# Recognizing Art Style Automatically in Painting Using Convolutional Neural Network

**Chapter** *in* Lecture Notes in Electrical Engineering · March 2022

DOI: 10.1007/978-981-16-8484-5_20



3 authors:

Mahfuza Akter

2 PUBLICATIONS 1 CITATION



Mst Rasheda Akther
International Islamic University Chittagong

2 PUBLICATIONS 1 CITATION



Md. Khaliluzzaman
International Islamic University Chittagong

99 PUBLICATIONS 382 CITATIONS

Lecture Notes in Electrical Engineering 834

Amit Kumar
Jacek M. Zurada
Vinit Kumar Gunjan
Raman Balasubramanian *Editors*

# Computational Intelligence in Machine Learning

## Select Proceedings of ICCIML 2021

Springer

# Recognizing Art Style Automatically in Painting Using Convolutional Neural Network

**Mahfuza Akter, Mst. Rasheda Akther, and Md. Khaliluzzaman**

**Abstract** Recognition of art style in painting is an active and important research area in the computer visions field as art style recognition is not depending on the definite feature; however, extracting important in-definitive features from a painting image is a crucial fact. Moreover, using deep convolutional neural network (CNN) makes the model overfit and small CNN model makes the model underfit. To overcome these problems, in this work, a model is developed by us for art style recognition in painting based on the shallow convolutional neural network (SCNN). The proposed model is developed based on the two consecutive convolutional layers and single fully connected layer to extract painting features and recognize the art styles accordingly. For training and testing purposes in the art style recognition, we have used Wikipaintings dataset. We also used our created real-time dataset in this work. Our proposed model shows 60.37% accuracy which is the significant improvement over some current state of the arts.

**Keywords** Feature extraction · Overfit · Underfit · Art styles in paintings · Shallow convolutional network · Wikipaintings dataset

## 1 Introduction

In the current decay, art style recognition is a vital issue in the computer vision's field because of its growing demands in digital multimedia industries as well as academic area. As art styles, visual features are not defined in identifiable features way such as object or facial recognition, so, it is really challenging task to identify the art style in automatic way. The art styles classification is also difficult for non-representational art work [1].

Many researchers have done many research works to recognize the art style through various handcrafted features and machine learning approaches. From this research, they identify many problems to classify the art style. Such as, in [2],

M. Akter · Mst. R. Akther · Md. Khaliluzzaman (✉)
Department of Computer Science and Engineering, International Islamic University Chittagong (IIUC), Chittagong 4318, Bangladesh

 
A. Kumar et al. (eds.), *Computational Intelligence in Machine Learning*, Lecture Notes in Electrical Engineering 834, https://doi.org/10.1007/978-981-16-8484-5_20

authors examine the efficiency of various renown image features, i.e., special envelops, histogram-oriented gradients (HOGs), discriminative color names with various machine learning classification approaches, i.e., support vector machine (SVM), random forests. From this work, authors observe that it is hard to classify some art styles with these techniques. They also observe that adding more features, the model's performance is not improved.

In the recent years, to derive the critical visual features from an image, researchers focus on deep convolutional neural networks. It is a bio-inspired hierarchical multi-layered neural network which can capable to learn visual patterns immediately from pixels of image frames except any preprocessing steps [2, 3]. Many researchers apply DCNN models to identify the previous problems and recognize the art styles efficiently from different classes. Such as, in [4], authors observed the limitations of handcrafted feature and proposed a CNN model. Their CNN model is based on the AlexNet [5], and the model was trained with ImageNet. This model performs better with state of the art and increases the accuracy of art style classification to 54.5% for 25 classes.

Each art contains different features in various regions of paintings. In this respect, methods of feature extraction should be able to extract characteristics from the art's important parts. However, the extracting feature from a painting image is a big issue in the art styles painting classification. In this regard, many researchers develop many CNN models for art styles painting classifications. However, these models are sometime overfitted and underfitted because of large and small amount of convolutional layers in this network. To overcome these problems and develop a regularized CNN model, in this work, we have proposed a shallow convolutional neural network (SCNN). The proposed network contains two consecutive convolutional layers and one fully connected layer. In this work, Wikipaintings and real dataset are used to train as well as test the proposed shallow model.

The remaining part of this paper is presented as follows. In Sect. 2, recent state-of-the-art work is presented. In Sect. 3, proposed model is presented. In Sect. 4, experimental results are showed. The conclusion is given in Sect. 5.

## 2 Related Works

Recognition of art style is an important system, and several researchers have done many works on the recognition of art style because of its diverse applications. In an automatic way, identifying the style of an art is a difficult task. Some primitive challenges stand in the method of recognition of art style in painting. CNN model is a better and effective model for recognizing art style.

So many pieces of research observed that most of the systems designed for recognition of art style automatically were created on the features of handcrafted and used algorithms which are focused on deep learning to identify art styles. For example, in [6], authors proposed a convolutional neural network to classify painting images from challenging environments. Authors tested their designed model with the challenging test dataset which is collected from different art movements. In [7], an algorithm of multi-task learning was presented by them to learn a style-special dictionary representation. To exhibit the usefulness of their method, they introduced the dataset whose name is DART, which contains over 1500 pictures of arts representative of various styles.

An auto-encoder-based deep CNN model is proposed in [8]. Here, auto-encoder is used for the purpose of preprocessing. The encoded image is used to train a supervised model which is based on the CNN. The performance shows by the propose model is better than the state of the art works. It also shows that the deep neural network performs better to recognize the painting classification. Another CNN model is proposed in [9], where generated distorted painting image is used for training and validation process. They compared the proposed model with the handcrafted feature, i.e., SIFT method and shows the outperformance.

