ISSN 2959-6157

Automated Artist Identification Using Deep Learning and Transfer Learning Techniques

Ke Xu

University at Buffalo
kxu25@buffalo.edu

1. Abstract:

Traditional methods of art analysis rely heavily on human expertise, which can be subjective and limited by an individual's familiarity with art history.
This paper introduces an innovation of art style recognition using deep learning and transfer learning techniques. We use the pre-trained ResNet50 architecture to identify artists based on their work. The study aims to show how convolutional neural networks (CNNs) can be applied to complex image recognition tasks, such as identifying artists from their paintings. Our model is trained on a large dataset of digitized artworks with astonishing accuracy, demonstrating the effectiveness of transfer learning in dealing with the high variability of artistic styles.

Keywords: Deep Learning, transfer learning techniques, automating, artworks, models

2. Introduction:

2.1 . Problem Statement:

Traditional methods of artistic analysis rely heavily on human expertise. This is often subjective and inefficient. It can be difficult to process the large data sets typical of digital art archives. In addition, identifying an artist by work requires the analysis of visual elements that are often subtle and highly stylized. Due to the complexity of the features of the work and the subtle differences between the styles of the artists, automating this process using traditional machine learning methods is a huge challenge.

2.2 . Challenges:

● Data Imbalance: Popular artists are overrepresented in datasets, while lesser-known artists may have very few samples available.
● High Dimensionality: Artworks are high-dimensional data points, making feature extraction computationally intensive.
● Subjectivity: Traditional analysis is subjective, varying significantly between experts.

2.3 . Goals:

Develop a neural network model that can accurately identify different artists by artwork.
Transfer learning is applied to improve the method of extracting relevant features from artistic images.
The class imbalance problem is solved systematically by data enhancement and class weights to improve the fairness and accuracy of the model.

3. Background and Related Work:

Artist identification has traditionally been the domain of art historians. They attribute the artwork to the creator primarily by analyzing style and subject matter. With the development of machine learning, CNNs have achieved remarkable results in classification tasks.
● Features from Deep Neural Networks:
○ Deep neural networks trained on object recognition have been utilized for style recognition by classifying artworks based on the period they were created in. This involves training classifiers on top of raw network activations known as content representations. [1]
○ The method of synthesizing images that combine content and style from different sources offers a new tool to study art perception, neural representation, and content-independent image appearance. It allows for designing novel stimuli with independent variations in appearance and content, beneficial for various experimental studies in visual perception. [1]
● Convolutional Neural Networks: CNNs are the backbone of most modern image processing models due to their ability to capture spatial hierarchies in images.
● CNN Architectures for Artist Identification: The research paper introduces the problem of artist identifica-

tion and explores various Convolutional Neural Network (CNN) architectures to maximize classification accuracy, a novel approach not previously undertaken in research. [2]

● Effectiveness of ResNet-18 with Transfer Learning: The best-performing network in the study is based on ResNet-18 pre-trained on ImageNet using transfer learning. This network outperforms traditional feature-based methods significantly, indicating the relevance of features learned from ImageNet data for artist identification. [2]

● Representation of Painting Style: When tasked with predicting artists, the CNNs in the study create a representation of the style of paintings. This is validated through experiments examining the underlying representation and the network's artist predictions. [2]

● Future Research Directions: The paper aims to delve deeper into model representations to quantify the influence of style versus content in artist predictions. This involves calculating Gram matrices of network layers to represent style and using them in a separate CNN for artist predictions. [2]

● Qualitative Analysis of Style Representation: The study conducts experiments with style-transferred images to evaluate the network's understanding of artistic style. Results show that the network can recognize artists' styles independently of specific painting content, further confirming the effectiveness of the approach. [2]

4. Data collection:

The dataset contains images of artwork labeled by artists. The dataset is collected from Kaggle's "Best Artworks of All Time", ensuring a diversity of styles and periods [3]. The dataset contains detailed metadata about each artist and their work, which helps to classify and select images for model training.

The dataset includes three primary components:

● artists.csv: This file contains metadata for each artist, including the number of artworks, their names, and other relevant information sourced from Wikipedia. [3]

● images.zip: This archive contains full-sized images of artworks, organized into folders named after each artist. [3]

● resized.zip: This archive offers the same collection of images, but it is resized for quicker processing and reduced data requirements. [3]

To ensure the quality and consistency of the data used for training, we focus on artists with a large body of work. Specifically, we only retained artists who had at least 300 works to generate a filtered dataset that balanced representation across different categories. This step is essential to mitigate class imbalances and ensure a robust and fair training process.

