Painter Identification Based On Their Paintings

MSc Research Project
MSc in Data Analytics

Priya Prakashbhai Patel
Student ID: x20119097

School of Computing
National College of Ireland

Supervisor: Jorge Basilio

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Priya Prakashbhai Patel
Student ID:	x20119097
Programme:	MSc in Data Analytics
Year:	2021
Module:	MSc Research Project
Supervisor:	Jorge Basilio
Submission Due Date:	16/12/2021
Project Title:	Title
Word Count:	
Page Count:	25

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	30th January 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Painter Identification Based On Their Paintings

Priya Prakashbhai Patel
x20119097

Abstract

With the digitization of paintings and creation of vast online libraries containing paintings of different painters the classification of paintings has never been more important than now. To do these various techniques are used such as computer vision and Artificial intelligence are used. Previously the classification of painters was done by experts in the field but in this paper tries to classify these painters by Implementing Convolution Neural Network (CNN) models. Also, the work that is already done is in the classification of paintings is vastly based on the style of paintings rather than the painters that created then. So, this paper will focus on the classification of painters. There are five models implemented and compared in the paper. Out of all models pretrained ResNet50 with imagenet weights as base weights and adding more layers on it had the highest accuracy of 85.38 percent.

Keywords - Convolution Neural Network, imagenet, Classification, Alexnet, ResNet50

1 Introduction

1.1 Background Research

Nowadays a lot of work is done in the digitization of artwork form their physical form. High-definition images of artworks are available online of everyone from all kinds of artist(Agarwal et al.; 2015). These works are then classified into genre, style, and other classes. But this work done in classification is slow and tedious. Hence, most of these artworks are still not classified yet. This shows that an automated system is required to make the process quicker and more efficient.
In these few years, due to efforts by everyone there have been some formation of digital libraries(Kelek et al.; 2019a). There is information on the genre, style, and artist of each of these pieces of art. Experts usually identify this information based on the artists, genre, or style that they are familiar with. But there is still a lot of information online to still be classified. Some art historian noted that there is metadata connected with these artworks(Saleh and Elgammal; 2015).
Machine learning approaches including image identification, object detection, and feature extraction have gained a lot of momentum in the field of computational vision in recent years(Khan and Al-Habsi; 2020). Convolution neural network (CNN) is a deep learning technique that is widely and popularly used in image processing, and it have developed models and related libraries that give the best results in the field of image processing (Seo and Shin; 2018).

1.2 Motivation

The major motivation for doing this project is to make a machine learning model and to test and compare with already made models to find the best models to make the task of automation to classification of painters faster. As, currently this is done by industry professional which takes a very long amount of time. Making this task faster will make the creation of online libraries faster and these artworks will be available to everyone and that will help in conservation and preservation of these artworks.

1.3 Research Question

"To what extent can Machine learning identify and predict a painter based on their paintings using Convolution neural network (CNN) models"

1.4 Objectives

The major objective of this project is to build and compare machine learning models for the classification and prediction of painters based on paintings. Some other objectives are to critically review papers and works that are done in the same or similar field.

1.5 Contribution

The major contribution of this project is towards the help of creation of an online library with their associated details. Other is to evaluate and compare different Machine learning models to find the best models to classify the paintings based in their painters.

2 Related Work

Researchers have been experimenting with computer vision as technology has advanced. This involves computer vision-based painting categorization work. However, because traditional machine learning models cannot extract exact information about the paintings due to a lack of brush stroke expertise, the role performed by them becomes irrelevant (Johnson et al.; 2008). In a paper by (Banerji and Sinha; 2016) they discussed the use of cnn to extract the features. And use of all kinds of layers to identify the beat of them to be used in the classification of paintings. Kim et al. (2019) research was centered on the visualization of information about the depth of brushstrokes. Because brushstrokes are so important, it was a very useful parameter that might aid improve the model's accuracy. They developed a novel data use technique called Reflectance Transformation Imaging (RTI).
(Nunez-Garcia et al.; 2018) has also researched the categorization of painting genres. His work is centered on extracting the most important aspects of a picture. By combining the characteristics in the framework of an algorithm, the study employs Artificial Neural Networks (ANNs) to classify paintings into seven distinct genres.(Pirrone et al.; 2009) worked on their study using the WikiArt dataset. Style, artists, genre, date, and other details are included in the WikiArt dataset. Converting visual data to numerical data is one of the most difficult aspects of image processing. Several techniques, including HOG, SIFT, and CNN-derived features, were employed to accomplish this. After transforming the data to numerical form, multiple methods were utilized to extract the characteristics

from the data.
(Arora and Elgammal; 2012) used many categorization algorithms to automate the classification of fine art categories. This problem was solved by unsupervised learning. They employed intermediate features called bag of words instead of low-level features such as color, contrast, and brightness (BOW). SVM classifier was utilized in this research. (Liu and Jiang; 2014) divided traditional Chinese paintings into two groups. School of free style sketch and school of careful art were the two classes. They employed feature extraction approaches to recognize and classify images using SVM. For feature extraction, categorization, and data production, they employed a variety of methodologies.
In order to complete this challenge, (Chen et al.; 2017) applied supervised learning approaches. He did this by using color characteristics and multi-view appearance on the Mogao Grottoes dataset, and he was able to outperform others. They developed a hypothesis based on the idea that a painting style and era may be determined using multi-view and color attributes. (Zujovic et al.; 2009) undertook a similar endeavor of categorizing paintings by genre. This categorization was based on two basic characteristics. Color values (Hue, Saturation, and Value) and Gray-scale characteristics (Gabor filters and edges) were the features in question. This study employed both Artificial Neural Networks (ANNs) and classic approaches (SVM, KNN, Nave Bayes, AdaBoost). The use of many features produced a better result, although each iteration increased the runtime. Although the findings vary owing to the dataset's volatility in terms of size and quality, Genetic algorithms (GAs) based on the Weighted Nearest Neighbor (WNN) classifier and deep RBMs outperform other classifiers, according to (Levy et al.; 2014). And it has a precision of up to 90 percent.(Jangtjik et al.; 2017)te used Long Short Term Memory to divide the pictures into numerous patches in order to analyze the correlation between the patches (LSTM). Their research focuses on categorizing writers based on the photographs of their paintings that are available online. They presented a CNN-LSTM model that would assign numerous labels to a given image, as well as a fusion approach that would determine the quality of each layer of patches. These two approaches were combined to provide a unique result.
In a research by (Cetinic and Grgic; 2013), the style of an individual artist was identified by extracting unique traits of distinct paintings. Support Vector Machine (SVM), Nave Bayes, Multi-layer Perceptron, and Random Forest were utilized to find measurable aspects of an image that were represented by a collection of global image attributes. Each artist's painting was represented by 25 pictures in the dataset. Using MLP, they were able to reach approximately 75 percent of accuracy. The CNNs were used to classify Chinese ink wash painting (IWP). Brushstrokes were extracted using CNN. This was used to replace the frequently utilized way of analyzing color and edges. In a study by (Sun et al.; 2015).
(Kelek et al.; 2019b) addressed the use of Deep Neural Networks to identify painters using online photographs of paintings. Using DenseNet, they were able to attain an accuracy of roughly 80 percent. In addition to GoogleNet, ResNet50, ResNet101, and Inceptionv3, they employed ResNet50, ResNet101, and Inceptionv3. However, they only employed a tiny dataset of 46 paintings by 17 different painters. A bigger dataset is known to improve the performance of CNNs. As a result, this study project will need a huge dataset of roughly 500 photographs.