In [10], a deep CNN model is proposed by the authors to explain the perceptiveness of visual features to identify the artistic styles in painting. The authors also suggest a rigid binary representation of the painting. A style predicting approach is proposed in [4], where authors evaluate various image features to perform the task. This work exhibits several types of styles not previously considered in the literature, and they exhibited state-of-the-art results in the prediction of both style and aesthetic quality for multilayer networks.

In [11], authors investigate the different applicability's of metric for the learning approach as well as different applicability of visual features to estimate the learning similarities with the collection of painting. In this work, authors consider three basic concept, i.e., style, artist, and genre. A deep CNN-based fine art painting classification model is proposed in [1]. The authors divide the objective of the work in twofold. Firstly, authors develop an end-to-end DCNN model to examine the capability of CNN model. Finally, authors argue that the fine art painting classification is more challenging with respect to the other classification model such as face or object recognition.

In [12], an artificial system is introduced to generate artistic images through a deep neural network with high perceptual quality. The proposed system introduces neural system which creates an arbitrary image by separating and recombining image contents. This work shows a flow of algorithmic representation of how a person creates and perceives artistic image. The effect of very DCNN is estimated in [13]. The proposed network's complexity is very high because the network is very deep. However, the accuracy of the model is significantly improved. A deep residual neural network is introduced in [14] detecting artistic style painting. For that, the model is pre-trained by using the ImageNet. However, the model is underfitted.

## 3 The Proposed Framework

In this paper, an automatic art style recognition framework is established. Improving a deep neural network conjointly depends on adding a lot of layers or neurons, facilitating network gradient flow, notably for classification problems with a massive number of classes. As the art style recognition methods contain the small number of output classes as well as small number of datasets, in this paper we demonstrate a model with the small number of convolutional and fully connected layers, which show promising outcomes.

### *3.1 Model Explanation*

In this thesis work, a CNN framework is proposed which contains two convolutional layers and one fully connected layer. In Fig. 1, the workflow diagram of our proposed model is presented. The proposed convolutional neural network architecture is presented in Fig. 2. Two convolutional layers are contained by the framework which extracts the important features of art's region. The framework contains one fully connected layer which recognizes art style.

In our proposed model, 32 filters consist in the first convolutional layer and filter's size is $3 \times 3$ and each has 3 channels. Same padding is used by us; that is, the input pictures are zero-padded so that the filters convolve over each input picture's pixel. In the convolution layer, we used an activation function that is ReLU. To improve model performance and stability, the batch normalization is used.

The size of max pooling filter is $2 \times 2$ that moves by a stride of 2. On the feature maps, it performs max pooling operation. We have used a dropout layer in the shake of dropping out the neurons which helps prevent this model from overfitting to the data which are trained. For the first convolution layer, dropout value has been set to 0.20.

64 filters of size $3 \times 3$ as well as each contain 3 channels the second convolutional layer is made. Here the SAME padding is also used. To the feature maps, the ReLU activation function is also applied. In the second layer to improve the model performance, again the batch normalization is used. The size of max pooling filter is $2 \times 2$ that moves by a stride of 2, and on the feature maps, it executes max pooling operation. For second convolution layer, the value 0.25 has been set as dropout value to overcome the problem of overfitting. After setting the dropout value of second convolutional layer, a flatten layer has been used.

The final pool layer's result is flattened into a victories type and succeeded by a fully connected layer which has 512 neurons. Then we have used fully connected layers for classifying the final features into the several classes in the output layer which are extracted from the preceding pooling layers and convolution layers. From features, the fully connected layers learn. We have applied batch normalization again. Then we used the dropout value 0.5. For activation function in dense layer, we used ReLU again.

Finally, the output layer consists of 6 nodes representing 6 classes. Then we used softmax activation function to classify the desired label in the output layer.

