IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON DATA MINING AND GRANULAR COMPUTING IN BIG DATA AND KNOWLEDGE PROCESSING

Received July 31, 2018, accepted September 3, 2018, date of publication September 17, 2018, date of current version October 12, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2869976

Architectural Style Classification Based on Feature Extraction Module

PEIPEI ZHAO, QIGUANG MIAO, JIANFENG SONG, YUTAO QI, RUYI LIU, AND DAOHUI GE

[1]School of Computer Science and Technology, Xidian University, Xi'an 710071, China
[2]Xi'an Key Laboratory of Big Data and Intelligent Vision, Xi'an 710071, China

Corresponding author: Jianfeng Song (jfsong@mail.xidian.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC0807500, in part by the National Natural Science Foundations of China under Grant 61772396, Grant 61472302, and Grant 61772392, in part by the Fundamental Research Funds for the Central Universities under Grant JB170306, Grant JB170304, and Grant JBF180301, and in part by the Xi'an Key Laboratory of Big Data and Intelligent Vision under Grant 201805053ZD4CG37.

ABSTRACT Standard classification tasks have already achieved good results in computer vision. However, the task of Architectural style classification yet faces many challenges, since the rich inter-class relationships between different styles may disturb the classification accuracy. To better classify buildings, we propose a feature extraction module based on image preprocessed with Deformable Part-based Models (DPM). Specifically, we first use DPM to remove elements that are not related to classification, and capture representative elements of buildings, and then these elements are sent to our feature extraction module. In our feature extraction module, we adopt our improved ensemble projection method to maximize the inter-class distance and minimize the intra-class distance to find the common features in the same style and differences among different styles. Finally, the performances of several classifiers are tested and the best one of SVM classifier is selected to output the ultimate accuracy. Experimental results show that our approach achieves promising performance and is superior to previous methods.

INDEX TERMS Architectural style classification, deformable part-based Models (DPM), feature extraction module, improved ensemble projection (IEP), SVM classifier.

I. INTRODUCTION

Architectural style classification, of which the purpose is to classify buildings by some algorithms, is of great importance in the development of a region. The generation of architectural styles evolves as a gradual process, where characteristics of the same classes exist differences at different times. It reflects the cultural development of a region.

Although architectural style classification seems just to be a classification problems, there are many challenges still associated with accuracy of it. First of all, the generation of architectural styles evolves as a gradual process. When styles from one region to another, each region has its own unique characteristics. Meanwhile, each building is unique due the personalities of different architects. Therefore, it is a challenge to find common features within a style. Secondly, when designing a building, an architect sometimes integrates several different architectural style elements. Thence, there are similar characteristics between different styles. As shown in Fig.1, the building in the bottom left corner consists of a chimney, whereas the building in the top right corner does not. They belong to the same architectural style, i.e. American craftsman style, but they have different elements. However, the buildings in the first row belong to different architectural styles, they have the same element of a triangular roof. These complicated relationships between architectural styles lead to some difficulties in classification. So it is significant for architectural styles to find common features of the same style and differences among the 25-class architectural styles.

In this paper, we propose a feature extraction module based on image preprocessed with DPM [1]. To learn the details of building better, we first conduct a preprocessing that using DPM to extract representative elements of buildings. Subsequently, the elements of these images are sent to a feature extraction module. The module consists of depth feature extraction [2], [3] model and IEP model. We use the first model to learn high-level semantic features. To find the common features in the same style and differences among different styles, we adopt our IEP model to maximize the inter-class distance and minimize the intra-class distance. After a comparison among various classifiers, the final result

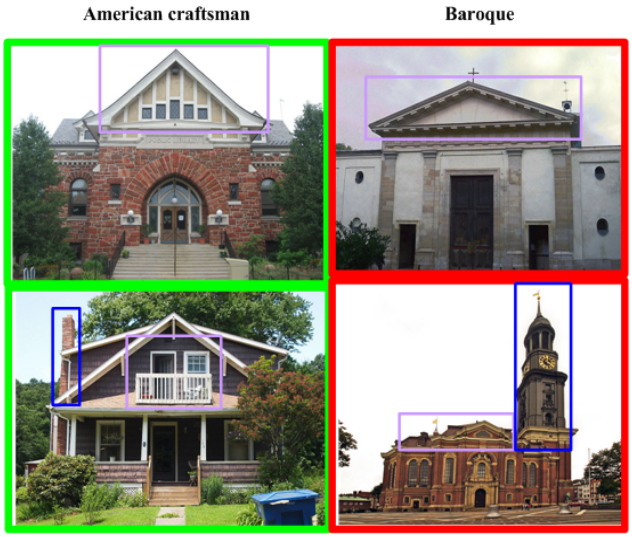


FIGURE 1. Relationships between architectural styles. There are two categories, each column as a category. The first column is American craftsman and the second column is Baroque. Purple bounding boxes in each row show the similarity between different classes. The roofs of these two classes are triangle. The blue's in each column represent the differences in the same class. For example, image in the bottom right corner has a tower. However, there isn't tower in the image at the top right.

is obtained with the classifier which has the best performance. The main contributions of this paper are as follows:

- Image preprocessing with DPM Models: Not all elements in the image are favorable for classification. Sometimes, only a few representative elements work on it. Therefore, it is very important to find these representative characteristics. In this paper, DPM can find representative characteristics in an image by matching it to root filter and part filters.
- The feature extraction model: Since it is a challenging to find common features within a style and differences among different architectural styles. We adopt our feature extraction module to minimize the intra-class distance and maximize the inter-class distance. The module is based on the local-consistency and the exotic-inconsistency assumptions. Thus it can capture the common characteristics of the same style and differences among different styles. The experiments prove that our extraction model is beneficial for boosting the final performance to a large extent.
- The analysis of different classifiers: We analysis two classifiers of SVM [4], [5] and LR [6] by groups of experiments, and give the final result with the classifier having best performance.