3 Methodology

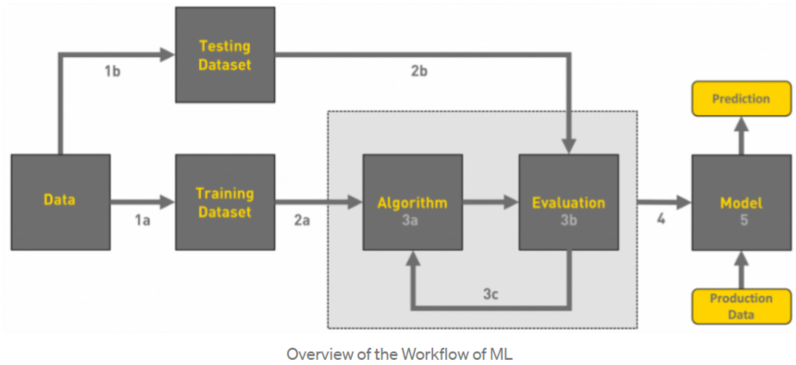


Figure 1: workflow diagram

The steps used in the implementation of this project are mentioned and explained below:

- **Step 1: Data Collection**
  The dataset selected for this research project is "Best Artwork of All Times" from Kaggle. It contains 8446 images from 50 different artists and an Artist table containing information about those artists.

- **Step 2: Data Preprocessing**
  Before using the raw data, preprocessing needs to be done.
  First the unnecessary columns are removed. They are id, bio, and Wikipedia.
  After that the years column is modified into birth year, death year, and their age.
  Using the age column, the age of artist is binned into four age groups young adult, early adult, adult, and senior.
  After this class weight is assigned to each painter based on their number of paintings.
  Next all the spaces in artist names are replaced with underscore to make all the names uniform.

- **Step 3: Exploratory Data Analysis**

  - The graph below shows the nationality of the painters present in the dataset. It can be inferred that the majority of the painters were French.

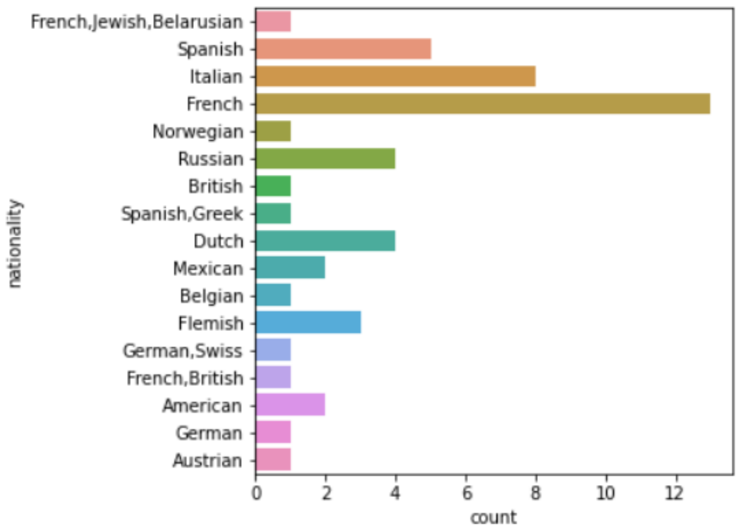


Figure 2: Nationality of painters

– The graph below shows the number of paintings by artist. From that we can see that Vincent Van Gogh has the highest and Jackson Pollock has the least number of paintings in the dataset.

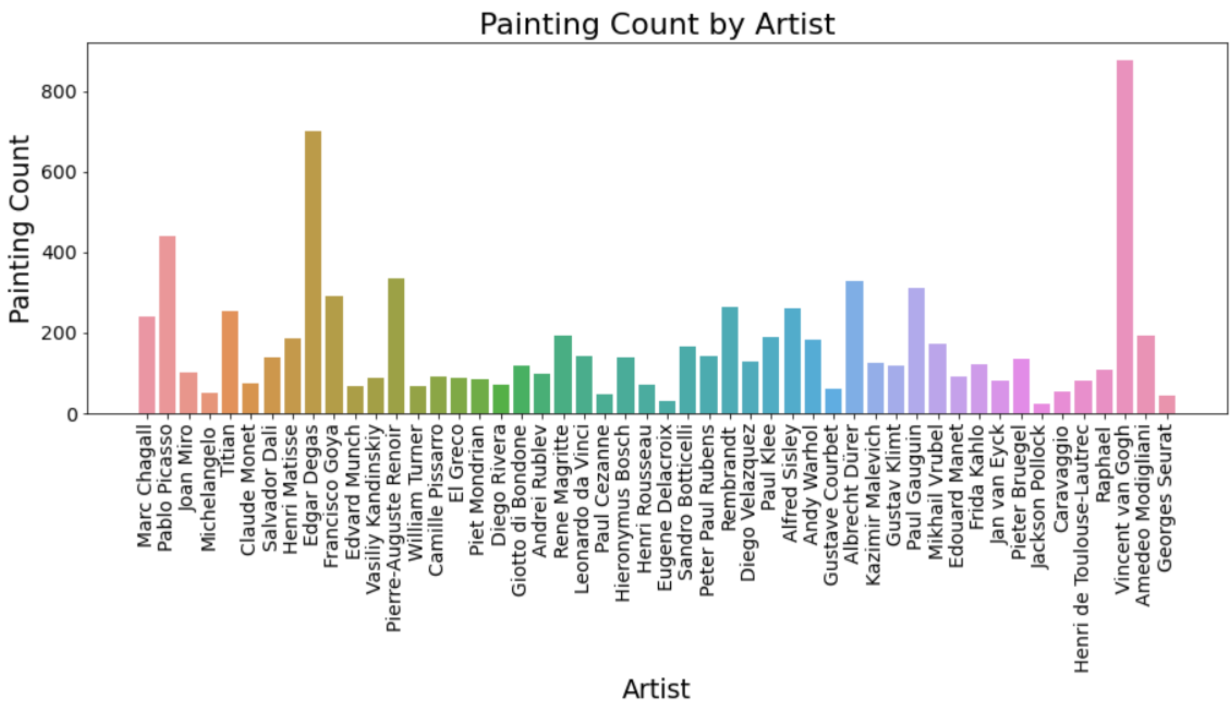


Figure 3: Number of Paintings

– The graph below the bar plot of genres of paintings. It shows the frequency of a genre occurring in the dataset. From that genre Post-Impressionism, Impressionism, Baroque, and Northern Renaissance are the most frequent genre.