The pre-processing process begins by verifying that a corresponding image exists for each artist listed in the metadata. Any artist entries missing relevant images in the dataset are excluded, thus maintaining the integrity of the data. The dataset was then sorted by the number of paintings by each artist, and only those with a larger number of works were retained for further processing.

5. Preprocessing data:

The pre-processing of the collected data is a key step in preparing the images for input into the ResNet50 architecture. The pre-processing process consists of the following stages:

5.1 . Resizing:

In order to better match the input size required for the ResNet50 model, we adjusted all images to 224x224 pixels. This resizing ensures consistency across the data set, which facilitates efficient batching during training.

5.2 . Normalization:

The pixel values of the image are re-scaled from the original [0, 255] range and normalized to the [0, 1] range. Normalization is standard practice in deep learning. It helps stabilize the training process and speeds up convergence by ensuring that the input data is at a consistent ratio.

5.3 . Data augmentation:

To solve the problem of class imbalance and enhance the generalization ability of the model, we adopt various data enhancement techniques. By creating modified versions of the original images, enhancement techniques artificially increase the diversity of the training set. The technologies used include:

● Horizontal and Vertical Flipping: Randomly flipping the images horizontally and vertically to introduce variability in the orientation of the artworks.

● Shearing: Applying random shearing transformations to the images to slightly distort them, mimicking variations in perspective.

Implemented using a class "ImageDataGenerator" by Keras, which applies these transformations during training.

Specific augmentation techniques such as width shift, height shift, zoom, and channel shift were deliberately excluded from the augmentation pipeline. Techniques such as width shift and height shift can crop out key parts of an image, changing the content and stylistic elements necessary to accurately identify the artist. Similarly, over-zooming may miss important parts of the artwork, or introduce artifacts that do not represent the original style.

Although these techniques often enhance a model's ability to recognize objects, they have no discernible effect on

understanding artistic styles. Especially in, say, abstract paintings, identifying objects can backfire. We would prefer the model to focus on understanding artistic style.
The main goal of this study is to capture the unique artistic style of each artist, and it is essential to maintain the integrity of the artwork. Therefore, it is crucial to maintain a balance between enhancement and content preservation to ensure that models learn to recognize an artist's unique style and are not misled by altered image features.

5.4 . Class Weights:

Since there is an imbalance in the number of works by each artist, class weights are calculated during the training process for under-represented classes. This approach ensures that the model is not biased towards more general classes and treats each class more fairly. Class weights are calculated based on the inverse frequency of each class and adjusted for the total number of samples. Finally, these class weights are added to the training process of the model to improve fairness and accuracy. Class weight for each = (total_paintings / (artists_number * current_artists_paintings)).

6. Model Architecture:

The architecture deployed in this study is based on the ResNet50 model, an efficient convolutional neural network (CNN) known for its deep residual learning framework. This architecture is particularly suitable for training very deep neural networks. It can alleviate the vanishing gradient problem by using residual blocks.

6.1 . Base Model:

The base model is ResNet50, pre-trained on the ImageNet dataset. ImageNet is a large visual database, designed for visual object recognition software research, containing more than 14 million images labeled with 1,000 different categories. By leveraging pre-trained models, we can leverage transfer learning, which allows us to take the model's knowledge gained from the ImageNet dataset and apply it to artist identification tasks.

6.2 . Adaptation for Artist Classification:

● Global Average Pooling Layer: A global average pooling layer was added in place of the top layers of the ResNet50 model. This layer reduces each feature map to a single value by averaging, which significantly reduces the output size and minimizes the overfitting
● Dense Layers: After the global average pooling layer, we added two dense (fully connected) layers.
○ In the first dense layer, we used a 'he_uniform' kernel initializer with 512 units. Next, we added a Dropout layer with a rate of 0.5, and Batch Normalization. Lastly, we added the "ReLU" activation function.
○ In the second dense layer, we used a 'he_uniform' kernel initializer with 16 units. Similarly, we added a Dropout layer with a rate of 0.5, Batch Normalization, and a "ReLU" activation function.
● Output Layer: The final layer is a dense layer with a softmax activation function, which provides a probability distribution over the five artist classes. This layer allows the model to output probabilities for each class, which makes it essential for multi-class classification problems.

```
X = base_model.output
X = keras.layers.GlobalAveragePooling2D()(X)  # Added a global average pooling layer
X = keras.layers.Dense(512, kernel_initializer='he_uniform')(X)
X = keras.layers.Dropout(0.5)(X)  # Reintroduce Dropout
X = keras.layers.BatchNormalization()(X)
X = keras.layers.Activation('relu')(X)

X = keras.layers.Dense(16, kernel_initializer='he_uniform')(X)
X = keras.layers.Dropout(0.5)(X)  # Another Dropout layer
X = keras.layers.BatchNormalization()(X)
X = keras.layers.Activation('relu')(X)

output = keras.layers.Dense(n_classes, activation='softmax')(X)
model = keras.Model(inputs=base_model.input, outputs=output)
```