The remainder of this paper is organized as follows. The development of Architectural style classification is briefly reviewed in Section II. Following in section III, image preprocessing with DPM Models and our feature extraction module with deep neural networks (DNN) and IEP method are discussed. Subsequently, extensive experiments in section IV demonstrate the effectiveness of our approach. Finally, our approach is concluded and the work for future is given in the last section.

II. RELATED WORK

Recent research in architectural style classification has obtained some success. There are various approaches to handle architectural style classification. In the early stage, providing efficient solutions to architectural style classification have a major focus on extracting elements or patterns [7]–[12]. Alexander C. Berg. [7] addressed image parsing in the setting of architectural scenes by the generic recognition results bootstrap an image specific model. They approached parsing as a recognition problem both at the coarse level of street, foliage, building, sky, and at the detailed level of window, door, etc. Chu and Tsai [8] devised a higher-level feature representation to describe configurations of repetitive elements. The feature representation was more discriminative than visual word by modeling spatial relationships between local features, and was flexible to tackle with object scaling, rotation, and viewpoint changes. Doersch *et al.* [9] proposed to use a discriminative clustering approach able to take into account the weak geographic supervision. It automatically found visual elements, such as balconies, street signs and windows, that were most distinctive for a certain geo-spatial area. Meanwhile, the approach could find out which of them are both frequently occurring and geographically informative in all possible patches in all images. Philbin *et al.* [11] introduced a novel quantization method based on randomized trees to address a major time and performance bottleneck. One recent study [12] proposed a method that was based on Deformable Part-based Models (DPM) and Multinomial Latent Logistic Regression (MLLR). DPM could capture morphological characteristics of basic architectural components. MLLR introduced the probabilistic analysis and tackled the multi-class problem in latent variable models.

With the rapid development of deep learning and powerful hardwares like GPU, a series of successes have been achieved with the approaches based on Convolutional Neural Network (CNN) for visual task. The concept of CNN is known as LeNet5 model due to its inventor [13]. However, it has not gone further because of the limitation of hardware at that time. A few years later, Hinton and Salakhutdinov [14] proposed a detailed implementation of deep learning, and introduced the model of deep belief network (DBN), which made it have a number of stacked restricted Boltzmann machines (RBM). A significant application of deep CNN is the AlexNet model [15]. It achieves a great success on the ImageNet competition with 10% higher accuracy than other state-of-the-art methods in 2012. Then, a series of models have been proposed for computer vision tasks such as ZF-Net [16], Deepval-Net [17], network in network [18], and so on. On the other hand, deeper and more sophisticated CNN models have made significant progress by increasing the number of layers [18], size of layers [19] and better

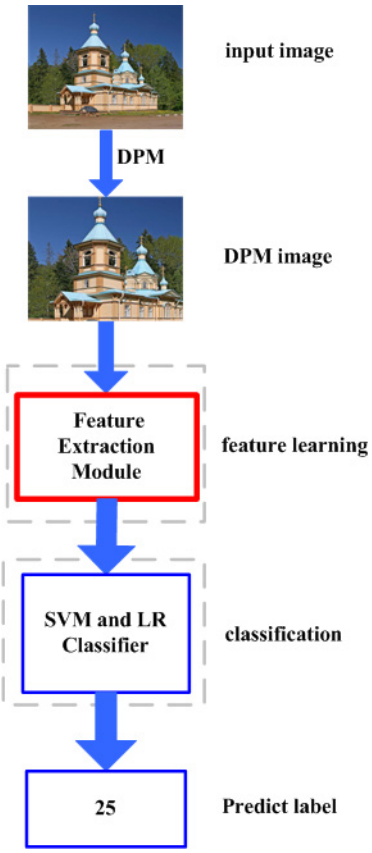


FIGURE 2. The structure of our approach.

activation function, e.g., Relu [20] to yield the best results on various challenges related to object classification, recognition and computer vision. In the 2014 ILSVRC [21] classification challenge, GoogLeNet [2] and VGGNet [22] yielded similarly high performance. These successes spurred a series of research that focused on finding higher performing convolutional neural networks for a task.

III. THE CLASSIFICATION OF ARCHITECTURAL STYLE WITH OUR FEATURE EXTRACTION MODULE

As mentioned by Xu *et al.* [12], architectural style classification has many difficulties in feature extraction. It is a challenge to find common features within a style and highlight the specific design of an individual building. It is another challenge to find different features among different styles. In this paper, we propose a method which can extract common and different features with our feature extraction model. Firstly, to learn better features, we use DPM to capture the morphological characteristics of basic architectural components. Then the features of these images are extracted respectively by our feature extraction model. The model is based on the local-consistency and the exotic-inconsistency assumptions. It captures not only the characteristics of individual images, but also the relationships among images [23]. Finally, we test the performance of several classifiers and the extracted feature is sent to the optimal classifiers - a linear SVM classifier and LR classifier for the finally results. The flowchart of our approach is depicted in Fig.2. The details of Deformable Parts Model, our feature extraction module implementation will be introduced in following subsections.