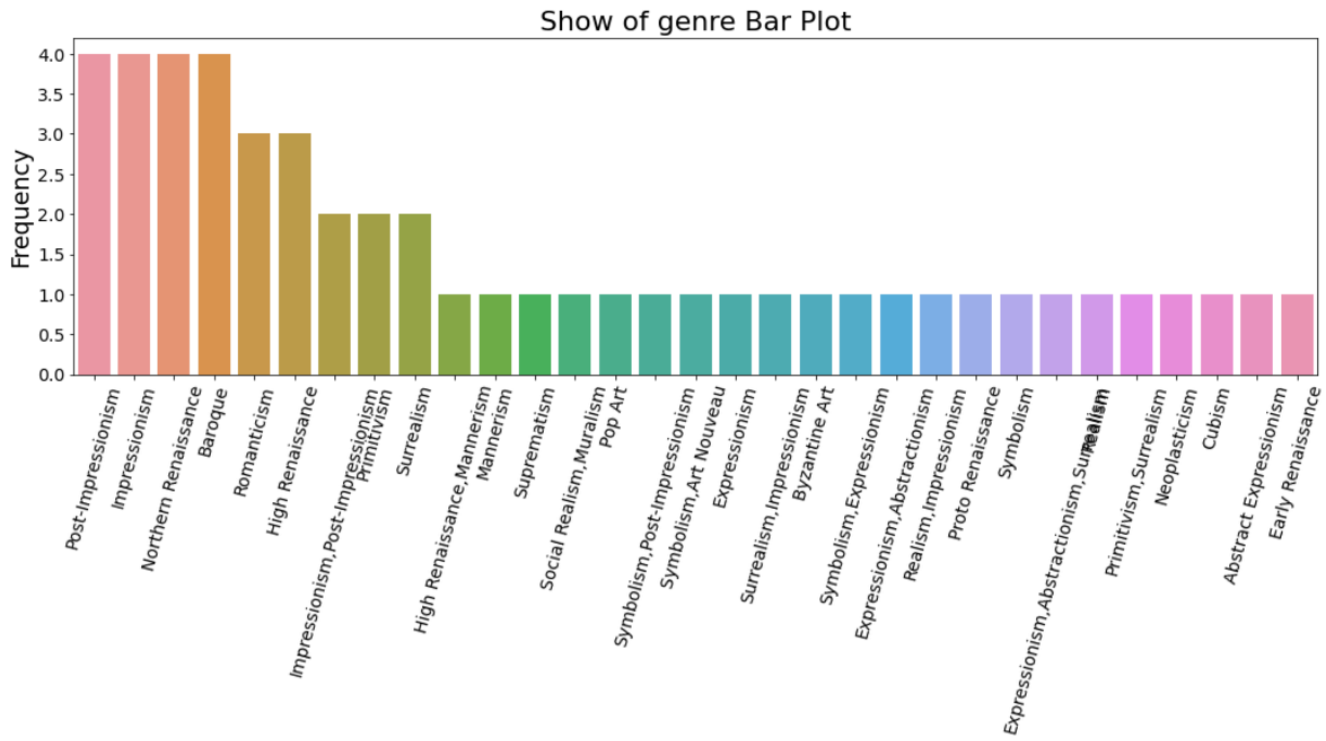


Figure 4: Nationality of painters

- The graph below gives the analysis of age group which shows that all the groups are evenly distributed. With adult and early adult having less percent than young adult and senior

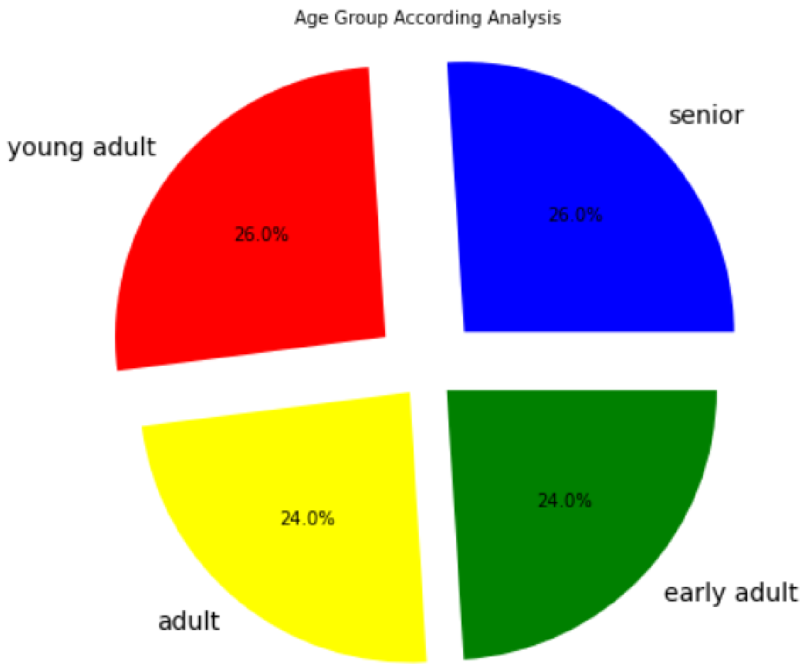


Figure 5: Painters Age Groups

– The graph below shows the density plot of ages of painters. It shows that most of the painters are around 70 years of age.

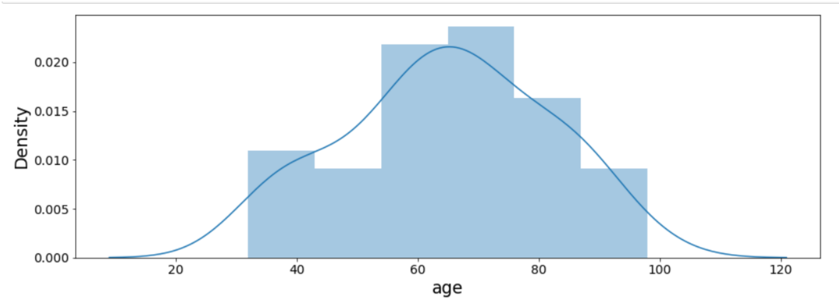


Figure 6: Painters Age Density

– The graph bellow shows the violin plot of the age group of paintings.

