods of the small target objects in the no-fly zone are of great significance to ensure safety in the no-fly zone. But, due to the differences of the color and texture for the small target objects in the sky, this may be unobvious, such as the birds, unmanned aerial vehicles (UAVs), and kites. In this paper, we introduced the higher layer visualizing feature extraction method based on the hybrid deep network model to obtain the higher layer feature through combining the Sparse Autoencoder (SAE) model, the Convolutional Neural Network (CNN) model, and the regression classifier model to classify the different types of the target object images. In addition, because the sample numbers of the small sample target objects in the sky may be not sufficient, we cannot obtain much more local features directly to realize the classification of the target objects based on the higher layer visualizing feature extraction; we introduced the transfer learning in the SAE model to gain the cross-domain higher layer local visualizing features and sent the cross-domain higher layer local visualizing features and the images of the target-domain small sample object images into the CNN model, to acquire the global visualizing features of the target objects. Experimental results have shown that the higher layer visualizing feature extraction and the transfer learning deep networks are effective for the classification of small sample target objects in the sky.

Keywords: Transfer learning, SAE, CNN, Deep network, Classification methods

1 Introduction

Due to the influence of the weather factors, camouflage, and other factors, it is often difficult to classify the target objects in the sky timely and accurately. How to effectively classify the target object is the key to defend successfully in the no-fly zone. The traditional feature extracting from the target object images based on the lower layer visualizing feature methods such as color and texture can only express the local surface of the image information, the limitation is larger, the lower layer features sometimes are difficult to fully express the global information, and the accuracy of the traditional lower layer visualizing feature extraction and classification algorithm is not high. The Convolutional Neural

Network (CNN) model [1] is a supervised learning method that achieved good application effect in many fields, but whether its performance is good or bad depends on the amount of the training sample marked. In recent years, the unsupervised feature learning of unmarked big data has become a hot topic at home and abroad [2], which simulates the intrinsic information of the human eye perception image. Through the image features transforming layer by layer, the sample feature of original space will be transformed to the new feature space, the higher layer visualizing features can be learnt, and the target detection and classification accuracy of the image are enhanced.

Now, the unsupervised learning technology [3] facing the vast unmarked data features has become the hot topic of the experts and scholars all over the world. Through simulating the human eyes to realize image scanning and perception, the most essential feature

* Correspondence: chenyu@zua.edu.cn; mhb@taru.edu.cn

¹School of Electronics and Communication Engineering, Zhengzhou University of Aeronautics, Zhengzhou, China

²College of Information Engineering, Tarim University, Tarim, China

information of data can be gotten. And through the feature transformation of the original signal, the feature of the original space can be transferred into the new feature space. The features of target image deep hierarchical [4] can be automatically learnt, which is more conducive to the target detection and classification. The Sparse Auto-Encoder (SAE) model [5] acts as a kind of unsupervised feature learning method; through the unmarked sample data rehabilitation training, the higher layer visualizing feature extraction for the target objects can be effectively generalized into the small sample target object image applications [6–8]. The SAE model does not need the prior defining features [9]; only through setting up the node numbers of hidden layer units, the hidden layer information of the target object images can be automatically learnt, and the essence of the internal correlation of the target object images can be obtained. The SAE model can automatically learn the implied relationship among the data without predefining the parameters, which is more expressive in the feature learning [10].

This paper proposed a kind of SAE higher layer visualizing feature extraction method for the small sample target objects in the sky. Firstly, the local features can be obtained through the non-transfer learning in the small sample target object images or transfer learning in the cross-domain database, and then, the global feature of the small sample target object can be obtained through the CNN model, as well as proposed to add the classification model to realize the classification of the target objects. With the help of the classification model, the different types of the target objects can be classified. Experiments verified that the algorithms this paper proposed can well classify the small sample target objects, and the classification performance comparisons between the transfer learning and non-transfer learning based on the SAE higher layer visualizing feature extraction model in the small sample target objects are realized.

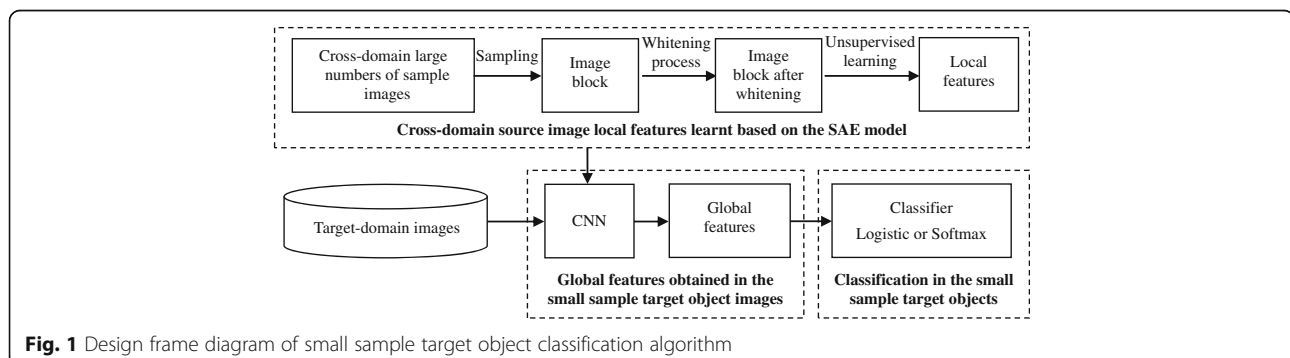
2 Algorithm design principle

The algorithms proposed in this paper consist of four modules: (1) the higher layer feature extraction module

based on the SAE model in the target-domain small sample target object, (2) the transfer learning SAE local feature extracting module in the cross-domain big data image database, (3) the global feature extracting module in the target-domain small sample target object images based on the CNN model, and (4) the classification module in the target-domain small sample target object image.