A. IMAGE PREPROCESSING STRATEGY

Each type of building has its own style. As shown in Fig. 3, there is a line in the middle of Chicago style buildings; gothic buildings have towering spires, arches and rose windows. Therefore, we only need these representative elements to classify, instead of all elements. To this end, we employ DPM model to extract such typical elements in images.

FIGURE 3. Images in chicago and gothic styles.

DPM describes an image by a multi-scale HOG feature pyramid [1]. The model is defined by a coarse root filter, a set of part filters and deformation costs. The root filter approximately captures the outline of the object such as the building boundary. The part filters are applied to the image with twice the resolution of the root, capturing finer resolution features of the object such as arches and rose windows. Deformation cost measures the deviation of the parts from their default locations relative to the root. We use labeled data to train models for each style. Fig.4 shows a trained model for Russian Style.

We use the trained models to detect representative elements in an image. A bounding box slides over the image to form a number of sub-images. In order to improve the resolution, we first conduct an up-sampling that converting sub-image into twice size with Gaussian pyramid [24]. Following, features of original sub-image and sub-image with enlarged resolution are extracted respectively by the DPM. Then the DPM features of original sub-image convolves with root filter to obtain the response diagram, which represents the match between image and root filter. Meanwhile, in order to get the response diagrams of the match between the sub-image and part filters, the DPM features of sub-image with enlarged resolution convolves with part filters. The response diagram of root filter can capture coarse resolution edges such as

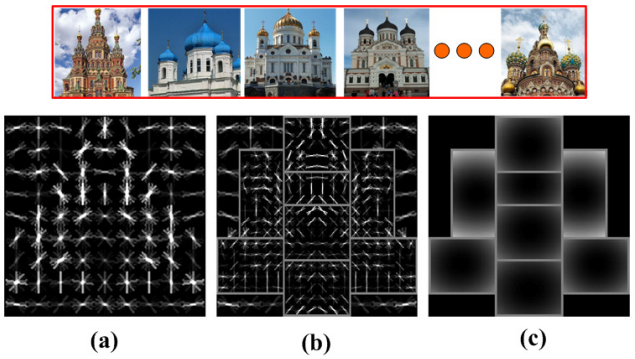


FIGURE 4. The model for Russian Style. The first line is the dataset in Russian style. The trained root filter and part filters for Russian Style are shown in (a) and (b). The root filter shows typical facade outline of Russian style buildings. The part filters captures discriminative architectural elements such as the full and round dome. (c) shows the deformation cost.

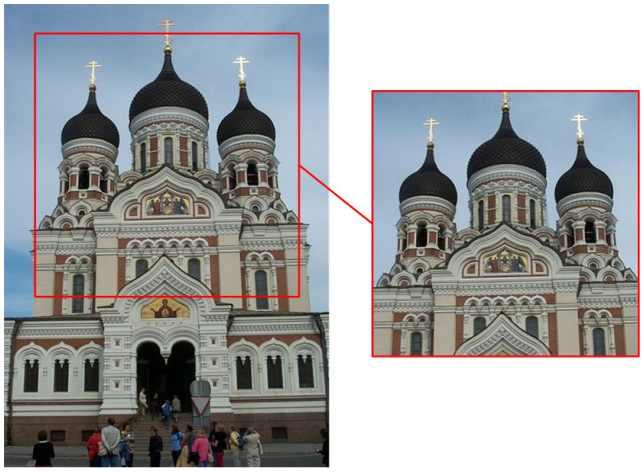


FIGURE 5. The image is decomposed into many bounding boxes of the same size. Red bounding box has the highest response. It is used to replace this image.

the structure of the building, but details like windows and doors are not visible. The response diagrams of part filters can captures the details. As these response diagrams can complement each other, these features are blended based on weighted average to obtain the final response diagram. After a comparison among the final response diagrams of all sub-images, an image is instead of the sub-image which has the highest response. As shown in Fig.5, the image is replaced by the sub-image of the highest response. The sub-image is generated from DPM model, which focuses on the representative elements, namely the full and round dome in Fig.5, and it is helpful to eliminate the extra elements, so it can boost the performance.

B. FEATURE EXTRACTION MODULE

The feature extraction module consists of DNN model and IEP model. Our first model can capture high-level features. Our IEP model can extract the common characteristics of the same style and differences among different styles.

1) DEPTH FEATURE EXTRACTION

The main advantage of CNN [25]–[31] is the ability to learn high-level features from the low-level ones and the details which are not related with the target can be ignored. Meanwhile, CNN is robust against light, surrounding clutter and rigid transformation [32], therefore the CNN based method can achieve superior performance on architectural style classification. In this paper, we use GoogLeNet model to extract features that are beneficial to architectural style classification. Although GoogLeNet and VGGNet [22] yielded similarly high performance in architectural style classification. However, VGGNet has the compelling feature of architectural simplicity, this comes at a high cost: evaluating the network requires a lot of computation [3]. GoogLeNet was designed with computational efficiency and practicality in mind, so that the computational cost of GoogLeNet is much lower than VGGNet.