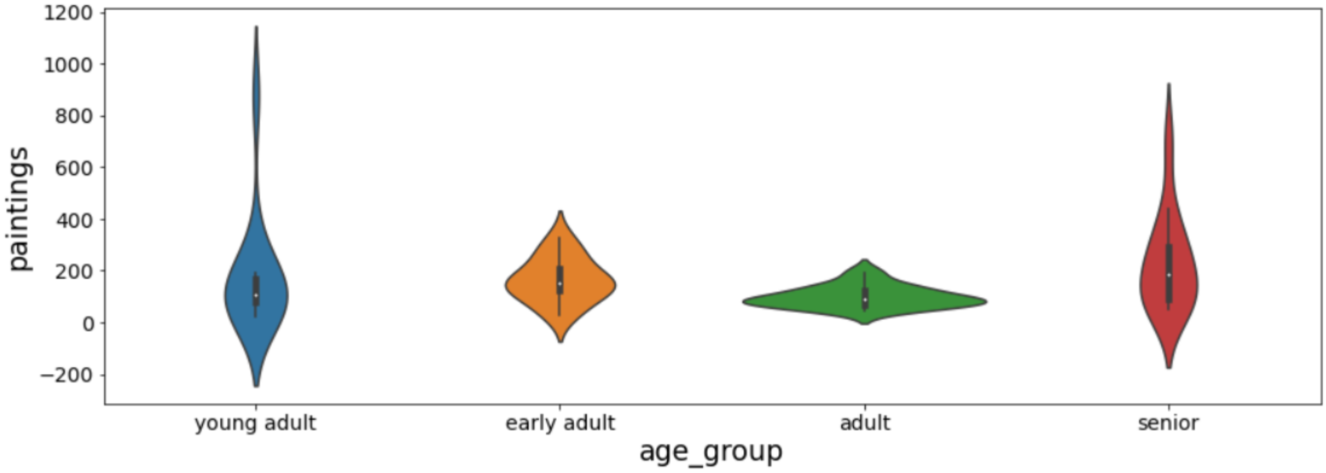


Figure 7: Paintings age groups

– The next plot shows the count of painting according to their genre and age group. From that certain style of paintings were more famous with different age groups.

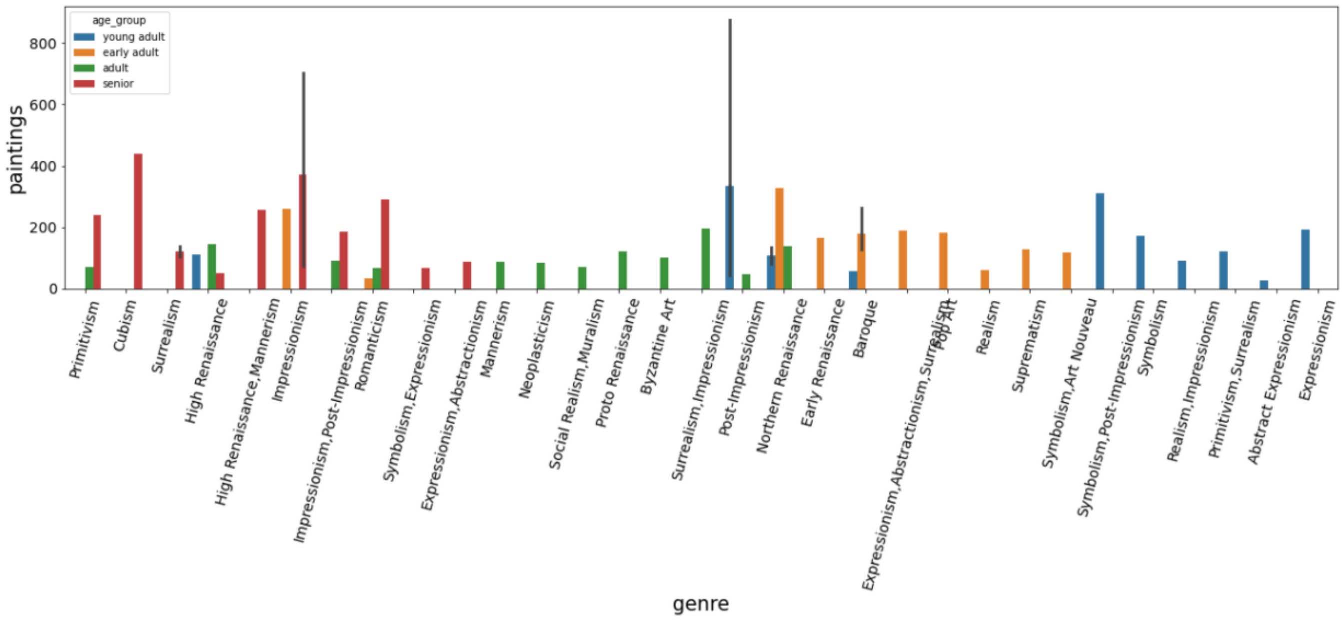


Figure 8: Paintings vs Genre

– The graph below shows the bar chart of paintings and age group according to the style of their paintings.

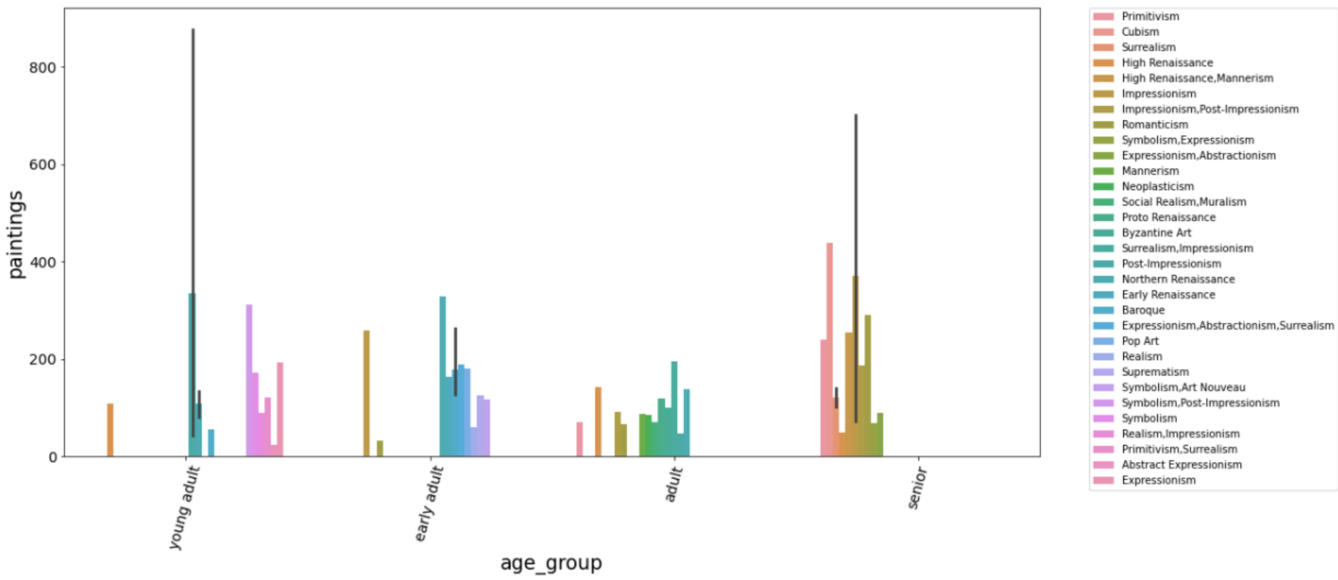


Figure 9: Paintings Vs Age group

– The graph below shows the violin plot of paintings and genre according to the nationality of the painters. It can be inferred that french painters prefer impressionism genre of paintings.