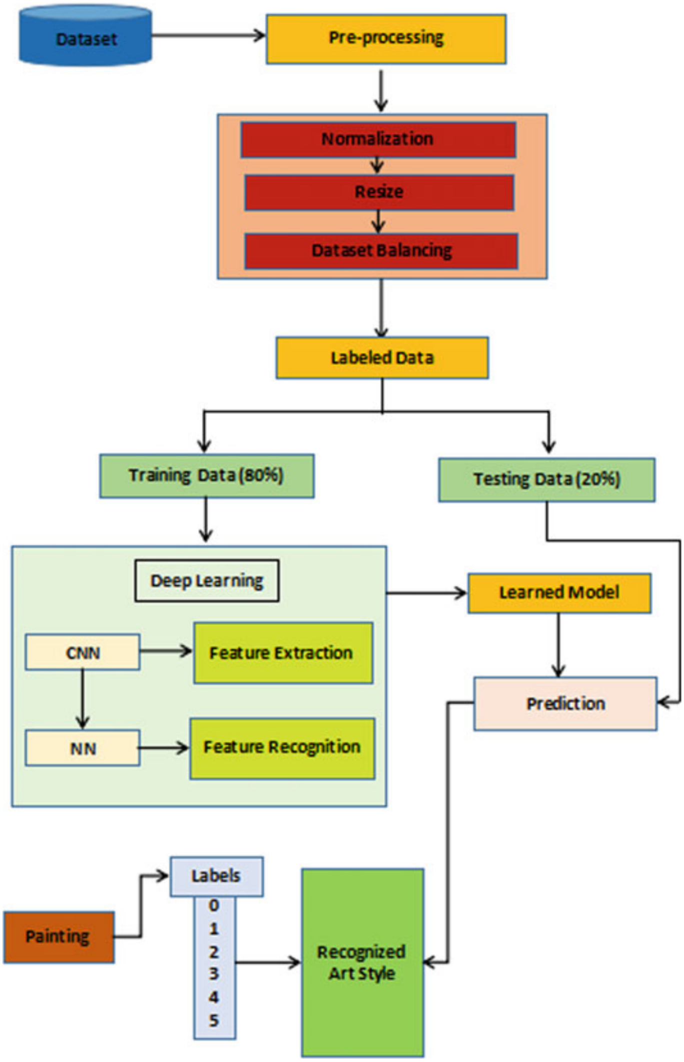



**Fig. 1** Workflow diagram of the proposed framework

## 3.2 *Classification Loss Function*

We used a loss function for assessing the classification model's performance whose output is a probability value. In machine learning context, a log loss and cross-entropy loss can be considered equal. Performance of a model is measured by it, and a probability value among 0–1 is its output. For classification of binary, mathematical formula which is used to calculate cross-entropy loss is shown in (1).

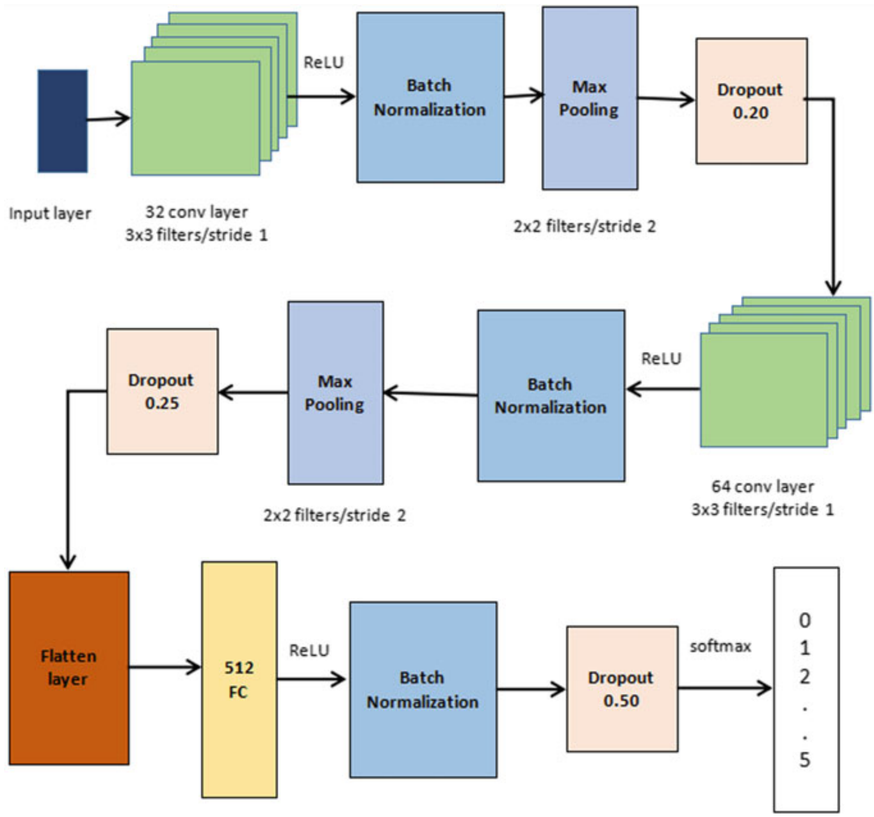



**Fig. 2** Proposed art style recognition model

$$\text{Loss} = -(y(p)) + (1 - y)\log(1 - p) \tag{1}$$

Cross-entropy loss function is used when there are two or more label classes. In this work, we consider the categorical cross-entropy loss.

## 4 Experimental Results

In this section, we presented the detail review of experiments of our proposed model based on the recognition of art style datasets. We have showed our proposed model's performance on the Wikipaintings dataset, and some encouraging recent works compare the results. Several real datasets are also shown during the evaluation of the performance of our proposed model. By using the ideal network miniVGGNet [15], we train and test Wikipaintings dataset and real-time paintings. We have also shown the comparison of performance between the proposed model and miniVGGNet.

## 4.1 Datasets

We have used Wikipaintings dataset which is moderate in size. For training and testing of our model, we have used this dataset. Our dataset is divided into two parts. In the shake of training as well as testing purpose in the art style recognition, we have used 8000 images of Wikipaintings dataset. To train the model, 80% of images are used and rest 20% of dataset images are used for testing. We used 6 levels of Wikipaintings dataset. Examples of some images of Wikipaintings dataset and real dataset are presented in Figs. 3 and 4.

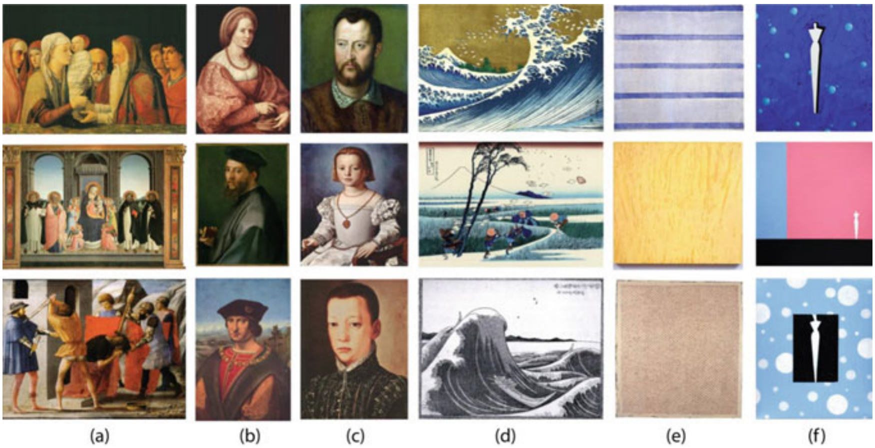


**Fig. 3** Sample images from Wikipaintings dataset: **a** early renaissance, **b** high renaissance, **c** mannerism late renaissance, **d** Ukiyo-e, **e** minimalism, and **f** pop art