GoogleNet has a new level of organization called "Inception Module" which consists of convolutions and max-pooling operation. There are nine Inception modules in GoogLeNet architecture. Fully-connected layers are being replaced with 1×1 convolutions at the bottom of the module. The 1×1 convolutions can reduce the number of inputs and hence decreases the computation cost dramatically. It also extracts the relevant features of an input image in the same region.

However, training a deep network needs to collect a large amount of labeled data since there are millions of parameters waiting for adjusting. In architectural style classification, it is expensive to recollect the needed training data. In such cases, a pre-trained GoogLeNet model has been applied and fine-tuned using our dataset in architectural style classification. In order to train the model, we use data augmentation in this paper. The number of images can be increased by horizontal/vertical flip. Therefore, data augmentation can prevent the model from over-fitting.

FIGURE 6. The similar styles. The image in the red bounding box that wrongly classified into Greek Revival style. However, it really belongs to Georgian style.

2) FEATURE EXTRACTION MODEL WITH IEP

After an analysis of the wrongly classified images of the GoogLeNet model, we find one factor that hampers the

accuracy is the similarities between different styles and the differences in the same styles. As shown in Fig.6, the image with red edges belongs to the Georgian style, but it is incorrectly classified into the Greek Revival style. The misclassified image has some similar features to the Greek Revival style. For example, they all have some cylinders. To distinguish the image from the Greek Revival style, we must discover differences between the image and the style. However, it is not enough just to find the differences. In order to classify the image correctly, it is also important to find out what it has in commons with the Georgian style.

We propose an improved ensemble projection method (IEP) based on ensemble projection (EP) [34] to learn a new image representation which can solve above problems. IEP exploits the local-consistency assumption that samples with high similarity should share the same label and the exotic-inconsistency assumption that samples with low similarity are in high probability come from different classes. IEP consists of many prototypes which are inter-distinct and intra-compact, so that each one represents a different visual concept. To ensure inter-distinct and intra-compact of each prototype, we employ a two-step sampling method, called Max-Min Sampling. The Max step is designed for the inter-distinct property which depicts the differences among different styles. The Min step is designed for intra-compact property which depicts the common characteristics of the same style.

Algorithm 1 Max-Min Sampling in t^{th} Trail

Data: Dataset D
Result: Prototype set P^t
begin
  $e_1 = 0$;
  While iterations $\leq m$ **do**
    $v = \{\text{r random image indexes}\}$;
    $e = \sum_{i \in v} \sum_{j \in v} dis\left(X_i, X_j\right)$
    if $e > e_1$ **then**
      $e_1 = e$
      $v_1 = v$
    end
  end
  for $i \leftarrow 1$ **to** r **do**
    $s_i^t =$ indexes of the n nearest neighbors of $v\,(i)$ in D;
    $c_i^t = (i, i, \ldots, i) \in \mathbb{R}^n$
  end
  $s^t = \left(s_1^t, \ldots, s_r^t\right) \in \mathbb{R}^{rn}$
  $c^t = \left(c_1^t, \ldots, c_r^t\right) \in \mathbb{R}^{rn}$
  $P^t = \left\{\left(s_i^t, c_i^t\right)\right\}_{i=1}^{rn}$
end

The algorithm for creating prototype sets is given in Algo.1. In particular, we first build a skeleton of the prototype set by looking for images with the large distances from each other. The distance of two images can be expressed as:

$$dis(x_i, x_j) = \sum_{k=1}^{n} (x_{ik} - x_{jk})^2 \tag{1}$$

Where $dis(x_i, x_j)$ is the square of a distance between x_i and x_j. They are the features of two different images, which are extracted by GoogLeNet. i and j are image indexes. $x_i = \{x_{i1}, x_{i2}, \ldots, x_{in}\}$, $x_j = \{x_{j1}, x_{j2}, \ldots, x_{jn}\}$. Therefore, the Max step guarantees that the sampled seed images are far from each other, which means it can find differences among different classes. Once the skeleton is created, we enrich the skeleton to a prototype set by looking for the closest neighbors of the skeleton images. In other words, the min step can extend each seed image to an image prototype by introducing its n closest images (including itself), which means it can extract the common characteristics of the same class. A single prototype set only defines a visual concept (image attribute). For large diversity, randomness is introduced in different trails of Max-Min Sampling to create an ensemble of diverse prototype sets, so that a rich set of image attributes are captured [23].

After the prototype sets are established, input the image X to the prototype sets to measure similarities. The vector of all similarities is concatenated to form a new image representation which is used for the final classification.

IV. EXPERIMENT RESULTS

In this section, we demonstrate how our strategies work by groups of experiments. Firstly, we discuss implementation details, including the architectural style dataset that our experiments process on, the parameter setting and running environment in Section A, and then the experiments that show the effect of image preprocessing with the Deformable Parts Model, the analysis of our feature extraction module and the comparison of the performance of different classifiers are given in Section B to D. For a fair comparison, the experiments on our proposed method are conducted on the same set with [12]. We run a ten-fold experiment.