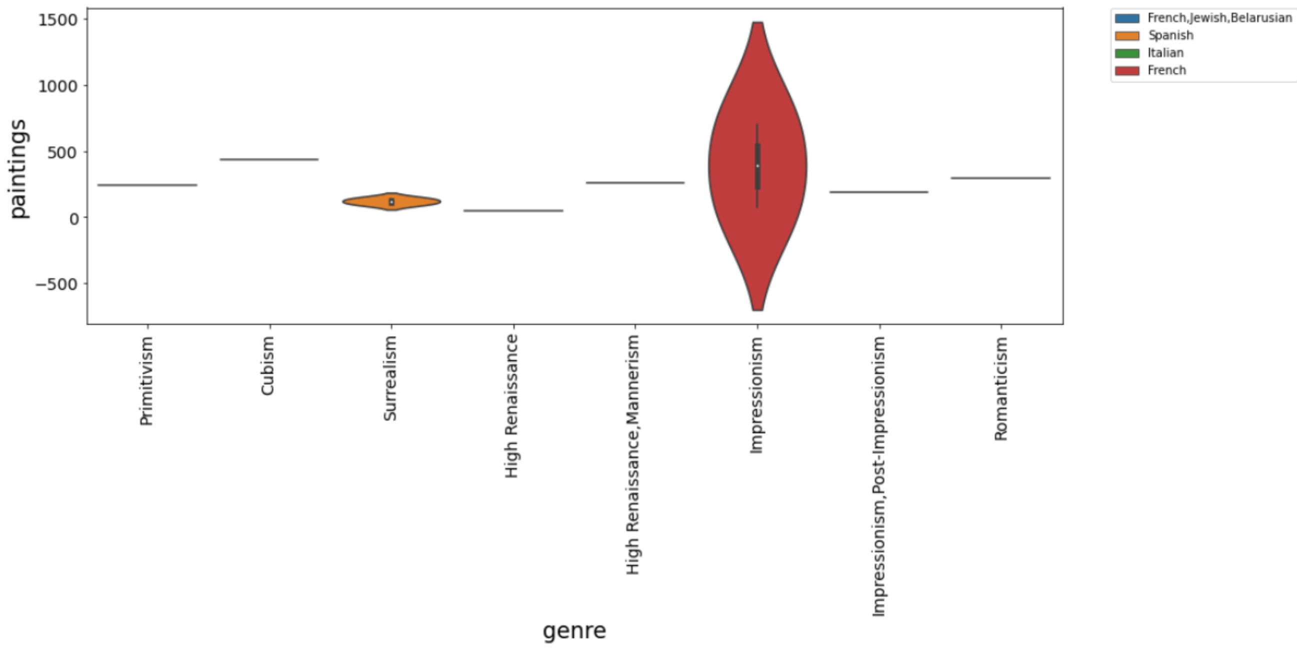


Figure 10: Genre of paintings

– The graph bellow shows the paintings Kde Plot System Analysis. It shows that the most frequent values are around 100-150.

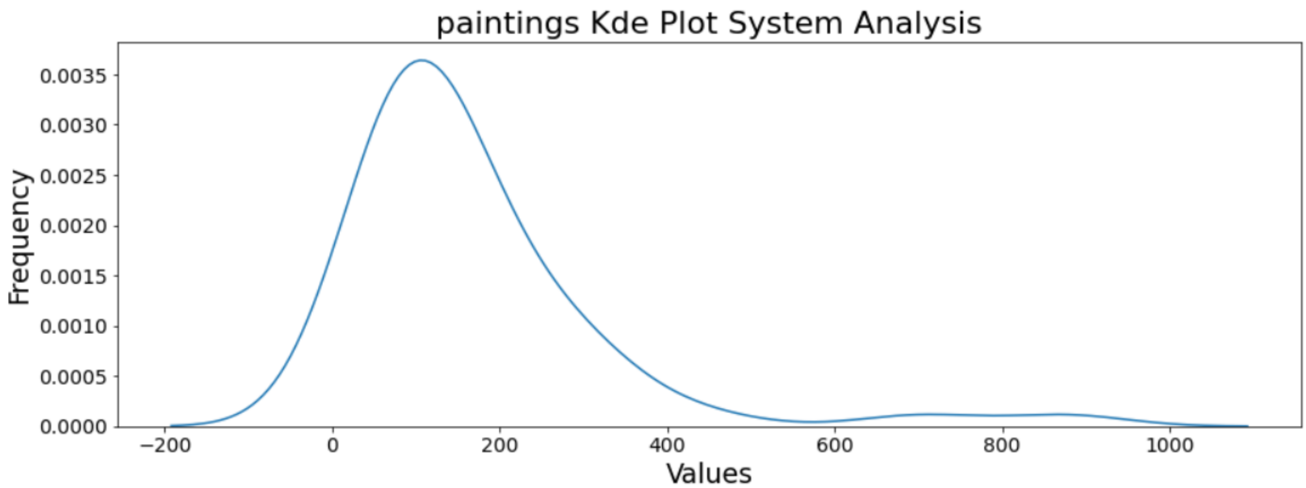


Figure 11: Paintings KDE Analysis

- **Step 4: Finding the correct Machine learning models that best suits the data**
  To find the models to be used one needs to find the nature of the data first. Here our data is in image form. Once this is done what kind of problem needs to be solved is to be known. Here is the problem of classification. Then models need to be researched. For this project Deep learning CNN models are used as they are capable at feature extraction in the hidden layers before passing through a dense layer to perform classification.

- **Step 5: Training and testing all the models used**
  Once the data is preprocessed and all the models are selected to be applied. The data needs to be divided into 3 groups for training and testing. These groups are 'training data', 'validation data' and 'testing data'.
  During the training phase of the process training and validation data is used and the testing data should only be used to test the efficiency of the models.

- **Step 6: Evaluation of models**
  There are multiple ways to evaluate models. Before evaluating these models some key terms are to be known. These terms are described below:

  - **True Positive (TP):** These are the values that are positive and predicted positive.
  - **False Positive (FP):** These are the values that are negative and predicted positive.
  - **True Negative (TN):** These are the values that are negative and predicted negative.
  - **False Negative (FN):** These are the values that are positive and predicted negative.