The traditional lower layer visualizing feature extraction methods usually adopt the color, texture, etc. to classify the types of the target objects. Because the lower layer feature has not enough information of the target objects, the classification accuracy of traditional algorithm is not high. Based on the SAE model, the local higher layer visualizing feature (local feature) of small sample target object images can be extracted, and then, we can transform the local feature and the target object images to the CNN model; through a continuous convoluting and pooling process, the global higher layer visualizing feature (global feature) in the training set can be extracted. Finally, the global feature can be sent into the classifier (such as logistic or Softmax regression model) to classify the small sample target objects in the test set. In this paper, during the process of the local feature extraction in the training set, because sometimes the sample number of target objects is not enough in the sky, we proposed the local features are transfer learnt from the images of the STL-10 database. The design frame diagram of the small sample target object classification algorithm proposed in this paper is shown in Fig. 1.

In Fig. 1, the higher layer visualizing features of the cross-domain images are extracted based on the SAE model; the higher layer visualizing feature extraction module is based on the unsupervised learning methods, using the back propagation (BP) training AutoEncoder (AE) Neural Network (NN). The sparse constraint is joined in the hidden layer NN to learn the typical local features of image sub-block, and the image edge information is strengthened through the whitening operation, which obtains the better visualizing features. Using the convolution point by point in the small sample target



object images by using the local feature learnt in the CNN model, the global features of the small sample target object images can be obtained. The CNN pooling operation can get the global eigenvector in the invariant rotation and scale. Finally, the global feature is sent to the classifier to realize the classification of the small sample target object images. In this paper, the method of transfer learning is proposed to reduce the shortage of the local eigenvector learning due to the small sample target objects. In the simulation experiments, the effectiveness and accuracy of the new algorithm is verified by dividing the target object image into the training set and test set.

3 Algorithm design

The algorithm this paper proposed mainly includes local feature learning, global feature obtaining, and target classification. The local feature is learnt by the SAE model, the global feature is obtained by the CNN model, and the classification realization is completed by the classifier model.

3.1 Local feature learning based on the SAE model

The SAE model this paper proposed is a kind of improved form adding the sparse constraint for the hidden layer unit response in the AE model. First, most of the neurons in the network are in inhibitory state. Then, the minimum cost function is found through the BP training method, and the key feature response of the target object is studied. In this paper, a kind of the zero-phase component analysis (ZCA) method is adopted to whiten the multiple image blocks in the target object learnt by the SAE model. Assuming that the size of the i th image block from the target object is $n \times n$, it can be sorted according to the RGB component to get $m = n \times n \times 3$ dimensional vector $x^{(i)}$. The input vector after the whitening treatment is $x^{(i)}$ and $x^{(i)} = W_{\text{white}} x^{(i)}$, where W_{white} represents $m \times m$ dimensional whitening transformation coefficient matrix. The response vectors in the SAE s th dimensional hidden layer is shown in Formula (1)

$$a^{(i)} = \sigma(Wx^{(i)} + b_1) = \sigma(W_{\text{SAE}}W_{\text{white}}x^{(i)} + b_1) \quad (1)$$

where W_{SAE} is the input weight coefficient of each image block connecting the SAE hidden layer with the whitening processing, b_1 represents the input bias, and the $\sigma(\bullet)$ is the activation function. $W = W_{\text{SAE}}W_{\text{white}}$ represents the overall weight coefficient after the whitening process (also representing the relationship between the hidden layer and the original data). After the whitening process, because the input value will exceed [0,1], the

activation function $\sigma(\bullet)$ should not be used to map the SAE output when the data is reconstructed.

$$\hat{x}^{(i)} = W_{\text{SAE}}^T a^{(i)} + b_2 \quad (2)$$

where $\hat{x}^{(i)}$ represents the i th recovery sample, W_{SAE}^T is the output weight, and b_2 is the output bias. In order to prevent overfitting and ensure the implicit response sparsely, the weight attenuation and sparse penalty term should be added in the cost function.

Acting as a kind of neural network, the SAE model is also trained in back propagation to find the minimum of the cost function. Specifically, the SAE model hopes the neural network can recover input data through training, namely $\hat{x}^{(i)} = W^{\text{white}} x^{(i)}$. Considering the constraint of weight attenuation and implicit responses' sparsity, the overall cost function can be expressed as Formula (3) [11, 12]:

$$J(W_{\text{SAE}}, b) = \frac{1}{M} \sum_{i=1}^M \frac{1}{2} \|\hat{x}^{(i)} - W_{\text{white}} x^{(i)}\|^2 + \frac{\lambda}{2} \|W_{\text{SAE}}\|^2 + \beta \sum_{j=1}^K \text{KL}(\rho \parallel \hat{\rho}_j) \quad (3)$$

where M is defined as the number of training samples, λ is the weight attenuation coefficient, β is defined as the weights of penalty term of sparse data, ρ is defined as sparse parameters, $\hat{\rho}_j$ is defined as the average activation value of the first j th unit of the hidden layer, K is defined as the numbers of hidden layer units, and $\text{KL}()$ is defined as a kind of relative entropy measurement function—Kullback Leibler (KL) divergence.

3.1.1 Whitening

Whitening operation can highlight the edge of the image information so that the deep learning algorithms can get more outstanding features. As the view of mathematics, the purpose of whitening is to remove the correlation of pixels and make it have standard covariance [13]. In practice, the whitening processing is usually combined with principal component analysis (PCA) or ZCA. In this paper, we use a kind of common ZCA whitening method to preprocess image sub-blocks.