A. IMPLEMENTATION DETAILS

1) DATASET

In order to study architecture styles and model their underlying relationships, we use the architectural style dataset [12], [33]. The dataset contains 25 architecture styles. The number of images in each style varies from 50 to 300, and altogether the dataset contains about 5000 images.

2) PARAMETER SETTING

As the model we use for feature extraction is GoogLeNet and IEP, these network settings are the same as [3] and [34]. However, for a better fine-tuning result and adapting to our architectural style classification, we use batches of 100, with the learning rate of 0.5 in the GoogLeNet. As to the parameters of our IEP method, we used the following for experimental sets: T = 300, r = 30, n = 6 and m = 50.

3) EXPERIMENTAL ENVIRONMENT

Our experiments are processed on a PC with Intel Core i5-4590 CPU @ 3.30GHz 3.30 GHz, 8 GB RAM and Nvidia Tesla K40c GPU. The experiments of the GoogLeNet model training and feature extracting are processed under tensorflow framework on Linux Ubuntu 14.04 LTS, others including image preprocessing with DPM, learning a new image representation with IEP and the final classification process with various classifiers are implemented by matlab R2014a on 64-bit Windows 7.

B. EFFECT OF IMAGE PREPROCESSING STRATEGY

In this subsection, we experiment on 25-class architectural style dataset and verify the effectiveness of our image preprocessing strategy. We compare the accuracy between classification results with preprocessed and original input. The results are shown in Fig.7.

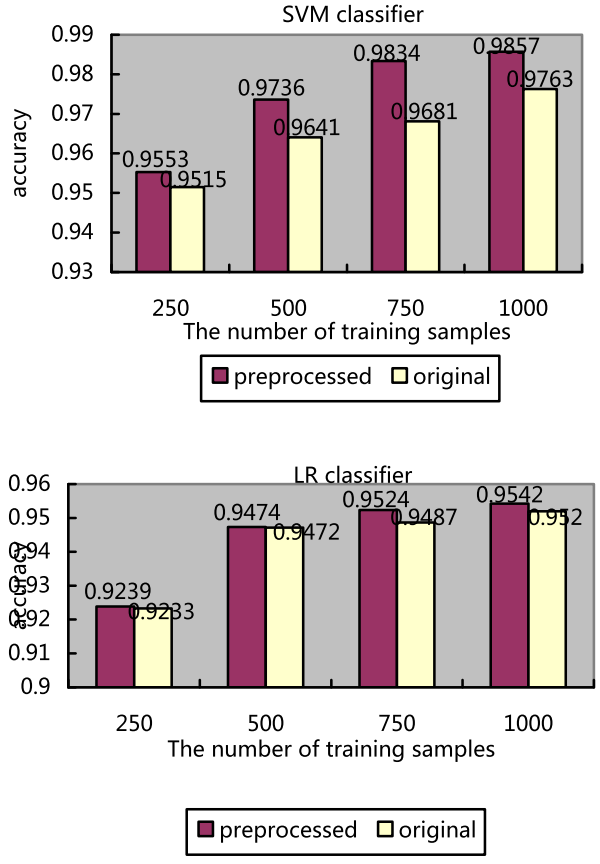


FIGURE 7. In this paper, SVM and LR are used to classify 25-class architectural style dataset. Result comparison between our feature extraction model with preprocessed and original input. The preprocessed input is superior to another on 25-class architectural style dataset.

As can be observed in Fig.7, our image preprocessing strategy is reasonable and achieves a great promotion. With the number of preprocessed training samples in each category increasing, the accuracy rate also increases. That is because the image preprocessing strategy can capture the representative elements which can illustrate the building better.

C. THE ANALYSIS OF OUR FEATURE EXTRACTION MODULE

After the correctness of our image preprocessing strategy has been proved, we verify the effect our feature extraction module. The verification of feature extraction module is in two-stage. Our feature extraction module consists of GoogLeNet and IEP. Firstly, we test the influence of high-level features with GoogLeNet, i.e., the differences between preprocessed images with/without GoogLeNet. The experiments on preprocessed images are illustrated in Fig.8.

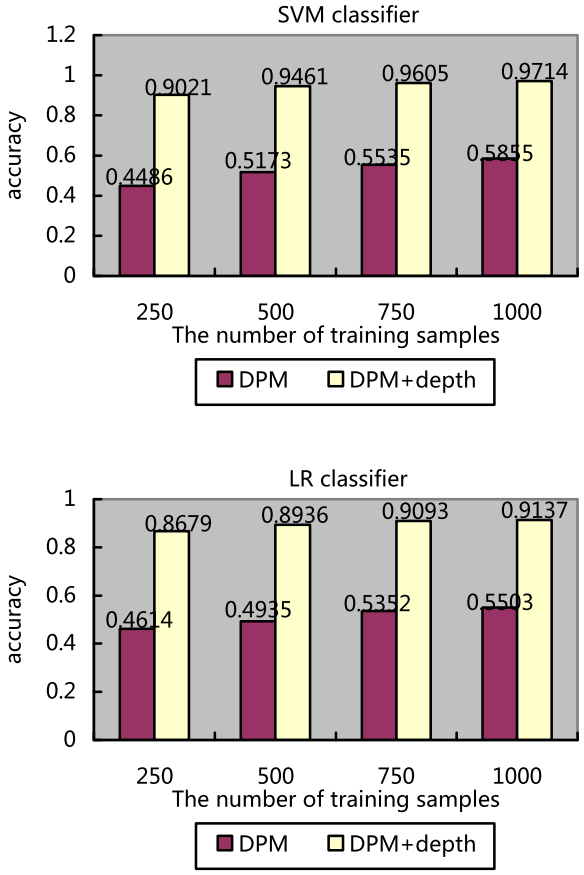


FIGURE 8. Result comparison between the accuracy of high-level feature extraction with GoogLeNet or without it. It can be find that the depth feature extraction helps to improve the classification performance.