  Some keyways to evaluate a model are described below:

  - **Accuracy:** It is the most famous way to evaluate a model. It is the ratio of all true prediction and all the observations. It shows how many times does the model predict the answer correctly

    $$\frac{TP + TN}{TP + FP + TN + FN}$$

  - **Precision:** It is the ratio of all the positive outcomes of the models i.e. all the correct positive prediction by all the positive prediction.

    $$\frac{TP}{TP + FP}$$

  - **Recall/Sensitivity:** It is the ratio of positive instances i.e. all correct positive prediction by all positive observation in the dataset.

    $$\frac{TP}{TP + FN}$$

– **Specificity:** It is the ratio of negative instances i.e. all correct negative prediction by all negative observation in the dataset.

$$\frac{TN}{TN + FP}$$

– **F1 score:** It is the harmonic mean of precision and recall

$$\frac{2}{\frac{1}{precision} + \frac{1}{recall}} = \frac{2 * precision * recall}{precision + recall}$$

4 Design Specification

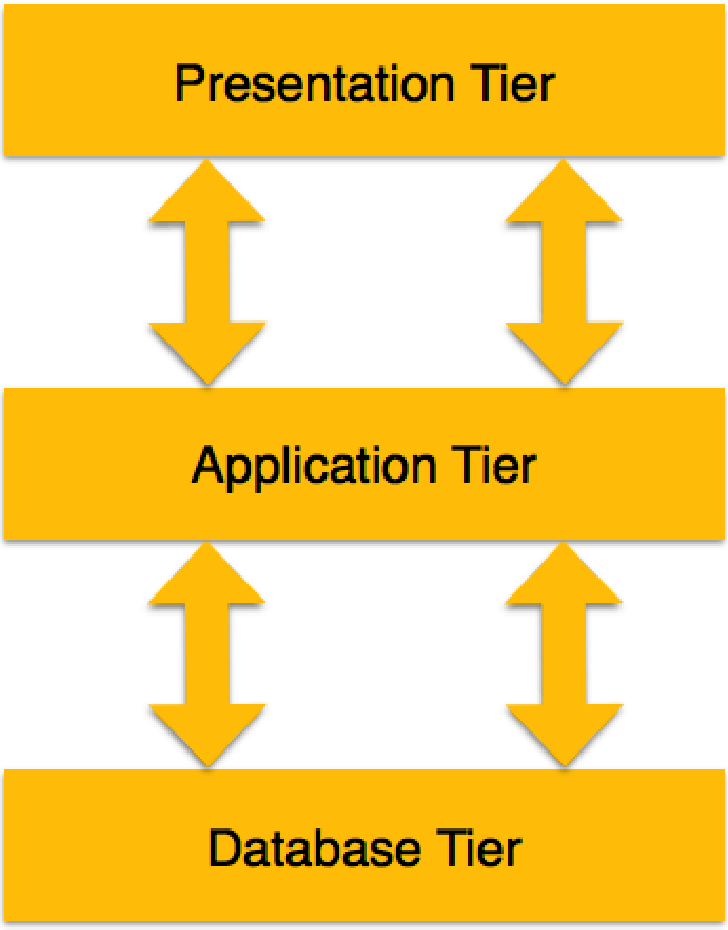


Figure 12: Three Tier Arhitecture

The architecture used for this project is 3-tier architecture. Where each layer represents different areas of project. These layers are discussed below:

- **Tier 1: Presentation tier**
  It is the final tier in the in the architecture. Here the final output is presented using visualization tools and libraries. Libraries like matplotlib, plotly, seaborn are used in this layer to have visually pleasing graphs and charts. Some prediction percent and some actual predictions are also shown in this layer.

- **Tier 2: Application tier**
  Also, known as Business logic tier or logical tier. Here all the logical processing is done. All the data processing, data augmentation and CNN models are applied to the data. Libraries like tensorflow, keras, etc. are used

- **Tier 3: Database tier**
  In this tier data files are stored. The application tier interacts with this tier to obtain the data needed. Paths to these files are given in the code to connect to these files.

5 Implementation and Evaluation

You will of course want to discuss the implementation of the proposed solution. Only the final stage of the implementation should be described.

It should describe the outputs produced, e.g. transformed data, code written, models developed, questionnaires administered. The description should also include what tools and languages you used to produce the outputs. This section must not contain code listing or user manual description.

5.1 Model 1: Convolution Neural Network(CNN)

5.1.1 Implementation

CNN is one of the most Used Image classification Algorithm now. It is because it is very good ad feature extraction through the hidden layers in the model. On leach layer some features are extracted and thus it does a very good job at image classification. Before implementation if this model some data augmentation is done. For these images are randomly flipped, rotated, zoomed, translated, and changed their contrast. The figure shows the augmentation of the data.

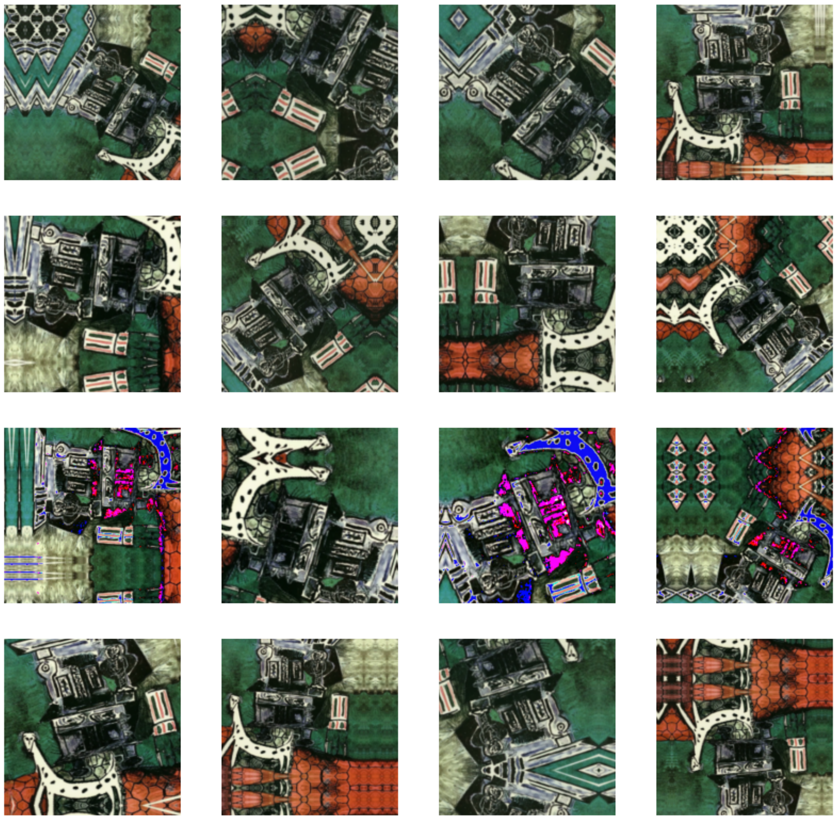


Figure 13: Image Data Augmented

The model summary shows the model outline. For the activation function in dense layer Rectified Linear unit (relu) is used. With momentum of 0.1 RMSprop optimizer is also used. To calculate the loss SparseCategoricalrossentropy function is used.