Whitening in the RGB space usually combines the data of three color channels into one vector, which is defined as Joint Whitening (J-W). Assuming that the size of the image block is n by n , because the color image includes three channels, then the combined image sub-block data is $n \times n$, due to color image including three number channels, which becomes $N = n \times n \times 3$ dimensional vector. Before the whitening processing, the average of all samples should be subtracted to ensure that the data

mean of each location is 0. And then, the covariance matrix of $N \times N$ size is calculated [14]:

$$S = E(xx^T) = \frac{1}{m} \sum_{i=1}^m (x^i)(x^i)^T \quad (4)$$

where x^i is the i th vector combining with three number components and m is the total number of image-block. Then, the $N \times N$ size ZCA coefficient matrix W_{white} can be expressed as:

$$W_{\text{white}} = U \begin{bmatrix} \frac{1}{\sqrt{\lambda_1 + \varepsilon}} & 0 & \cdots & 0 \\ 0 & \frac{1}{\sqrt{\lambda_2 + \varepsilon}} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \frac{1}{\sqrt{\lambda_N + \varepsilon}} \end{bmatrix} U^T \quad (5)$$

where U is the eigenvector u_1, u_2, \dots, u_N of S , $\lambda_1, \lambda_2, \dots, \lambda_N$ is the eigenvalue of u_1, u_2, \dots, u_N , and ε is the regularization constant of each corresponding eigenvalue.

Performance of ε can solve the numerical instability problem caused by the small eigenvalues (even near to zero), and it can also smooth the input data to improve the learning features. Or it is the restriction of the whitening operation itself, which avoids the noise interference caused by excessive whitening. The selection of the numerical value is very important; a too large value may cause the features obscure and a too small value may lead to the appearance of noise feature values. For the SAE model, the training model based on reconstructing, the selection standard of ε is to make it more than most of the smaller eigenvalue, so as to filter out those who reflect the features of the noise in the data values. In the case that the training data has been normalized to $[0,1]$, the adjustment is usually performed from $\varepsilon = 0.01$ or $\varepsilon = 0.1$. In addition, we can visually display the data before and after the whitening process to coordinate the value of ε .

3.1.2 Local feature extraction in transfer learning

In this paper, we conduct unsupervised local feature learning on the cross-domain database and then carry out the global feature extraction and classification on the small number of sample data sets. We select three kinds of cross-domain databases such as Abstract100, Abstract 280, and STL-10 database [15, 16].

These samples are not associated with the subsequent images used for the sky target object classification. In this paper, we respectively collect 1000, 10,000, and 100,000 number 8×8 size of image blocks to carry on the feature learning by a completely random way from

the database of Abstract100, Abstract280, and STL-10. And in the final classification experiments, we repeat five times random sampling to test sample size's impact on the overall performance. The regularization coefficient was set to 0.1 in the whitening pretreatment process, and adopted 400 hidden layer units (corresponding to 400 self-learning features) in the SAE model; the training parameters were set to be the same as $\lambda = 3 \times 10^{-3}$ and the number of the training parameters $\beta = 5$ and $\rho = 0.035$. When the training samples were too small, the weights of three databases were ambiguous. With the increase of training samples, the learning effect has increased significantly based on the features of the STL-10 database; when the training sample reaches to 100,000, feature weights with a relatively clear edge can be learnt in the STL-10 database. However, the learning effect on Abstract100 and Abstract280 is not significantly improved, which indicates that it is less effective to collect a large amount of data from the small sample for unsupervised feature learning. In addition, the lower part of the feature weight is more obvious after ascending the weight according to the mAG value. That is to say, mAG value can reflect the edge of the self-learning weight performance; according to its order, the purpose of roughly dividing feature weight marginal strength can be achieved. And according to this order, we can more intuitively observe and compare the effect of learning. In the following experiment, we select the STL-10 composed of 100,000 unmarked images covering a wide range of vehicles and animals database to complete the cross-domain unsupervised learning.

3.2 Global feature extraction of CNN

Because of the too little target object image sample data, this paper introduced the transfer learning method; the local features of image sub-block large sample image data can be learnt from the cross-domain big data database sampling, making convolution operation with the current small sample target image to get feature response of the image and combining these response to get global features of target object images, which is used to realize identification and classification of the small sample target object images. The CNN global feature model structure [17] is shown in Fig. 2.

In order to improve the operational efficiency, we carry out a method of two-dimensional convolutions in three color channels during the convolutional process and sum the results. We divide every local feature SAE learnt according to three color channels and respectively carry out convolution point by point between them and the RGB components of $d \times d$ size images to get three number $(d - n + 1) \times (d - n + 1)$ size features after convolutions, and then, the global features can be obtained

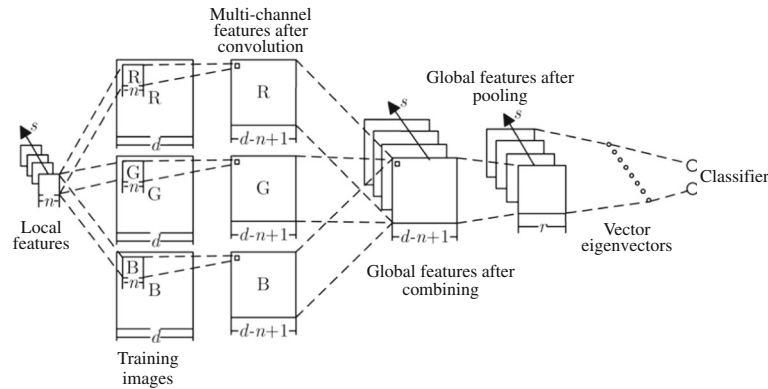


Fig. 2 CNN model structure

through adding three groups of feature images after multi-channel convolutions. In addition, to reduce dimension and avoid overfitting, CNN adopts a pooling operation to realize aggregated sampling for the previous layer network features. In this paper, we choose average pooling on the premise of whitening processing.