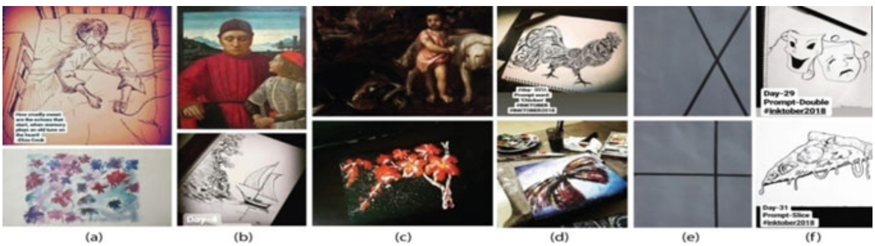


**Fig. 4** Sample images from real dataset: **a** early renaissance, **b** high renaissance, **c** mannerism late renaissance, **d** Ukiyo-e, **e** minimalism, and **f** pop art

### 4.2 Experimental Analysis and Comparison

We drive the network on the Wikipaintings dataset's sub-set, justify the validation set, and investigate the accuracy through the test set. Before we get into the performance details of the model on various datasets, we discuss our training operation briefly. We trained each system per dataset in our simulations, and among such various models, we tried to keep the architecture and hyper-parameters identical. In this work, 40 epochs are used for training. For optimization, an empirical learning rate is used that is 0.001. The optimizer algorithm used in this method is Adam optimizer. As it is adaptive optimizer, it takes less reasonable time to train a model for Wikipaintings datasets. For the Wikipaintings dataset, we have used 20% of images for validation and rest of the images in the training which is 6400 images. We have used 1600 images for validation.

We evaluate the performance based on the different machine learning matrices such as accuracy, recall, precision, and $F1$-score. The proportion of total samples correctly recognized by the classifier is known as accuracy present in (2). The recall is the total positive samples that the classifier correctly predicted as positives present in (3). The total predicted positive samples by the classifier that are true positives are known as precision present in (4). $F1$-score measures a balanced average outcome by combining precision and recall present in (5). The above equations show how to calculate different machine learning matrices with true positive (TP), false positive (FP), true negative (TN), and false negative (FN), respectively.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4}$$

$$F1 - \text{score} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}} \tag{5}$$

In Fig. 5a, curve of validation loss as well as training loss for miniVGGNet of Wikipaintings dataset is shown, and in Fig. 5b, curve of validation accuracy as well as training accuracy is presented. From the accuracy curve, we can see on the Wikipaintings dataset, miniVGGNet gives 41.87% accuracy. We can estimate the error in the validation from the loss curve. From the curves, it is observed that the loss both of training and testing are reduced constantly till 8 epochs; however, after that the loss is increasing rapidly. That means the model is going to overfit.

In Fig. 6, the evaluated result's confusion matrix is presented. How well the class labels are predicted can be calculated from the confusion matrix. In Table 1, the contribution of miniVGGNet on the prediction of each label is shown. The

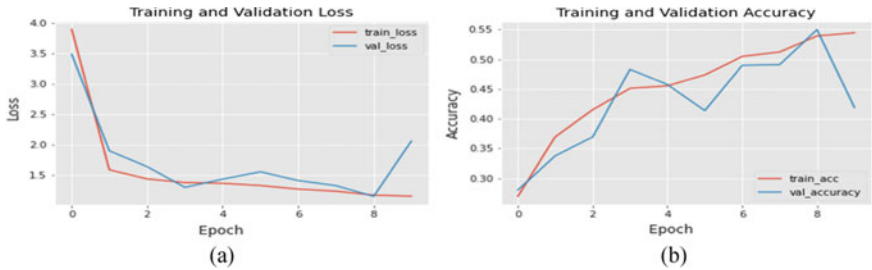



**Fig. 5** Loss and accuracy curve for miniVGGNet for Wikipaintings dataset: **a** training loss and validation loss curve, **b** training accuracy and validation accuracy curve

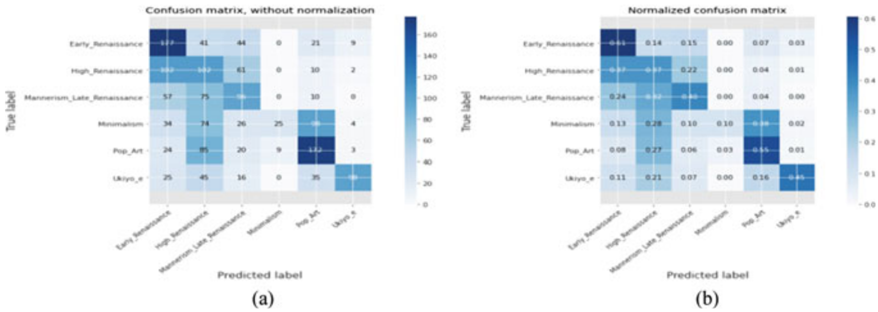



**Fig. 6** Confusion matrix for Wikipaintings dataset for miniVGGNet: **a** confusion matrix and **b** normalized confusion matrix