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 sequential (Sequential)     (None, 300, 300, 3)       0

 rescaling_1 (Rescaling)     (None, 300, 300, 3)       0

 conv2d (Conv2D)             (None, 300, 300, 16)      448

 max_pooling2d (MaxPooling2D  (None, 150, 150, 16)     0
 )

 conv2d_1 (Conv2D)           (None, 150, 150, 32)      4640

 max_pooling2d_1 (MaxPooling  (None, 75, 75, 32)       0
 2D)

 conv2d_2 (Conv2D)           (None, 75, 75, 64)        18496

 max_pooling2d_2 (MaxPooling  (None, 37, 37, 64)       0
 2D)

 dropout (Dropout)           (None, 37, 37, 64)        0

 flatten (Flatten)           (None, 87616)             0

 dense (Dense)               (None, 128)               11214976

 dense_1 (Dense)             (None, 51)                6579

=================================================================
Total params: 11,245,139
Trainable params: 11,245,139
Non-trainable params: 0
_________________________________________________________________
```

Figure 14: Model Summary

5.1.2 Evaluation

For the evaluation of the model Accuracy was taken in consideration. from that we got an accuracy of 38.98 percent and validation accuracy of 28.66 percent.

```
212/212 [==============================] - 292s 1s/step - loss: 2.2490 - accu
racy: 0.3898 - val_loss: 2.7218 - val_accuracy: 0.2866
```

The graphs bellow shows the training accuracy and loss function of the model. From that we can see that the Accuracy in increasing while the loss is decreasing.

5.2 Model 2: Alexnet

Alexnet is a deep convolution neural network that has 11 layers of which 5 are convolution and 3 are fully connected layers.
The Overlapping Max Pooling layers, come after the first two Convolutional layers. Direct connections exist between the third, fourth, and fifth convolutional layers. The output of the fifth convolutional layer is fed into a sequence of two fully connected layers through an Overlapping Max Pooling layer. Relu activation function is then applied to fifth layer. Normalization layer is also applied to second convolution layer.

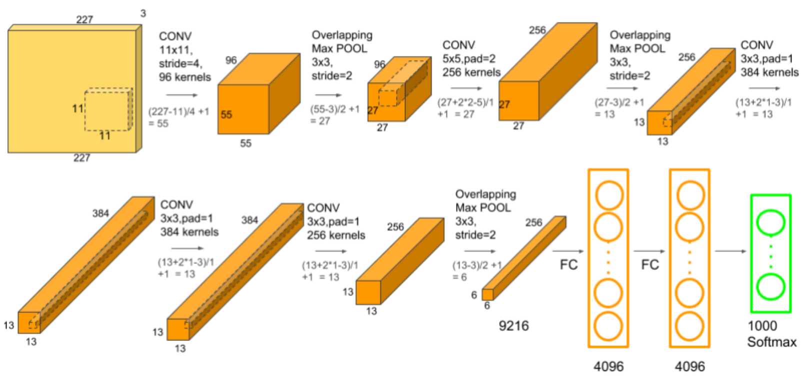


Figure 15: AlexNet architecture

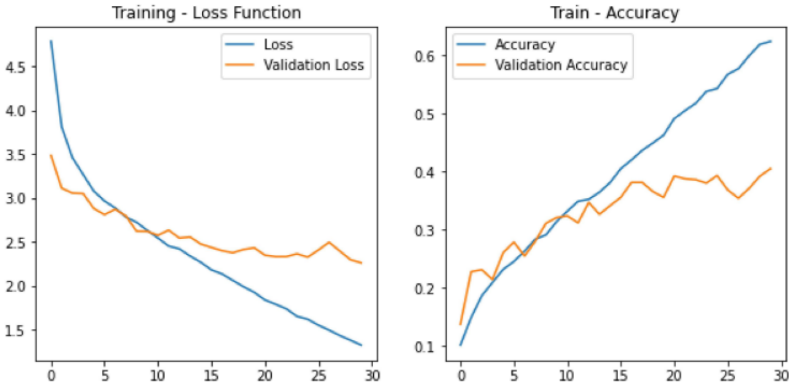
5.2.1 Implementation

For the implementation of this model the images size needed to be set. It was (227,227). After that a batch size of 32 was assigned to the model. 5913 out of 8446 images were used for training. 2553 images are for validation. A total of 30 epochs are runned for the model

5.2.2 Evaluation

The accuracy of the model gained was 62.39 percent and loss function was 1.32. the graph below shows the training curve of accuracy and loss function. As accuracy is increasing the loss is decreasing and accuracy is increasing but validation function is getting flat.

```
185/185 [==============================] - 278s 2s/step - loss: 1.3272 - accuracy: 0.6239 - val_loss: 2.2636 - val_accuracy: 0.4051
```



5.3 Model 3: ResNet-50 with imagent Weights

Resnet is a type of Resnet model with 50 layers out of which 48 are convolution layer, 1 Maxpooling layer and 1 Average Pooling layer. The image below shows Resnet architecture with different layers of which 50 layers are used. It also shows that Floating points operation are 3.8 X 10$\hat{9}$.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix}3\times3, 64\\3\times3, 64\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 64\\3\times3, 64\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix}3\times3, 128\\3\times3, 128\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 128\\3\times3, 128\end{bmatrix}\times4$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix}3\times3, 256\\3\times3, 256\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 256\\3\times3, 256\end{bmatrix}\times6$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times6$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times23$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix}3\times3, 512\\3\times3, 512\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 512\\3\times3, 512\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Figure 16: ResNet50 Architecture

5.3.1 Implementation

The batch size taken for the model is 64. And input shape is (224,224,3). For the weights of ResNet model imagenet weights are used. Output layer has softmax activation function. For the optimizer function adam is used with learning rate of 0.0001 and loss function of categorical_crossentropy. And a total of 30 epochs are runned.

5.3.2 Evaluation

The accuracy gained of training data was 99.59 percent and test data accuracy was 85.38 percent. Some random images are taken to the model are their painters are predicted. It is shown below:

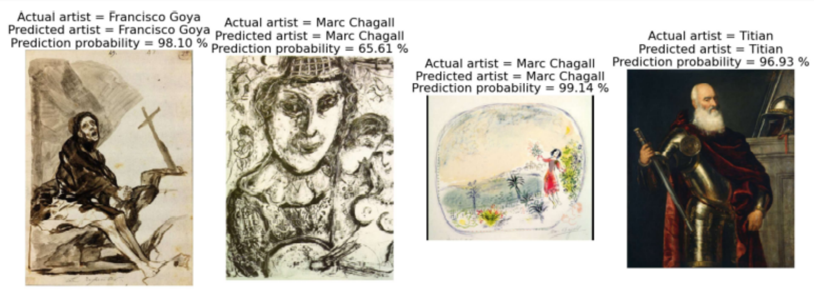
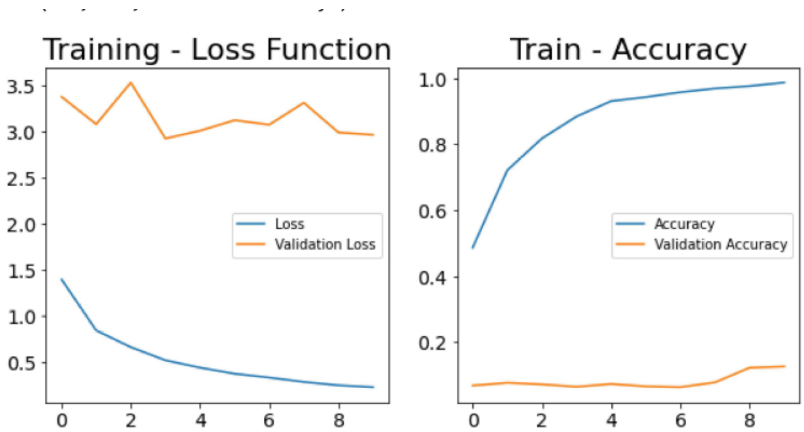


Figure 17: Training data Accuracy

