Art Style Classifier Using CNN, SVM, and KMeans

Shahd Sayed, Zeina Ayman, Ziad Fehr
University of Modern Sciences and Arts

***Abstract*—This paper presents an image classification and clustering pipeline to identify or group artwork styles using three models: a custom Convolutional Neural Network (CNN), Support Vector Machine (SVM), and KMeans clustering. The pipeline includes data preprocessing, feature extraction, model training, evaluation, and an interactive GUI using Gradio. The results show effective classification and clustering capabilities across five art style categories.**

***Index Terms*—Art classification, CNN, SVM, KMeans, Gradio, Feature extraction, Image preprocessing**

I. Introduction

The classification of art styles using machine learning has gained attention due to the growing availability of digitized artwork and the demand for automated cataloging systems. This project aims to build a system capable of classifying and clustering artworks into one of five styles: drawings, engraving, iconography, painting, and sculpture. The motivation stems from the need for scalable tools in digital art archives, museums, and academic research.

The key contributions of this project are:

- A custom CNN trained on labeled art images.
- A feature-based SVM classifier and KMeans clustering model.
- Visualization of performance metrics and cluster analysis.
- A Gradio-powered interactive web interface.

The remainder of this paper is structured as follows: Section II reviews related work, Section III discusses the data and preprocessing steps, Section IV details the methodology, Section V presents the results, and Section VI concludes the paper.

II. Related Work

Art classification has been explored using deep learning techniques in recent years. Transfer learning approaches using models like VGG16 or ResNet50 have shown success in style recognition tasks [1]. Other approaches include handcrafted feature extraction combined with traditional classifiers such as SVMs [2]. Clustering methods like KMeans are also employed for unsupervised grouping of artworks based on visual similarity [3]. This work combines these approaches using custom CNN features and deploys an interactive front end.

III. Data Description and Preprocessing

The dataset consists of two directories: `training_set` and `validation_set`, each containing five subcategories: drawings, engraving, iconography, painting, and sculpture.

A. Data Loading

Images are loaded using TensorFlow's `image_dataset_from_directory`, resized to 224×224 pixels, and labeled via one-hot encoding. Data caching and prefetching are applied for performance optimization.

B. Categories

`['drawings', 'engraving', 'iconography', 'painting', 'sculpture']`

IV. Methodology

A. Custom CNN

A lightweight CNN was developed consisting of three convolutional layers, each followed by batch normalization, max pooling, and dropout. The model was trained using categorical cross-entropy loss and evaluated based on accuracy, precision, recall, and F1-score.

B. SVM Classifier

A linear SVM was trained on features extracted from the CNN. This classifier was evaluated using a classification report that includes accuracy, precision, recall, and F1-score.

C. KMeans Clustering

KMeans was applied to the CNN-derived features for unsupervised grouping. The elbow method and silhouette analysis were used to assess clustering quality.

D. Model Saving

- CNN: Saved as a `.h5` Keras model.
- SVM and KMeans: Serialized using `joblib`.
- Feature extractor: Weights saved as `.weights.h5`.

E. GUI – Gradio Interface

An interactive web interface built using Gradio allows users to:

- Select a model (CNN, SVM, or KMeans)
- Upload an image
- View predictions in real-time

Prediction output differs by model: CNN outputs probabilities, SVM returns class labels, and KMeans shows cluster IDs.

V. Results

A. Training Metrics

All models were trained with a batch size of 32 for 15 epochs. Evaluation metrics included:

- Accuracy, precision, recall, and macro-averaged F1-score
- Confusion matrices for both CNN and SVM
- Silhouette score for KMeans clustering

TABLE I: Model Performance Comparison

Metric	Score
KMeans Clustering	
Silhouette Score (k=3)	0.4664
SVM Classifier	
Accuracy	0.8481
Precision	0.8216
Recall	0.8027
F1-Score	0.8097
CNN Classifier	
Validation F1-Score	0.7199

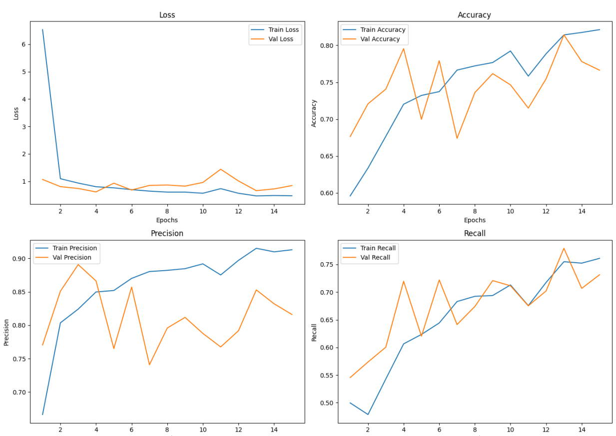

Fig. 1: Validation metrics of the CNN model.