Fig.8 reports that the depth feature extraction can really help the preprocessed image to achieve higher accuracy. The improvement on preprocessed images is about 40%. That is because the depth data helps to filter out building-irrelevant factors and it can mining high-level feature.

Then, in the Fig.9, we can see the IEP data is still useful for the improvement of accuracy overall. Although the participation of depth data and preprocessed images can provide more information as what the IEP data does, the addition of IEP data can still present a positive influence to improving the classification accuracy.

As can be observed in Fig.10, our feature extraction module helps to achieve higher accuracy. That is because the depth data can build high-level features from the low-level ones

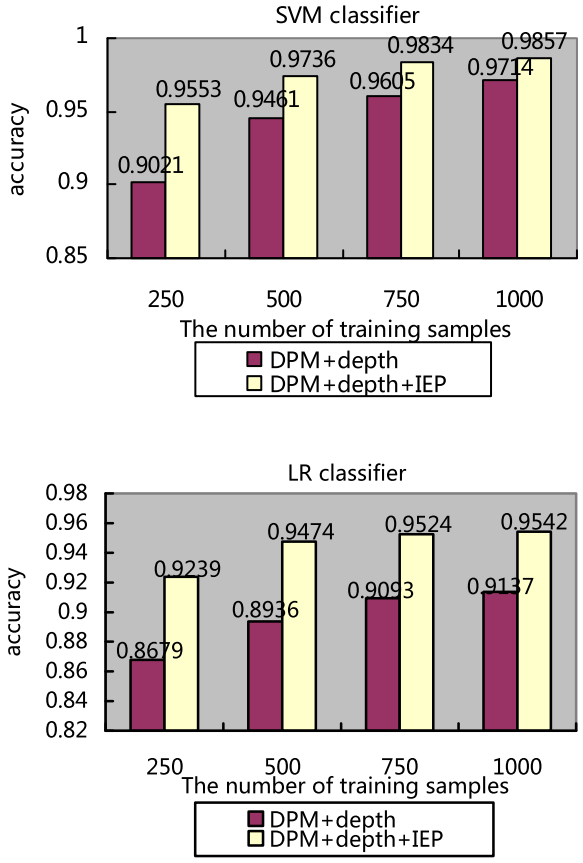


FIGURE 9. Result comparison between the accuracy of preprocessed images with/without IEP method. Although more information is available with the depth data and the image preprocessing strategy, the addition of IEP data can still show positive influence to rise the accuracy.

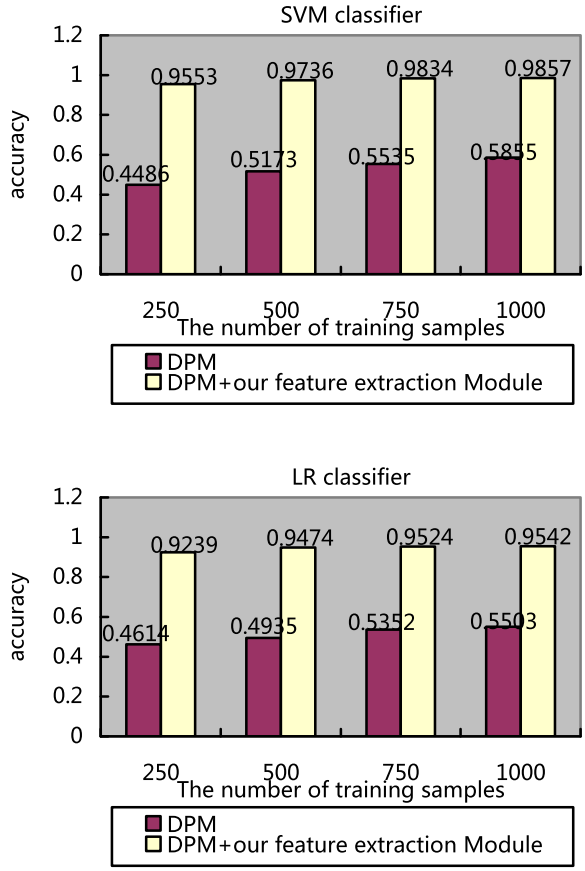


FIGURE 10. Result comparison between the accuracy of preprocessed images with/without our feature extraction module. It can be find that the feature extraction module helps to improve the classification performance.

and the details which are not related with the target can be ignored. Meanwhile, IEP method can capture the common characteristics of the same style and different features among different styles.