**Table 1** Predicted score for each label by miniVGGNet for Wikipaintings dataset

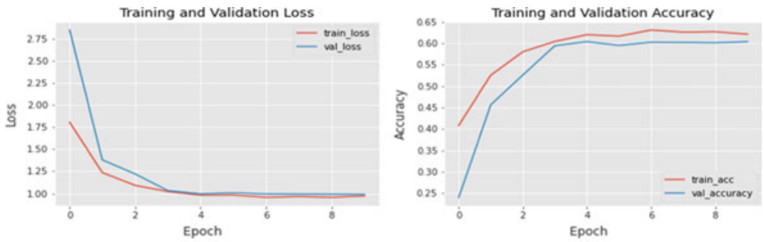
| Labels | Predicted score |
|---|---|
| Early renaissance | 0.61 |
| High renaissance | 0.37 |
| Mannerism late renaissance | 0.40 |
| Minimalism | 0.10 |
| Pop art | 0.55 |
| Ukiyo-e | 0.45 |

classification report based on the Wikipaintings dataset applied by miniVGGNet is shown in Table 2.

In Fig. 7a, curve of training loss as well as validation loss for the Wikipaintings dataset for the proposed model is shown. In Fig. 7b, curve of training accuracy as well as validation accuracy is presented. The curve shows a smooth increase in accuracy. Accuracy remains constant from the 30th epoch. From the curves, it is observed that the loss both of training and testing are reduced constantly. That means the model is learning perfectly. On the overall evaluation of the loss and training curves, it is confirmed that the model is performed regularized. In Fig. 8, we have shown the

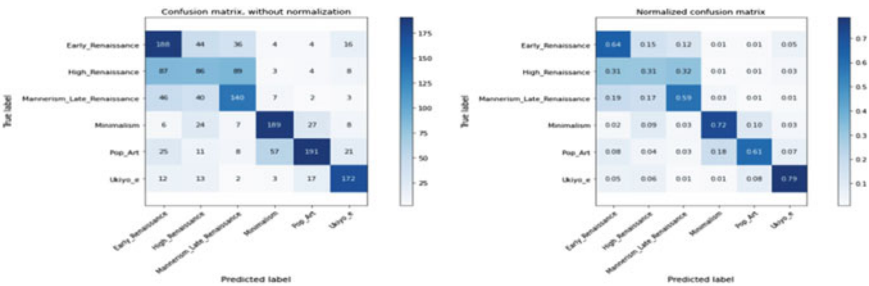
**Table 2** Classification report for Wikipaintings dataset by miniVGGNet

| Labels | Precision | Recall | $F1$-score | Support |
|---|---|---|---|---|
| Early renaissance | 0.42 | 0.61 | 0.50 | 292 |
| High renaissance | 0.24 | 0.37 | 0.29 | 277 |
| Mannerism late renaissance | 0.37 | 0.40 | 0.38 | 238 |
| Minimalism | 0.74 | 0.10 | 0.17 | 261 |
| Pop art | 0.50 | 0.55 | 0.52 | 313 |
| Ukiyo-e | 0.84 | 0.45 | 0.59 | 219 |
| Accuracy | | | 0.42 | 1600 |
| Macro average | 0.52 | 0.41 | 0.41 | 1600 |
| Weighted average | 0.51 | 0.42 | 0.41 | 1600 |




(a) (b)

**Fig. 7** Loss and accuracy curve of the proposed model for Wikipaintings datasets: **a** training loss and validation loss curve, **b** training accuracy and validation accuracy curve




(a) (b)

**Fig. 8** Confusion matrix of Wikipaintings dataset for proposed model: **a** confusion matrix and **b** normalized confusion matrix

confusion matrix. The proposed network's accuracy on the Wikipaintings dataset is 60.37%. The proposed model of the Wikipaintings dataset is more than 18.50% better than the miniVGGNet. In Table 3, the predicted score for each label is pesented, where Table 4 presents the classification report for Wikipaintings dataset by proposed model.

**Table 3** Predicted score for each label by proposed model

| Labels | Predicted score |
|---|---|
| Early renaissance | 0.64 |
| High renaissance | 0.31 |
| Mannerism late renaissance | 0.59 |
| Minimalism | 0.72 |
| Pop art | 0.61 |
| Ukiyo-e | 0.79 |

**Table 4** Classification report for Wikipaintings dataset by proposed model

| Labels | Precision | Recall | $F1$-score | Support |
|---|---|---|---|---|
| Early renaissance | 0.52 | 0.64 | 0.57 | 292 |
| High renaissance | 0.39 | 0.31 | 0.35 | 277 |
| Mannerism late renaissance | 0.50 | 0.59 | 0.54 | 238 |
| Minimalism | 0.72 | 0.72 | 0.72 | 261 |
| Pop art | 0.78 | 0.61 | 0.68 | 313 |
| Ukiyo-e | 0.75 | 0.79 | 0.77 | 219 |
| Accuracy | | | 0.60 | 1600 |
| Macro average | 0.61 | 0.61 | 0.61 | 1600 |
| Weighted average | 0.61 | 0.60 | 0.61 | 1600 |

The model consequences for the Wikipaintings dataset are presented in Figs. 9 and 10 for random test images, respectively. Here, different art styles are recognized by the miniVGGNet and our proposed model competently. The corresponding confidence score of art style recognition is also shown in Figs. 9a and 10a, respectively. Then the recognition confidence scores in the bar chart of different art styles are presented in Figs. 9b and 10b, respectively.