3.2.1 Convolutional layer

In Fig. 2, assuming that the SAE learnt K local features (K is defined as a hidden layer unit number in the SAE network) from the $n \times n$ size of the image sub-block, we make convolution between the local features and the $d \times d$ size of the target object image each pixel, which gets the size of K number $(d - n + 1) \times (d - n + 1)$ size of global features. We divide each local feature weight according to three number color channel and make convolution point by point with R, G, and B components of $d \times d$ size of target object image to obtain the $(d - n + 1) \times (d - n + 1)$ size of global feature vector. The global eigenvectors of each component is shown as Formula (6).

$$a = f(W_{SAE} W_{white} x' + b) \quad (6)$$

where $f(\cdot)$ is defined as the activation function, W_{SAE} and b are defined as local feature coefficients that the SAE model learnt, x' is defined as the value of the convolution sub-region in the training image sample, and W_{white} is defined as the ZCA whitening factor in the whitening process.

3.2.2 Pooling layer

Because the image has static attributes, the useful feature of the image is likely to also work in another area; based on this method, we can carry out the same aggregate statistics operation to the features extracted from the different areas of the convolution layer and remove the redundant features, reducing the feature resolution. This kind of aggregation statistical method is called

pooling. The pooling methods mainly include average pooling, summation pooling, maximum pooling, and l_p norm pooling. In this paper, because the convolution SAE structure adopts the white pretreatment, the average pooling is suitable, which is shown as Formula (7).

$$sj = \frac{1}{|R_j|} \sum_{i \in R_j} a_i \quad (7)$$

We choose the average pooling method, which makes some region average value in the previous layer features as a statistical representation of this region. It has the advantages of scaling and rotation invariance with the lower dimensions and prevents fitting, leading to polymerization features in space. The $(d - n + 1) \times (d - n + 1)$ size of global feature image will become $p \times p$ ($p \times p \times K$ dimension features) size of image after pooling operation.

3.3 Target object classification

To get the classification results of the small sample target objects at the output layer, in this paper, we add the regularized logistic regression model to realize classification in two kinds of image types and the Softmax regression model in the multiple kinds of image types behind the CNN model [18].

3.3.1 Logistic regression model

In the detection module of two kinds of target objects, behind the CNN model, we add the logistic regression model to realize the classification of the image second category. On the basis of linear regression, we add a logic function to realize the multivariate logistic regression model design and adopt the following Formula (8) to realize classification of $y = \{0, 1\}$.

$$\begin{cases} p(y=1|x, \theta) = \frac{1}{1+e^{-\theta^T x}} \\ p(y=0|x, \theta) = \frac{1}{1+e^{-\theta^T x}} = 1-p(y=1|x, \theta) = p(y=1|x, -\theta) \end{cases} \quad (8)$$

where $h_\theta(x) = g(\theta^T x) = \frac{1}{1+e^{-\theta^T x}}$ and $g(z) = \frac{1}{1+e^{-z}}$.

In the training data set, the feature data is expressed as $x = \{x_1, x_2, \dots, x_m\}$, and the corresponding classification data is expressed as $y = \{y_1, y_2, \dots, y_m\}$. The mostly traditional constructing method of the logistic regression model $f(\theta)$ is the maximum likelihood estimation (MLE). The posterior probability of the single sample is shown as Formula (9).

$$p(y|x, \theta) = (h_\theta(x))^y (1-h_\theta(x))^{1-y}, \quad y = \{0, 1\} \quad (9)$$

Then, the function of MLE is shown as Formula (10).

$$L(\theta|x, y) = \prod_{i=1}^m p(y^{(i)}|x^{(i)}; \theta) = \prod_{i=1}^m h_\theta(x)^{y^{(i)}} (1-h_\theta(x))^{1-y^{(i)}} \quad (10)$$

The function of the loglikelihood is shown as Formula (11).

$$\begin{aligned} \log(L(\theta|x, y)) &= \sum_{i=1}^m y^{(i)} \log(h(x^{(i)})) \\ &\quad + (1-y^{(i)}) \log(1-h(x^{(i)})) \end{aligned} \quad (11)$$

Then, the logistic regression model $f(\theta)$ is equivalent to $\theta^* = \arg \min_{\theta} (l(\theta))$, and we can adopt the gradient descent method shown as Formula (12).

$$\begin{aligned} \frac{\partial}{\partial \theta_j} (l(\theta)) &= \frac{\partial}{\partial \theta_j} \left(\sum_{i=1}^m y^{(i)} \log(h(x^{(i)})) + (1-y^{(i)}) \log(1-h(x^{(i)})) \right) \\ &= \left(\frac{y^{(i)}}{h(x^{(i)})} - (1-y^{(i)}) \frac{1}{1-h(x^{(i)})} \right) \frac{\partial}{\partial \theta_j} (h(x^{(i)})) \\ &= \left(\frac{y^{(i)}}{h(x^{(i)})} - (1-y^{(i)}) \frac{1}{1-g(\theta^T x^{(i)})} \right) \frac{\partial}{\partial \theta_j} (g(\theta^T x^{(i)})) \\ &= \left(\frac{y^{(i)}}{h(x^{(i)})} - (1-y^{(i)}) \frac{1}{1-g(\theta^T x^{(i)})} \right) g(\theta^T x^{(i)}) (1-g(\theta^T x^{(i)})) \frac{\partial \theta^T x^{(i)}}{\partial \theta_j} \\ &= \left(\frac{y^{(i)}}{h(x^{(i)})} - (1-y^{(i)}) \frac{1}{1-g(\theta^T x^{(i)})} \right) g(\theta^T x^{(i)}) (1-g(\theta^T x^{(i)})) \frac{\partial \theta^T x^{(i)}}{\partial \theta_j} \\ &= (y^{(i)} - h_\theta(x^{(i)})) x_j \end{aligned} \quad (12)$$