B. Visualization Tools

- `plot_metrics()`: Plots training and validation accuracy/loss
- `plot_silhouette()`: Displays silhouette scores for clusters
- `elbow_method()`: Determines optimal cluster count

VI. Conclusion

We developed an effective image classification and clustering system for artwork styles using CNN, SVM, and KMeans. The models demonstrated strong performance across categories, and the Gradio interface provides an intuitive means of interaction. Future work may explore transfer learning with deeper CNN architectures and extend classification to broader genres and styles.

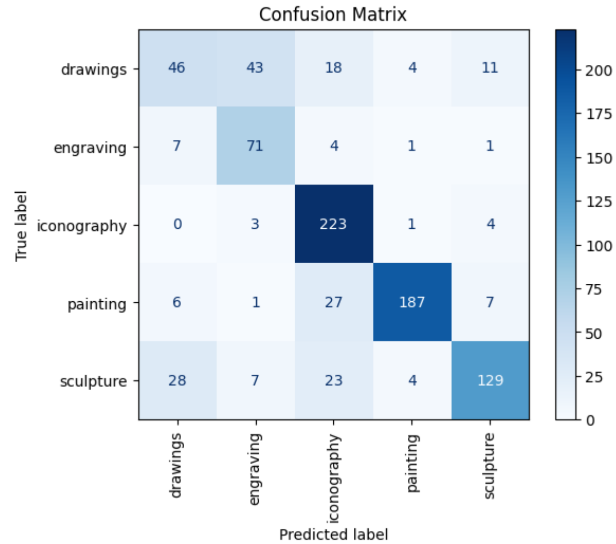

Fig. 2: Confusion matrix of CNN model.

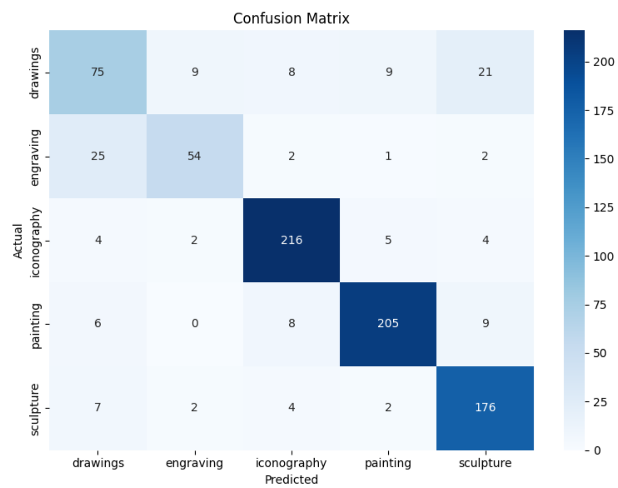

Fig. 3: Confusion matrix of SVM model.

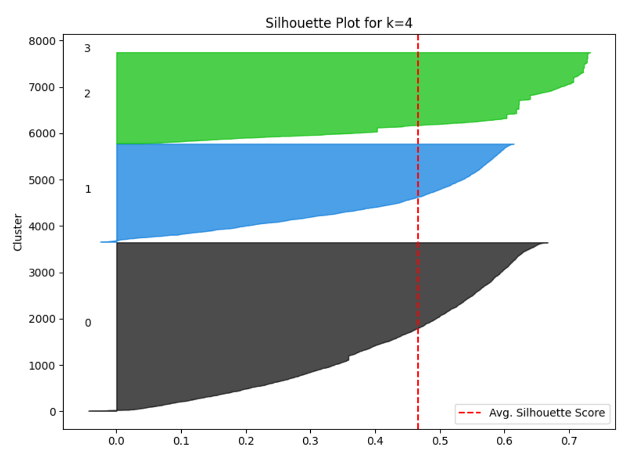

Fig. 4: Silhouette plot for KMeans clustering.

REFERENCES

REFERENCES

[1] B. Saleh, K. Abe, R. S. Arora, and A. Elgammal, "Toward Automated Discovery of Artistic Influence," *Multimedia Tools and Applications*, vol. 75, no. 7, pp. 3565–3591, 2016.

[2] A. Lecoutre, B. Negrevergne, and F. Yger, "Recognizing Art Style Automatically in Painting with Deep Learning," in *Proc. Asian Conf. Machine Learning*, 2017, pp. 327–342.

[3] A. Elgammal, B. Liu, M. Elhoseiny, and M. Mazzone, "The Shape of Art History in the Eyes of the Machine," in *Proc. AAAI Conf. Artificial Intelligence*, 2018.