### Misclassified Art Style by miniVGGNet

Figure 11 shows some misclassification images by miniVGGNet. Confidence scores with recognition label are shown in Fig. 11a and b shows the respective confidence score in the bar chart.

### Correctly Classified Art Style by Proposed Model Which Is Misclassified by miniVGGNet

Sample images of the Wikipaintings dataset which are predicted as wrong by miniVGGNet are tested by the proposed network. Predicted recognition label with confidence score is shown in Fig. 12a. Besides, Fig. 12b shows the respective confidence score in the bar chart. Prediction results of Fig. 12 show the accurate output as expected.

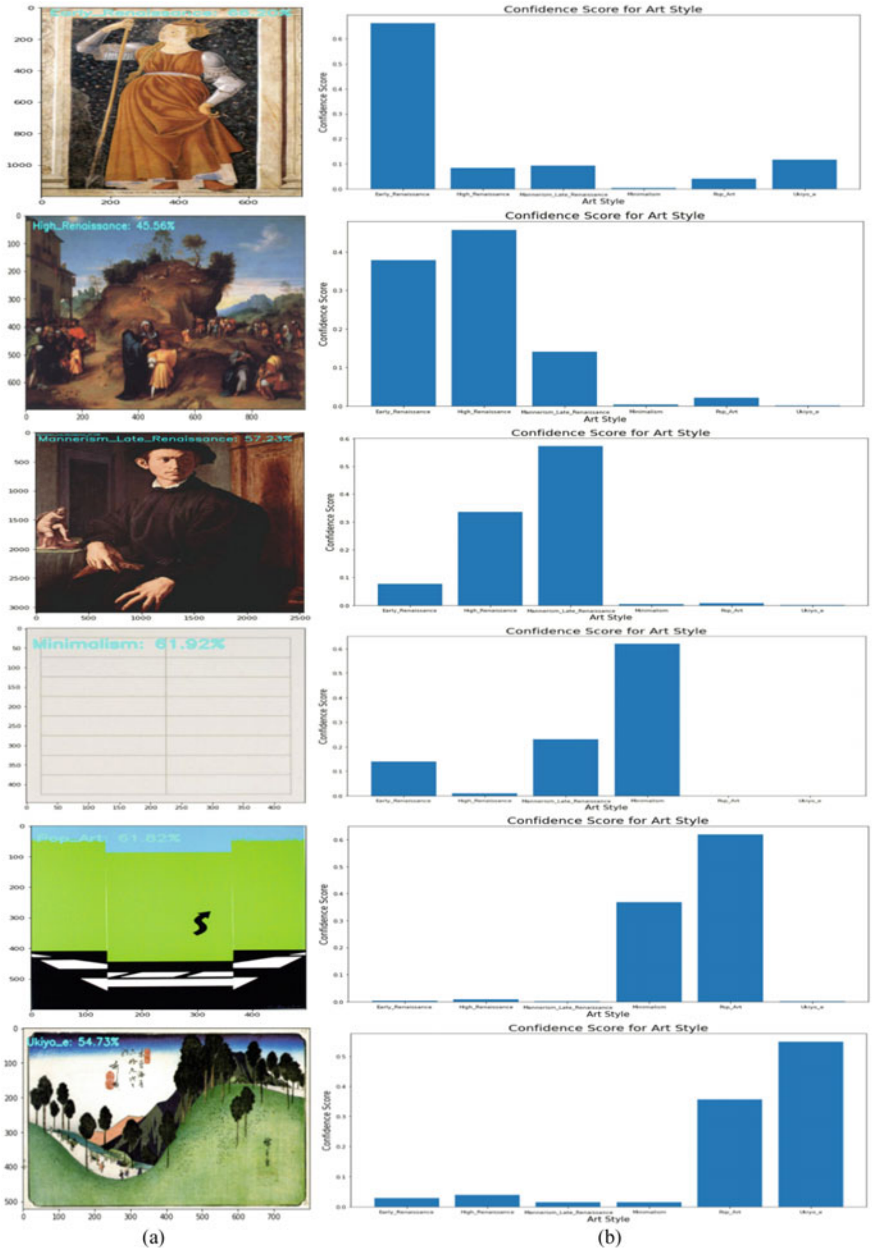
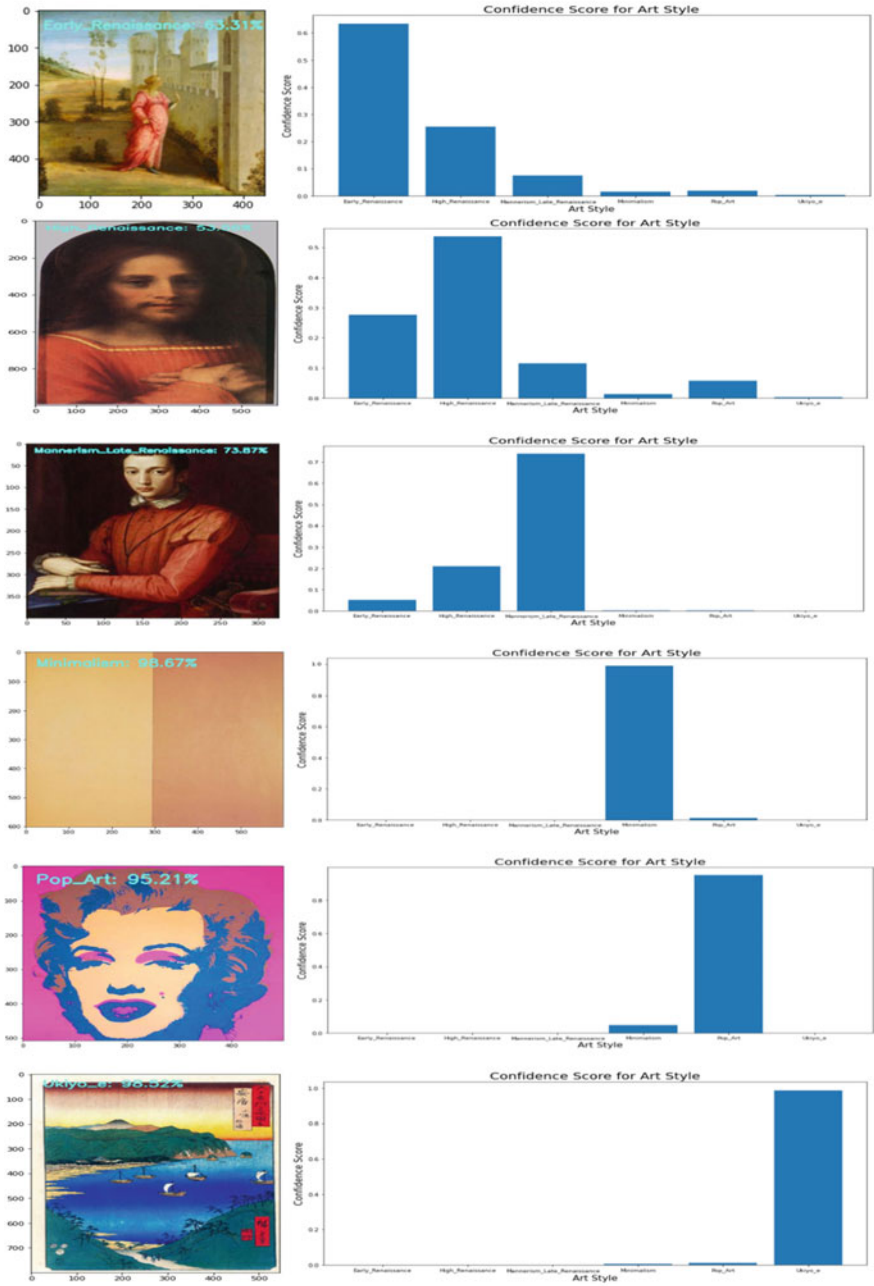