7. Training Process:

The training process involves multiple key components to fine-tune the model, which enhances the performance of the model.
● Loss Function: Categorical cross-entropy was used as the loss function. This loss function is suitable for multi-class classification problems. It minimizes the difference between the predicted probability distribution and the actual distribution.
● Optimizer: The Adam optimizer is used with a learning rate of 0.0001. It is very efficient in dealing with sparse gradients in noise problems.
● Metrics: We set accuracy as our primary goal of this training.

```
model.compile(loss='categorical_crossentropy',
              optimizer=keras.optimizers.Adam(learning_rate=0.0001),
              metrics=['accuracy'])
```

● Callbacks: In order to enhance the training process, two callbacks were set:
○ EarlyStopping: This callback was configured with 20 epochs of patience and set to restore the best weights when training stops improving on the validation set. Once the model's performance on the validation set stops improving, it will stop training to prevent overfitting.
○ ReduceLROnPlateau: This callback was configured with 5 epochs of patience and reduces the learning rate by a factor of 10 when the validation performance plateaus, which helps in fine-tuning the model and achieving better convergence.

```
early_stop = keras.callbacks.EarlyStopping(
    patience=20,
    verbose=1,
    restore_best_weights=True
    )

reduce_lr = keras.callbacks.ReduceLROnPlateau(
    patience=5,
    verbose=1,
    )
```

● Training Strategy:
○ Initial Training:
■ 10 epochs.
■ The callback ReduceLROnPlateau was set.
■ All layers of the ResNet50 base model were unfrozen. The artist-identify task is significantly different from the image classification originally trained by ResNet50. ImageNet classification involves recognizing multiple common objects, but artist identity involves discerning subtle features and artistic nuances. By unfreezing all layers, the model can adjust its low-level feature detectors, such as edges, textures, and patterns. This allows the model to better adapt to the unique characteristics of the artwork.
○ Fine-Tuning:
After the initial training phase, the model was further fine-tuned to improve its performance while preventing over-fitting
■ Up to 50 epochs with the first 50 unfrozen layers, which allow for more specialized feature learning.
■ Both callbacks EarlyStopping and ReduceLROnPlateau were set.
■ Layers of the core ResNet50 were frozen to retain the pre-trained feature. Only the first 50 layers were kept trainable to allow fine-tuning.

8. Results:

The model has made significant advances in artist identification by applying transfer learning and fine-tuning techniques.
In the initial 10-epoch training phase:

Epoch	loss	accuracy	val_loss	val_accuracy	learning_rate
1/10	1.3634	0.3886	1.8488	0.1269	1.00E-04
2/10	1.0356	0.5883	1.7765	0.1629	1.00E-04
3/10	0.9271	0.6686	1.6149	0.1364	1.00E-04
4/10	0.8665	0.704	1.9846	0.1402	1.00E-04
5/10	0.8414	0.7299	2.4135	0.1439	1.00E-04
6/10	0.791	0.7441	1.6326	0.3409	1.00E-04
7/10	0.7439	0.7786	2.3939	0.2424	1.00E-04
8/10	0.7014	0.8069	1.3951	0.5152	1.00E-04
9/10	0.687	0.8211	0.6769	0.822	1.00E-04
10/10	0.6738	0.8069	0.5684	0.8731	1.00E-04

● The optimization of the model is very obvious. The validation accuracy increased from 12.69% in the first epoch to 87.31% by the final epoch.
● The training accuracy showed a steady increase from 38.86% to 80.69%.
● The loss function decreased from 1.3634 to 0.6738 for training data and from 1.8488 to 0.5684 for validation data.
● A notable jump in validation accuracy occurred between epochs 8 and 9, from 51.52% to 82.20%, which indicates a significant learning improvement.
The following fine-tuning phase allows up to 50 epochs with early stopping:

Epoch	loss	accuracy	val_loss	val_accuracy	learning_rate
1/50	0.4176	0.8895	0.5597	0.8504	1.00E-04
2/50	0.3867	0.8928	0.5164	0.8542	1.00E-04
3/50	0.3645	0.8966	0.5069	0.8731	1.00E-04
4/50	0.3622	0.8938	0.456	0.8769	1.00E-04
5/50	0.3662	0.8886	0.4785	0.8731	1.00E-04
6/50	0.3556	0.8867	0.427	0.8788	1.00E-04

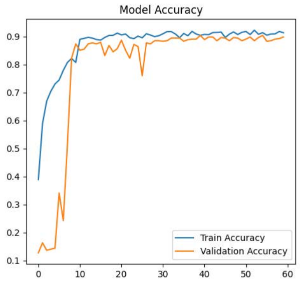
7/50	0.35	0.8961	0.5743	0.8314	1.00E-04
8/50	0.3249	0.9032	0.4525	0.8674	1.00E-04
9/50	0.3309	0.9037	0.4409	0.8447	1.00E-04
10/50	0.3023	0.9117	0.4541	0.8561	1.00E-04
11/50	0.3324	0.9051	0.3909	0.8864	1.00E-04
12/50	0.327	0.9084	0.4709	0.8504	1.00E-04
..	..	..	..	..	..
49/50	0.2692	0.9174	0.3702	0.892	1.00E-08
50/50	0.2928	0.9131	0.3567	0.8977	1.00E-08

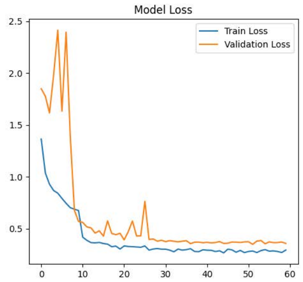
● Immediately boosted validation accuracy to 85.04% in the first epoch.
● The training accuracy rate increased steadily to 91.31%.
● The model converged after 50 epochs, with the learning rate reducing to 1e-8, indicating an optimization plateau.
● Overall Performance:
○ Final Training Accuracy: 99.58%.
○ Final Validation Accuracy: 90.77%.
○ The gap between training and validation accuracy suggests some overfitting, even regularization techniques applied.

9. Performance Evaluation:

9.1 . Learning Curves of Model:

Training and validation accuracy   training and validation loss

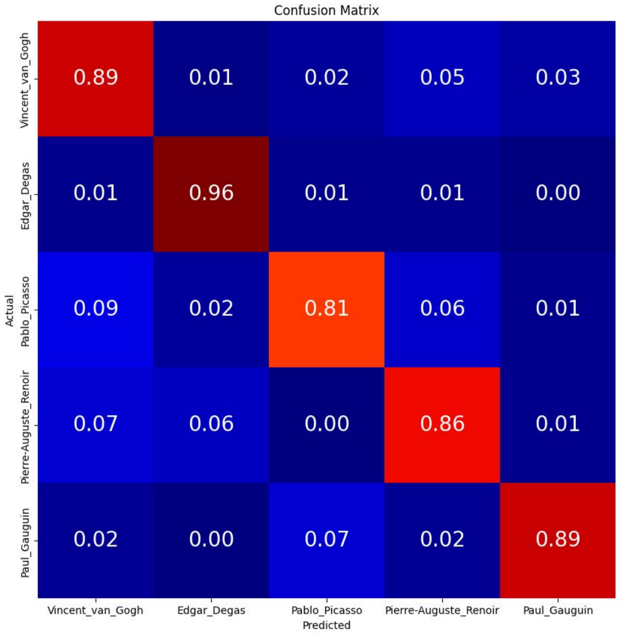




Accuracy trend:
Both the training and validation accuracy continued to improve over time. The training accuracy is near perfect. The accuracy of verification is stable at about 90%.
Loss trend:
Training loss is steadily declining. Validation loss shows an uneven but overall downward trend, especially in the early stages. This pattern is common in deep learning models. It means that the model is learning meaningful features, but occasionally overfits the training data.