3.3.2 Softmax regression model

In the Softmax training set $\{(x^{(1)}, y^{(1)}), \dots, (x^{(i)}, y^{(i)}), \dots, (x^{(m)}, y^{(m)})\}$, $y^{(i)} \in \{1, 2, \dots, k\}$, k is defined as the number of target object classification, Softmax classifier [19] uses the probability $h_\theta(x)$ to carry out the classification

probability calculation of the input samples, and the function of $h_\theta(x)$ is defined as Formula (13).

$$h_\theta(x^{(i)}) = \begin{bmatrix} p(y^{(i)} = 1|x^{(i)}; \theta) \\ p(y^{(i)} = 2|x^{(i)}; \theta) \\ \vdots \\ p(y^{(i)} = k|x^{(i)}; \theta) \end{bmatrix} = \frac{1}{\sum_{j=1}^k e^{\theta_j^T x^{(i)}}} \begin{bmatrix} e^{\theta_1^T x^{(i)}} \\ e^{\theta_2^T x^{(i)}} \\ \vdots \\ e^{\theta_k^T x^{(i)}} \end{bmatrix} \quad (13)$$

where $p(y^{(i)} = j|x^{(i)})$ is defined as the probability that the input $x^{(i)}$ of the i th number sample belongs to class j . θ is defined as a model parameter, and it normalizes the probability distribution by the formula $1/\sum_{j=1}^k e^{\theta_j^T x^{(i)}}$. This moment, the cost function of system, is defined as Formula (14).

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^m \sum_{j=1}^k 1\{y^{(i)} = j\} \log \frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^k e^{\theta_l^T x^{(i)}}} \right] \quad (14)$$

where $1\{\cdot\}$ is defined as indicative function, that is, $1\{\text{expression that value is true}\} = 1, 1\{\text{expression that value is false}\} = 0$. For the minimization problem of $J(\theta)$, in this paper, we adopt the gradient descent method to solve the problem and ensure to converge into the global optimal solution. The gradient parameter is shown as Formula (15).

$$\nabla_{\theta_j} J(\theta) = -\frac{1}{m} \sum_{i=1}^m \left[x^{(i)} \left(1\{y^{(i)} = j\} - p(y^{(i)} = j|x^{(i)}; \theta) \right) \right] \quad (15)$$

We substitute $\nabla_{\theta_j} J(\theta)$ into the gradient descent method to renew the parameter, in order to gain the unique solution, and we add weight damping item to modify the cost function, which make cost function become strictly convex functions and prevent excessive parameter values during network training.

4 Experimental verification

In order to verify the validity and accuracy of the algorithm proposed in this paper, we adopt non-transfer learning algorithm and transfer learning algorithm in the higher layer visualizing feature extraction based on the SAE model and apply the classifier model to realize the classification in the small sample target object images in the sky background.

4.1 Two kinds of target object classification

Firstly, we carry out the experiment on the two types of target object, the UAVs and birds in the sky. In order to

verify the validity of higher layer visualizing feature extraction and the traditional lower layer visualizing feature extraction algorithm, we respectively take 50 number UAV target images and 50 number bird target images; parts of experimental images are shown in Fig. 3. We respectively carry out the higher layer visualizing feature extraction and lower layer visualizing feature extraction in the target object images. We adopt the color and texture of the lower layer visualizing feature to classify the different types of target object. Further, we sent the target-domain images and local feature learning through higher layer visualizing feature extraction by the SAE model into the CNN model to get the global features of target object images, and by the logistic regression classifier, we can classify the different types of the sample set. In the target object images, we select 0.2 ratios of images to act as training set and 0.8 ratios of images act as test set. During the higher layer visualizing feature learning processing, we adopt 400 number hidden layer units corresponding to 400 number self-learning features. The visualizing displays of the 400 number

higher layer local eigenvector extracted by the SAE model are shown in Fig. 4.

Normally, the algorithm performance is verified by *Precision*, *Recall*, *Accuracy*, and *F1-Measure*. The *Precision* index represents the proportion of real positive samples in the positive samples predicted by the classifier. The *Accuracy* index indicates the proportion of correct prediction of the classifier, namely the overall judgment ability of the algorithm. The *Recall* index represents the proportion of real positive samples in all positive samples [20]. And the *F1-Measurement* represents the harmonic mean between the index of Precision and Recall. In this experiment, we adopt each index as shown in Formula (16–19).

$$\text{Precision} = \frac{TP}{TP + FP} \times 100\% \quad (16)$$

$$\text{Recall} = \frac{TP}{TP + FN} \times 100\% \quad (17)$$

$$\text{Accuracy} = \frac{TP + TN}{TP + NP + TN + FN} \times 100\% \quad (18)$$



Fig. 3 Part experimental images of two kinds of target object images in the sky. The left side of the figure shows the object images of (a) Unmanned Aerial Vehicle (UAV), the right side of the figure shows the target object images of (b) Birds target object images

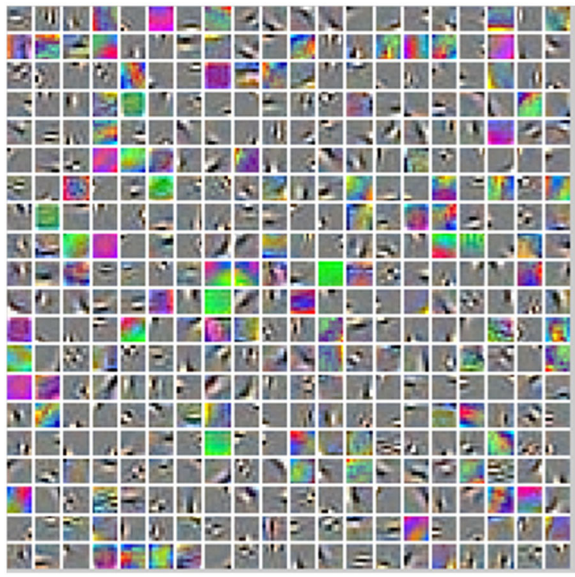


Fig. 4 The visualizing display of the higher layer local eigenvector extracted by the SAE model

$$\text{F1-Measure} = \frac{2 \times TP}{2 \times TP + FP + TN} \times 100\% \quad (19)$$

where TP is the positive sample number that predicts correct, FP is the positive sample number that predicts error, TN is the negative sample number that predicts correct, and FN is the positive sample number that predicts error. The results of classification performance in two kinds of target objects under the lower layer visualizing feature extracted and higher layer visualizing feature extracted this paper proposed are listed in Table 1.