**Fig. 9** Random experimental result of Wikipaintings dataset for miniVGGNet: **a** art style recognition with confidence score and **b** recognition confidence scores in the bar chart




(a) (b)

**Fig. 10** Random experimental result of Wikipaintings dataset for proposed model: **a** art style recognition with confidence score and **b** recognition confidence scores in the bar chart

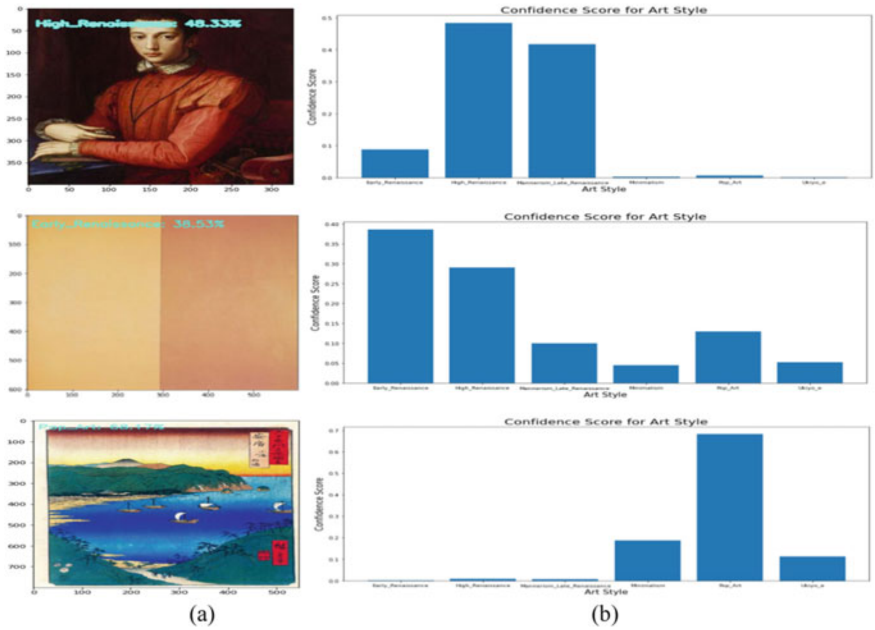



**Fig. 11** Some sample images which are wrong predicted by miniVGGNet, **a** confidence scores with recognition label and **b** confidence scores in the bar chart