D. ANALYSIS OF CLASSIFIERS

The goal of classification is using an image's characteristics to identify which class it belongs to. There are many kinds of linear classifiers like naïve Bayesian, logistic regression, support vector machine (SVM), random forest and k-Nearest Neighbors (K-NN). In this paper, we empirically test two kinds of classifiers – SVM and logistic regression in terms of the properties of the dataset we experiment on, and choose the best for obtaining the final classification result. To make a fair comparison, all the other factors, such as image preprocessing strategy and feature extraction parameters of GoogLeNet and IEP are all the same.

The result of SVM and Logistic Regression are reported with the optimal parameter as aforementioned. From Fig.11, we can see that the classification effect of SVM is 3% higher than that of Logistic Regression. The result of SVM is the best. Thus we adopt the SVM as our final classifier.

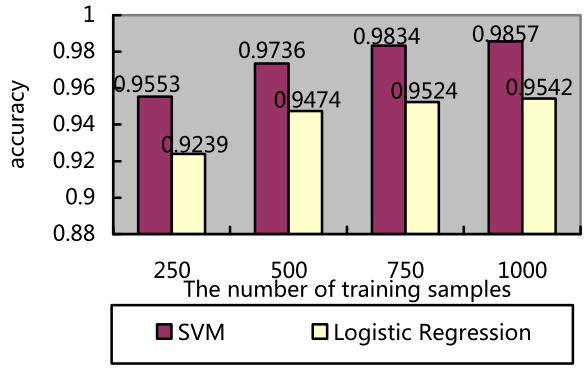


FIGURE 11. The comparison of our proposed method with SVM classifier and Logistic Regression classifier.

E. THE FINAL COMPARISON

In this subsection, we compare the proposed approach with state-of-the-art approaches. The results of the comparison are shown in table 1. It compares the classification accuracy of our method with other algorithms, including DPM-LSVM [35], DPM-MLLR [12], MLLR Spatial Pyramid (MLLR-SP) [36]. The classification of accuracy of our

TABLE 1. Results on the architectural style classification.

Methods	Accuracy
DPM+LSVM	37.69%
DPM+MLLR	42.55%
MLLR+SP	46.21%
DPM+IEP+SVM	55.35%
DPM+FEATURE EXTRACTION MODEL+SVM(OUR)	98.57%

method has achieved the best results and our feature extraction model has made a great contribution to this.

V. CONCLUSIONS

We propose a feature extraction module based on DNN and our IEP method. DNN can learn high-layer features from the architectural style images. IEP is based on the local-consistency and the exotic-inconsistency assumptions that can find common features of the same style and differences among 25-class architectural styles. The new features are used to classify the architectural style. Experimental results show that our method achieves the best performance. The method is competitive comparing with other algorithms.

REFERENCES

[1] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010, doi: 10.1109/TPAMI.2009.167.

[2] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.

[3] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2818–2826.

[4] B. Schlkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2002.

[5] T. Joachims *et al.*, *Transductive Support Vector Machines*. Cambridge, MA, USA: MIT Press, 2006, pp. 105–118.

[6] D. W. Hosmer and S. Lemeshow, "Introduction to the logistic regression model: Testing for the significance of the coefficients," in *Applied Logistic Regression*, 2nd ed. New York, NY, USA: Wiley, 2000, pp. 1–30.

[7] A. C. Berg, F. Grabler, and J. Malik, "Parsing images of architectural scenes," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.

[8] W.-T. Chu and M.-H. Tsai, "Visual pattern discovery for architecture image classification and product image search," in *Proc. 2nd ACM Int. Conf. Multimedia Retr.*, 2012, Art. no. 27.

[9] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros, "What makes Paris look like Paris?" *ACM Trans. Graph.*, vol. 31, no. 4, 2012, Art. no. 101.

[10] A. Goel, M. Juneja, and C. V. Jawahar, "Are buildings only instances?: Exploration in architectural style categories," in *Proc. 8th Indian Conf. Comput. Vis., Graph. Image Process.*, 2012, Art. no. 1.

[11] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[12] Z. Xu, D. Tao, Y. Zhang, J. Wu, and A. C. Tsoi, "Architectural style classification using multinomial latent logistic regression," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 600–615.

[13] H. Wang and B. Raj, "A survey: Time travel in deep learning space: An introduction to deep learning models and how deep learning models evolved from the initial ideas," Cornell Univ. Library, Nov. 2015, pp. 1–43.

[14] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[16] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.* Zürich, Switzerland: Springer, 2014, pp. 818–833.

[17] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *Proc. Brit. Mach. Vis. Conf.*, Swansea, U.K., 2014, pp. 1–12.

[18] M. Lin, Q. Chen, and S. C. Yan, "Network in network," in *Proc. Int. Conf. Learn. Representations*, Banff, AB, Canada, 2014, pp. 1–10.

[19] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," in *Proc. ICLR*, Banff, AB, Canada, 2014.

[20] Z. Zhong, L. Jin, and Z. Xie, "High performance offline handwritten chinese character recognition using googlenet and directional feature maps," in *Proc. 13th Int. Conf. Document Anal. Recognit. (ICDAR)*, Aug. 2015, pp. 846–850.

[21] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, San Diego, CA, USA, 2015.

[23] D. Dai and L. Van Gool, "Unsupervised high-level feature learning by ensemble projection for semi-supervised image classification and image clustering," ETH Zurich, Zürich, Switzerland, Tech. Rep., May 2015. [Online]. Available: https://arxiv.org/abs/1602.00955

[24] Z. Lan, M. Lin, X. Li, A. G. Hauptmann, and B. Raj, "Beyond Gaussian pyramid: Multi-skip feature stacking for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 204–212.