From Table 1, we can see the classification performance by using the higher layer visualizing feature extraction based on the SAE model is better than the traditional lower layer visualizing feature extraction, such as color and texture, which proved the accuracy of the higher layer visualizing feature extraction algorithm. But, because the color and texture in the two kinds of target objects are enough, we can also get the

satisfactory classification results under the lower layer visualizing feature extraction method.

4.2 Multiple kinds of target object classification

In the classification of the multiple kinds of the target objects, especially under the small sample target object images, the performance is different between the transfer learning algorithm and the non-transfer learning algorithm. In the multiple target object classification experiments, we take three kinds of small sample target objects to carry out the classification. The three kinds of the small sample target objects respectively are the UAVs, birds, and kites. And each type includes 50 number images with the size of 64×64 ; the UAVs and birds are the same with the above experiments shown in Fig. 3, and the third type of the small target objects (kites) are shown in Fig. 5.

In addition, in order to prove effectiveness of transfer learning algorithm, this paper proposed for the small sample target object classification; we extract the local eigenvectors from the irrelevant unmarked STL-10 database. To compare the classification performance of transfer learning algorithm, we respectively get 400 numbers of 8×8 size local eigenvector visualizing expression of the STL-10 database and the small sample target object image set. The visualizing displays of the higher layer local eigenvectors extracted by the SEA model to the STL-10 database and the small sample target objects are respectively shown in Fig. 6.

The edge performance strength of the local eigenvector can reflect the similarity between the local features. From Fig. 6, we can see the local feature weight extracted from the target object images is obscure compared with the local feature weight extracted from the STL-10 image database. Due to the strength and weakness of the local eigenvector reflecting the similarity among the local features, the greater the similarity is, the better the effect of local features extraction is by the SAE model. During the experiments, we select 120 number images from the 150 number target object images to training and 30 number target object images to test; the classification performance results are obtained by the average of the five times consecutive cross-validation. We respectively send the higher layer visualizing feature extracted by the transfer learning or non-transfer learning in the training set and the target object images in the test set to the CNN model; the global features of target objects were obtained; finally, we sent the global features to the Softmax regression model to classify the target objects. We select 400 node numbers of hidden layer and the iteration number 400. Because the Accuracy index represents the whole performance ability, during the experiments, we adopt the Accuracy index to measure the performance of the classification algorithm.

Table 1 Classification performance comparative results in two kinds of different target objects

Visualizing feature extraction mode		Classification performance			
		Precision (%)	Recall (%)	Accuracy (%)	F1-Measure (%)
Lower layer visualizing feature	Color	83.33	80	80.21	86
	Texture	84.55	82	80.53	85
Higher layer visualizing feature paper proposed		91.75	84	82.15	87



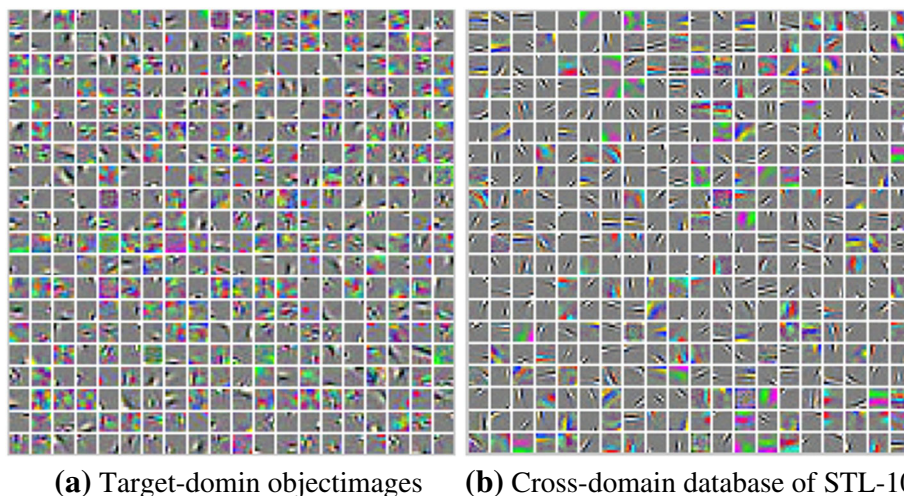
Fig. 5 The third type of the small sample target objects (Kites)

The Accuracy index of the algorithm in the small sample target object classification is shown in Table 2.

As shown in Table 2, the performance of the classification algorithm based on the higher layer feature extraction is obviously better than the traditional lower layer visualizing feature extraction, such as the features of color histogram, LBP, and GIST. And the classification performance based on the transfer learning higher layer visualizing feature extraction algorithm is a little bit better than the non-transfer learning higher layer feature extraction algorithm. The result indicates that the transfer learning mode can also effectively make up for the lack of training process under the small sample target object images. The classification performance did not fall during adoption of the transfer learning algorithm, which effectively avoids the classification performance degradation or even failure under the lack of target data

samples and the traditional lower layer visualizing feature extraction algorithm. In order to verify the classification performance impact of the transfer learning SAE algorithm under the changing of node numbers of a hidden layer, we respectively select the node numbers 50, 100, 150, 200, 250, 300, 350, and 400 of the hidden layers, then send the local features learnt from the non-transfer learning and transfer learning in the small sample target object images to the CNN model. The classification performance results under the different node numbers of the hidden layer are shown in Fig. 7 through the Softmax regression model classification.