9.2 . Confusion Matrix:

The confusion matrix provides a detailed analysis of the performance of the model in different classes(artists). It shows how often the model correctly identifies the artist of the artwork and highlights any misclassifications.



The model maintained a very high accuracy rate for the works of three artists (Vincent van Gogh, Edgar Degas, and Paul Gauguin)
The accuracy of the model for the works of the two artists (Pablo Picasso and Pierre-Auguste Renoir) is relatively lower.
● Pablo Picasso 84%:
This may be because most misclassifications confuse the works of Pablo Picasso with the works of Van Gogh or Renoir.
● Pierre-Auguste Renoir 84%:
This may be because most misclassifications confuse the works of Pierre-Auguste Renoir with the works of Van Gogh and Degas.

9.3 . Classification Report:

The classification report provides precision, recall, and F1-score for each artist:

Classification Report:

	precision	recall	f1-score	support
Vincent_van_Gogh	0.91	0.89	0.90	164
Edgar_Degas	0.95	0.96	0.95	141
Pablo_Picasso	0.87	0.81	0.84	81
Pierre-Auguste_Renoir	0.79	0.86	0.83	72
Paul_Gauguin	0.88	0.89	0.89	57
accuracy			0.89	515
macro avg	0.88	0.88	0.88	515
weighted avg	0.89	0.89	0.89	515

Precision: Ranges from 0.79 (Renoir) to 0.95 (Degas). It is the percentage of all correct positive predictions.

Recall: Varies from 0.81 (Picasso) to 0.96 (Degas). It is the percentage of positive predictions that are correct among all actual positive results.

F1-scores: Range from 0.83 (Renoir) to 0.95 (Degas). It is the harmonic average of precision and recall provides a single metric to balance precision and recall.

Overall accuracy: 89%. This indicates high accuracy across all classes.

10. Conclusion:

Through this study, the methods of deep learning and transfer learning have shown promising results in the field of art recognition. By using the pre-trained ResNet50 architecture, the model achieves an extremely high accuracy in the recognition of artwork of different artists. This method effectively solves multiple challenges within the field. For example, the subjectivity of traditional methods, the diversity of art styles, and the inherent class imbalance of artwork datasets. Key findings:

● High accuracy and robustness:

The final verification accuracy of the model reached 90.7%. It indicates that the model has a high accuracy in most artist classes. This extremely high performance shows the robustness of the CNN architecture in handling the artistic style.

● Effectiveness of Transfer Learning:

Through this study, transfer learning can save time and resources by using the pre-trained ResNet50 model and then fine-tuning it on specific datasets. Transfer learning significantly reduces the training time and improves the overall performance of the model. It shows the effectiveness of transfer learning in extracting relevant features from complex images.

● Handling Data Imbalance:

Data augmentation and class weighting can significantly alleviate the problem of class imbalance. These methods ensure that the model is not biased towards classes that have more or less data. Therefore, it improves its fairness and accuracy among different artists.

While the current model shows high accuracy, there are several ways to enhance its functionality for future research.

● Integrating multi-modal data:

By adding other types of data, the model can have a richer understanding of the artwork and improve the classification performance. For example, datasets include textual descriptions of the artwork, the size of the artwork, or the texture of the artwork.

● Explore data sets:

Expand the model to handle more artist classes and larger data sets. This could make deep learning more useful in classifying artworks and their related fields.

● Explore different architectures:

There are also other deep learning architectures we can experiment with, such as Vision Transformers or newer variants of CNN. Different models may get better results in unexpected aspects because of their different network architectures.

Finally, this study highlights the potential of deep learning and transfer learning in automating the artist identification process. This approach mitigates the challenges of data scarcity and imbalance. The results show that well-suited techniques can build art classification models with high accuracy. For future work, we will try to further explore new techniques on this basis and extend the model to a wider range of datasets and classes of artworks and artists.

References

[1] L. A. Gatys, A. S. Ecker, and M. Bethge, "A Neural Algorithm of Artistic Style," arXiv.org, 2015. https://arxiv.org/abs/1508.06576

[2] N. Viswanathan, "Artist Identification with Convolutional Neural Networks." Accessed: Jul. 20, 2024. [Online]. Available: https://cs231n.stanford.edu/reports/2017/pdfs/406.pdf

[3] Icaro, "Best Artworks of All Time," *Kaggle.com*, 2019. https://www.kaggle.com/datasets/ikarus777/best-artworks-of-all-time/data (accessed Jul. 20, 2024).