[25] Q. Zhang *et al.*, "Wildland forest fire smoke detection based on faster R-CNN using synthetic smoke images," *Procedia Eng.*, vol. 211, pp. 441–446, Feb. 2018.

[26] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[27] H. Nakahara, H. Yonekawa, and S. Sato, "An object detector based on multiscale sliding window search using a fully pipelined binarized CNN on an FPGA," in *Proc. IEEE Int. Conf. Field Program. Technol. (ICFPT)*, Dec. 2017, pp. 168–175.

[28] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[29] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian Denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[30] N. T. V. Thieu *et al.*, "An evaluation of purified *Salmonella* Typhi protein antigens for the serological diagnosis of acute typhoid fever," *J. Infection*, vol. 75, no. 2, pp. 104–114, 2017.

[31] J. Li, X. Liang, S. Shen, T. Xu, J. Feng, and S. Yan, "Scale-aware fast R-CNN for pedestrian detection," *IEEE Trans. Multimedia*, vol. 20, no. 4, pp. 985–996, Apr. 2018.

[32] Y. LeCun, F. J. Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun./Jul. 2004, pp. 96–104.

[33] Q. Miao, R. Liu, P. Zhao, Y. Li, and E. Sun, "A semi-supervised image classification model based on improved ensemble projection algorithm," *IEEE Access*, vol. 6, pp. 1372–1379, 2018.

[34] D. Dai and L. Van Gool, "Ensemble projection for semi-supervised image classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2072–2079.

[35] M. Pandey and S. Lazebnik, "Scene recognition and weakly supervised object localization with deformable part-based models," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV).*, Nov. 2011, pp. 1307–1314.

[36] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, Jun. 2006, pp. 2169–2178.

PEIPEI ZHAO received the M.E. degree from the School of Computer Science and Technology, Xidian University, in 2016, where she is currently pursuing the Ph.D. degree. Her research interests include pattern recognition and digital image processing.

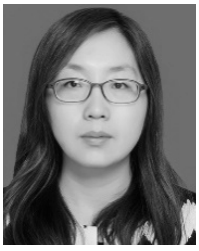
QIGUANG MIAO is currently a Professor and also a Ph.D. Student Supervisor with the School of Computer Science and Technology, Xidian University. He is a member of the Professor Committee. In 2012, he was supported by the Program for New Century Excellent Talents at the University by the Ministry of Education. He is a committee member of the CCF, a committee member of CCF Computer Vision, and the Vice Chairman of CCF YOCSEF.

He received the Ph.D. degree in computer application technology from Xidian University in 2005. His research interests include computer vision, machine learning, and Big Data. As the Principal Investigator, he is doing or has completed four projects of NSFC, two projects of Shaanxi Provincial Natural Science Fund, over 10 projects of National Defence Pre-research Foundation, and the 863 and Weapons and Equipment Fund. He has hosted one project supported by the Fundamental Research Funds for the Central Universities by MOE. In the field of teaching, he was awarded as one of the Pacemaker of Ten Excellent Teacher twice in 2008, 2011, and 2014, respectively.

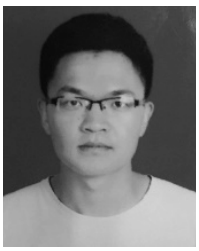
In recent years, he has published over 100 papers in the significant domestic and international journal or conference including the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, the *Journal of Visual Communication and Image Representation*, *NeuroComputing*, and the *IET Image Processing, Knowledge Based System*, of which over 30 papers are indexed by SCI and over 40 papers are included in EI. He has served as the Committee Chairman of the first CCF Youth Elite Association, CNCC2008, CIS 2012, CCFAI 2013, CCDM2014 PC member, and CIS 2013 Special Session Chair. He is a committee member of the Editorial Board of the Internet of Things, and the Assessment Expert of the State Science and Technology Prizes and the National Defense Basic Scientific Research Project. He has been a recipient of a Prize at the ministerial and provincial level twice.

JIANFENG SONG was born in 1978. He received the M.S. degree in computer science in 2001. His research interests include broadband wireless network cross-layer protocol design and performance analysis, heterogeneous wireless networks, computer system security, and malware analysis.

YUTAO QI was born in Henan, China, in 1981. He received the B.S. degree in software engineering from the School of Software Engineering, Xidian University, Xi'an, China, in 2003, and the M.S. degree in computer science and technology from the Institute of Information Processing, Xidian University, in 2006. He is currently a Professor with Xidian University. His research interests include evolutionary computation, multi-agent systems, artificial immune systems, parallel computing, and data mining.



RUYI LIU received the B.S. degree from Shaanxi Normal University, Shaanxi, China, in 2012. She is currently pursuing the Ph.D. degree with the School of Computer and Technology, Xidian University, Shaanxi, China. Her current interests include image classification and segmentation, pattern recognition, and computer vision methods with applications in remote sensing.



DAOHUI GE received the B.S. degree from the Shandong University of Finance and Economics, Shandong, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Computer and Technology, Xidian University, Shaanxi, China. His current interests include object tracking, pattern recognition, and deep learning.