From Fig. 7, we can see with the increasing of the node numbers of the hidden layer, at the beginning, the classification performance of algorithm obviously improves; when the node number is over 200, the



(a) Target-domain object images **(b)** Cross-domain database of STL-10

Fig. 6 The visualizing displays of the higher layer local eigenvectors extracted by the SAE model. The left side of the figure shows the higher layer local eigenvectors extracted by the SAE model from **(a)** The target-domain object images, and the right side of the figure shows the higher layer local eigenvectors extracted by the SAE model from **(b)** The cross-domain database of STL-10

Table 2 Average classification performance comparison under the different feature extraction modes

Different feature extraction modes		Classification performance Accuracy (%)
Lower layer visualizing feature	Color histogram	48.67
	LBP	68
	GIST	74
Higher layer visualizing feature	Non-transfer learning	91.33
	Transfer learning	90.67

classification performance enhancements of the algorithms are not obvious. In addition, the iterations in the algorithm have little affection for the classification performance, and when the iterations are bigger than 30, it has little effect to the performance of the algorithm when the numbers of iterations continue to increase.

4.3 Analysis of classification results

During the non-transfer learning higher layer visualizing feature extraction process, we chose the node number 400 of the hidden layer in the SAE model and the iterations 400, the training parameters were set to be the same as $\lambda = 3 \times 10^{-3}$, and the number of the training parameters $\beta = 5$ and $\rho = 0.035$ during the SAE higher layer visualizing feature training process. We analyze one time classification experimental results by using the Softmax classifier, the classification results respectively listed in Table 3. From the results, we can see the non-

Table 3 The continuous five times classification results

Experimental numerical order (no.)	Accuracy (%)	
	Non-transfer learning	Transfer learning
1	96.667	96.667
2	90	90
3	93.33	93.33
4	96.667	93.33
5	80	80
Average	91.33	90.67

The total numbers of target object images are 150 pictures, and the sample numbers are all 50 pictures in each type of UAVs, birds, and kites. The training set has 120 pictures of target object images, and the test set has 30 pictures of target object images

transfer learning algorithm and transfer learning algorithm also gain the higher precision. Because the sample numbers are enough, the classification precision under the SAE higher layer visualizing feature extraction based on non-transfer learning is slightly higher than transfer learning.

When the sample numbers of the target object images are not enough, for example, the sample numbers of three types are respectively 20, the error classification results of the multi-target object images during one experiment are shown in Table 4. From Table 4, we can see the classification performance based on the non-transfer learning algorithm cannot efficiently reach to a certain classification precision. But the classification

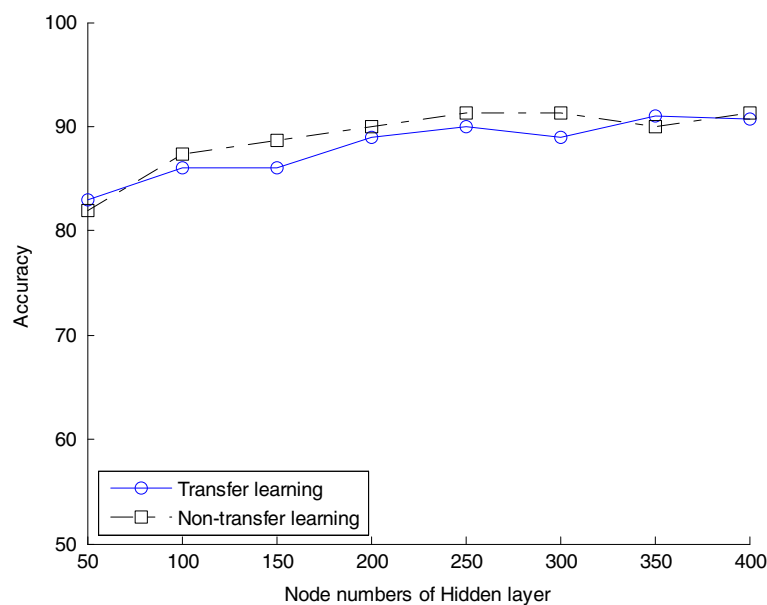




**Fig. 7** Classification performance results under the different node numbers of hidden layer

Table 4 Classification results in one time

Higher layer visualizing feature extraction mode	Accuracy (%)	Classification error images (Fact types→ Classification error types)		
Non-Transfer learning	70			
		(Bird→kite)	(Bird→Kite)	(UAV→Bird)
Transfer learning	90			
		(Bird->kite)		

The total numbers of target object images are 60 pictures, and the sample numbers are all 20 pictures in each type of UAVs, birds, and kites. The training set has 50 pictures of target object images, and the test set has 10 pictures of target object images

performance can also reach satisfied results with higher precision under the SAE transfer learning model. So, we can use the transfer learning algorithm based on the SAE model to classify the target objects under the insufficient sample numbers.

5 Conclusions

In this paper, firstly, we proposed a kind of higher layer visualizing feature extraction algorithm based on the SAE model; then, we used the transfer learning methods to realize the global higher layer feature obtained from the small sample target objects in the sky based on the CNN model. Finally, we adopted the logistic regression classifier or the Softmax regression classifier to realize the classification in the second category classification or multiple classifications to the target objects in the sky, such as UAV, birds, and kites. Experiments verified the effectiveness and accuracy of the methods this paper proposed. Further, we will discuss the higher layer visualizing feature combining the lower layer visualizing features, such as color and texture, to find the most suitable methods to classify the small sample target objects in the complex sky background.

Abbreviations

BP: Back propagation; CNN: Convolutional Neural Network; J-W: Joint Whitening; KL: Kullback Leibler; MLE: Maximum likelihood estimation; NN: Neural network; PCA: Principal component analysis; SAE: Sparse AutoEncoder; UAV: Unmanned aerial vehicle; ZCA: Zero-phase component analysis

Acknowledgements

The research presented in this paper was supported by Zhengzhou University of Aeronautics, China.