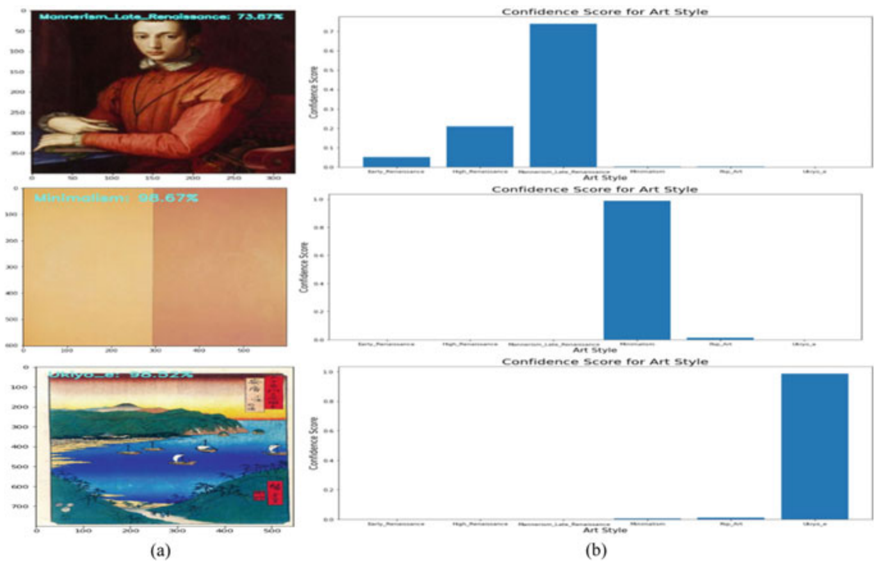



**Fig. 12** Correctly predicted by proposed model those are misclassified by miniVGGNet: **a** confidence scores with recognition label and **b** confidence scores in the bar chart

**Table 5** Accuracy using different learning rate schedule

| learning rate schedules | Accuracy (%) |
|---|---|
| Step-based | 31.13 |
| Linear | 41.31 |
| **Polynomial (Proposed method)** | **60.37** |

The text is bold to indicate the best performance learning rate and method

**Table 6** Classification accuracy on Wikipaintings dataset

| Method | Accuracy (%) |
|---|---|
| **The Proposed method** | **60.37** |
| miniVGGNet | 41.87 |
| ResNet [14] | 57.00 |

The text is bold to indicate the best performance learning rate and method

In our proposed model, we applied three types of learning rate schedule: step-based learning rate schedule, linear learning rate schedule as well as polynomial learning rate schedule. From Table 5, it is shown that the polynomial learning rate schedule shows the best accuracy. In this regard, in our work, we have used the polynomial learning rate schedule. When we used step-based learning rate schedules, we got 31.13% accuracy. When we used linear learning rate unless schedules, we got 41.31% accuracy. When we used polynomial learning rate schedules, we got 60.37% accuracy. Our model is compared with some existing state of the art that is miniVGGNet and ResNet. The comparison of our model with miniVGGNet and ResNet is shown in Table 6.

## 5 Conclusion

In this work, CNN based an art style recognition architecture is proposed. The proposed CNN model automatically detects several characteristics of individual paintings to support fast, correct, and reliable identification and recognition of an art style. We also have shown comparison between our proposed model and existing network, i.e., miniVGGNet and ResNet using Wikipaintings dataset. We also show some sample examples of real-time paintings. With the accuracy of 60.37%, our model which we have proposed performs efficiently on the Wikipaintings dataset using only one fully connected layer as well as two convolutional layers. In addition, the performance of miniVGGNet is also précised on the same dataset. In the future, we will try to provide better model that will extract the complicated painting features from the painting image and recognize the art style painting more accurately.

## References


1. W.R. Tan, C.S. Chan, H.E. Aguirre, K. Tanaka, Ceci n'est pas une pipe: a deep convolutional network for fine-art paintings classification, in *2016 IEEE International Conference on Image Processing (ICIP)* (IEEE, 2016), pp. 3703–3707
2. Y. LeCun, K. Kavukcuoglu, C. Farabet, Convolutional networks and applications in vision, in *IEEE International Symposium on Circuits and Systems*, pp. 253–256 (2010)
3. A. Binder, W. Samek, G. Montavon, S. Bach, K.R. Müller, Analyzing and validating neural networks predictions, in *Proceedings of the ICML 2016 Workshop on Visualization for Deep Learning* (2016)
4. S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, H. Winnemoeller, Recognizing image style. arXiv preprint arXiv:1311.3715 (2013)
5. A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks. Adv. Neural. Inf. Process. Syst. **25**, 1097–1105 (2012)
6. M.O. Kelek, N. Calik, T. Yildirim, Painter classification over the novel art painting data set via the latest deep neural networks. Proc. Comput. Sci. **154**, 369–376 (2019)
7. G. Liu, Y. Yan, E. Ricci, Y. Yang, Y. Han, S. Winkler, N. Sebe, Inferring painting style with multi-task dictionary learning, in *IJCAI International Joint Conference on Artificial Intelligence* (2015)
8. O.E. David, N.S. Netanyahu, Deeppainter: Painter classification using deep convolutional autoencoders, in *International Conference on Artificial Neural Networks* (Springer, Cham, 2016), pp. 20–28
9. Y. Hong, J. Kim, Art painting identification using convolutional neural network. Int. J. Appl. Eng. Res. **12**(4), 532–539 (2017)
10. Y. Bar, N. Levy, L. Wolf, Classification of artistic styles using binarized features derived from a deep neural network, in *European Conference on Computer Vision* (Springer, Cham, 2014), pp. 71–84
11. B. Saleh, A. Elgammal, Large-scale classification of fine-art paintings: learning the right metric on the right feature. arXiv preprint arXiv:1505.00855 (2015)
12. L.A. Gatys, A.S. Ecker, M. Bethge, A neural algorithm of artistic style. arXiv preprint arXiv:1508.06576 (2015)
13. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
14. A. Lecoutre, B. Negrevergne, F. Yger, Recognizing art style automatically in painting with deep learning, in *Asian Conference on Machine Learning* (PMLR, 2017), pp. 327–342
15. A. Ismail, S.A. Ahmad, A.C. Soh, K. Hassan, H.H. Harith, Improving convolutional neural network (CNN) architecture (miniVGGNet) with batch normalization and learning rate decay factor for image classification. Int. J. Integr. Eng. **11**(4) (2019)