Funding

This paper is supported by the Science and Technology Innovation Team of Henan Province Supports Project (Grant: 17IRTSTHN014), Science and Technology Project of Henan Province (Grant 182102210110 and

182102210111), and Key Scientific Research Projects in Henan Province (Grant 18A510018 and 18A510019).

Authors' contributions

YC, HM, and XW contributed to the conception and design of the study. YC, XW, ZM, and YQ worked on the data acquisition. YC, XW, and HM contributed to the data analysis and interpretation of the data. YC, ZM and PM contributed to the drafting the manuscript and gave the final approval of the version to be published. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 7 February 2018 Accepted: 26 April 2018

Published online: 21 May 2018

References

1. A Krizhevsky, I Sutskever, GE Hinton, ImageNet classification with deep convolutional neural networks. *Int. Conf. Neural Inf. Process. Syst. Curran Assoc. Inc* **60**(2), 1097–1105 (2012). <https://doi.org/10.1145/3065386>
2. Y Bengio, A Clurville, P Vincent, Representation learning: a review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(8), 1798–1828 (2013). <https://doi.org/10.1109/TPAMI.2013.50>
3. Y Lecun, Y Bengio, G Hinton, Deep learning. *Nature* **521**(7553), 436–444 (2015). <https://doi.org/10.1038/nature14539>
4. J Schmidhuber, Deep learning in neural networks: an overview. *Neural Netw.* **61**, 85–117 (2015). <https://doi.org/10.1016/j.neunet.2014.09.003>
5. J Masci, U Meier, C Dan, J Schmidhuber, in *Proceedings of the 21st International Conference on Artificial Neural Networks, Espoo*, 6791. Stacked convolutional auto-encoders for hierarchical feature extraction (2011), pp. 52–59. https://doi.org/10.1007/978-3-642-21735-7_7
6. F Zhang, B Du, L Zhang, Saliency-guided unsupervised feature learning for scene classification. *IEEE Trans. Geosci. Remote Sens.* **53**(4), 2175–2184 (2015). <https://doi.org/10.1109/TGRS.2014.2357078>
7. Y Bengio, Learning deep architectures for AI. *Found. Trends Mach. Learn.* **2**(1), 1–127 (2009). <https://doi.org/10.1561/2200000006>
8. P Sermanet, K Kavukcuoglu, S Chintala, Y Lecun, in *Proceedings of Computer Vision and Pattern Detection (CVPR)*, Portland. Pedestrian detection with unsupervised multi-stage feature learning (2012), pp. 3626–3633. <https://doi.org/10.1109/CVPR.2013.465>

9. H Yin, X Jiao, Y Chai, B Fang, Scene classification based on single-layer SAE and SVM. *Expert Syst. Appl.* **42**(7), 3368–3380 (2015). <https://doi.org/10.1016/j.eswa.2014.11.069>
10. Liu H, Taniguchi T, Takano T, Tanaka Y. Visualization of driving behavior using deep sparse autoencoder. *Proceedings of the 2014 IEEE Intelligent Vehicles Symposium, Dearborn: 1427-1434* (2014). doi: <https://doi.org/10.1109/IVS.2014.6856506>
11. Z Li, Y Fan, W Liu, The effect of whitening transformation on pooling operations in convolutional autoencoders. *EURASIP J. Adv. Signal Process.* **2015**(1), 37 (2015). <https://doi.org/10.1186/s13634-015-0222-1>
12. E Othman, Y Bazi, N Alajlan, H Alhichri, F Melgani, Using convolutional features and a sparse autoencoder for land-use scene classification. *Int. J. Remote Sens.* **37**(10), 2149–2167 (2016). <https://doi.org/10.1080/01431161.2016.1171928>
13. R Wang, L Du, Z Yu, W Wan, in *2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), San Jose, USA*. Infrared and visible images fusion using compressed sensing based on average gradient (2013), pp. 1–4. <https://doi.org/10.1109/ICMEW.2013.6618257>
14. AJ Bell, TJ Sejnowski, in *Proceedings of the 10th Annual Conference on Neural Information Processing Systems (NIPS), Denver*. Edges are the “independent components” of natural scenes (1997), pp. 831–837
15. H Zhang, Z Yang, M Gönen, M Koskela, J Laaksonen, T Honkela, E Oja, Affective abstract image classification and retrieval using multiple kernel learning. *Int. Conf. Neural Inf. Process.* **8228**, 166–175 (2013). https://doi.org/10.1007/978-3-642-42051-1_22
16. He Zhang, Eimontas Augilius, Timo Honkela, Jorma Laaksonen, Hannes Gamper, and Henok Alene. Analyzing emotional semantics of abstract art using low-level image features. In *Proceedings of 10th International Symposium on Intelligent Data Analysis (IDA 2011)*. Springer, 2011. https://doi.org/10.1007/978-3-642-24800-9_38
17. DC Ciresan, U Meier, J Masci, LM Gambardella, J Schmidhuber, in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona*. Flexible, high performance convolutional neural networks for image classification (2011), pp. 1237–1242. <https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-210>
18. R Zeng, J Wu, Z Shao, L Senhadji, S Huazhong, Quaternion softmax classifier. *Electron. Lett.* **50**(25), 1929–1930 (2014). <https://doi.org/10.1049/el.2014.2526>
19. A Coates, H Lee, AY Ng, in *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics, Ft. Lauderdale, USA*. An analysis of single-layer networks in unsupervised feature learning (2011), pp. 215–223
20. WX Mao, ZM Cai, L Tong, Malware detection method based on active learning. *J. Softw.* **28**(2), 384–397 (2017). <https://doi.org/10.13328/j.cnki.jos.005061>

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)