```
14/14 [==============================] - 49s 3s/step - loss: 0.5456 - accuracy:
0.8538
Prediction accuracy on test data = 0.8538011908531189
```

Figure 18: Testing data Accuracy

```
54/54 [==============================] - 194s 4s/step - loss: 0.0803 - accurac
y: 0.9959
Prediction accuracy on train data = 0.9959349632263184
```



5.4 Model 4: ResNet-50 With Xception layer

5.4.1 Implementation

The Xception model is same as the model above but with an xceprion layer between the model. Below is the model summary of this model:

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_2 (InputLayer)        [(None, 180, 180, 3)]     0

 rescaling_1 (Rescaling)     (None, 180, 180, 3)       0

 random_zoom_1 (RandomZoom)  (None, 180, 180, 3)       0

 xception (Functional)       (None, 6, 6, 2048)        20861480

 global_average_pooling2d (G  (None, 2048)             0
 lobalAveragePooling2D)

 dropout_1 (Dropout)         (None, 2048)              0

 dense_2 (Dense)             (None, 51)                104499

=================================================================
Total params: 20,965,979
Trainable params: 104,499
Non-trainable params: 20,861,480
_________________________________________________________________
```

Figure 19: Model Summary

5.4.2 Evaluation

The Accuracy we got from this model is 83.21 percent.

```
185/185 [==============================] - 281s 2s/step - loss: 0.6094 - accu
racy: 0.8321 - val_loss: 1.8899 - val_accuracy: 0.5290
```

Figure 20: Model Summary

5.5 Model 5: Resnet-50 fine-tuned

5.5.1 Implementation

The images are first augmented randomly using a random transform function. The input figure size is (224,224,3) and the batch size is 16. The activation function used is Relu. And output activation function is Softmax. loss function of categorical_crossentropy. For the optimizer function adam is used with learning rate of 0.0001. A total of 50 epochs are runned.
The image below shows an example of image augmentation that is done by the model.

Figure 21: Augmented Images

5.5.2 Evaluation

The accuracy of the test data 80.11 percent. And the accuracy of train data was 93.20 percent. Moreover, confusion matrix was also presented, and a classification report is also shown down below. Also, the training graphs are also shown below:

```
54/54 [==============================] - 54s 992ms/step - loss: 0.8428 - accura
cy: 0.8012
Prediction accuracy on CV data = 0.8011695742607117
```

```
216/216 [==============================] - 214s 989ms/step - loss: 0.4906 - acc
uracy: 0.9321
Prediction accuracy on train data = 0.9320557713508606
```

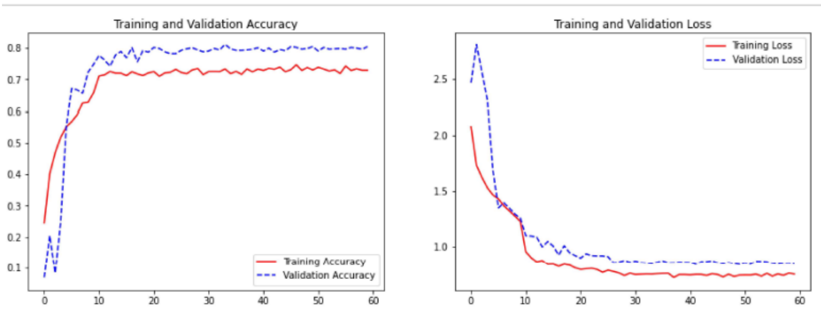


Figure 22: Training graphs

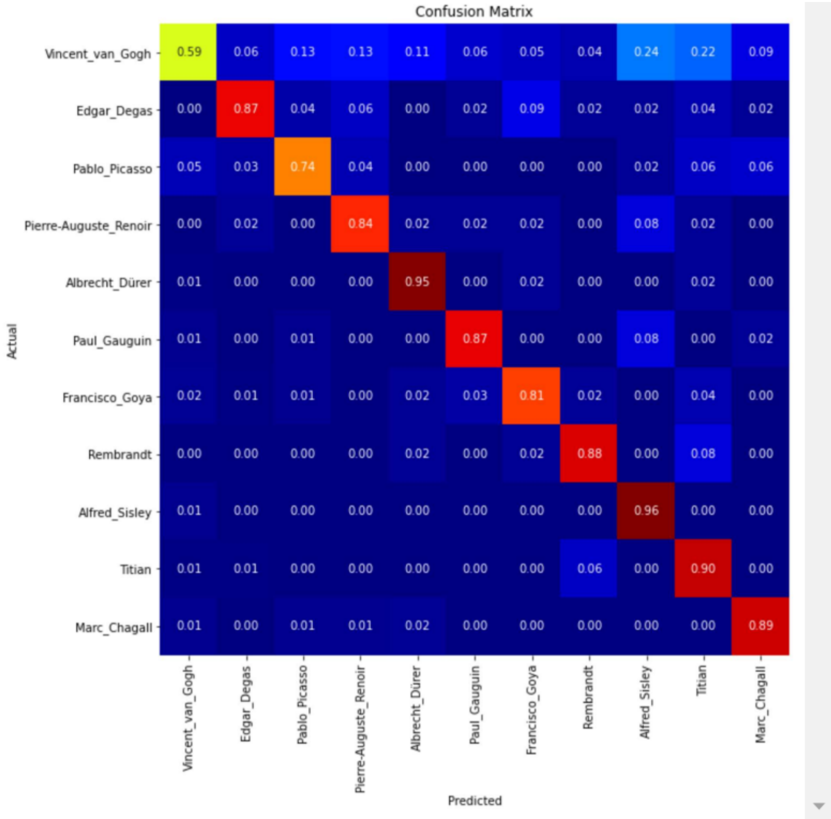


Figure 23: Confusion Matrix

Classification Report:

	precision	recall	f1-score	support
Vincent_van_Gogh	0.84	0.59	0.69	173
Edgar_Degas	0.88	0.87	0.87	139
Pablo_Picasso	0.79	0.74	0.76	85
Pierre-Auguste_Renoir	0.77	0.84	0.80	67
Albrecht_Dürer	0.85	0.95	0.90	65
Paul_Gauguin	0.87	0.87	0.87	62
Francisco_Goya	0.81	0.81	0.81	57
Rembrandt	0.87	0.88	0.87	51
Alfred_Sisley	0.69	0.96	0.80	51
Titian	0.66	0.90	0.76	51
Marc_Chagall	0.82	0.89	0.86	47
accuracy			0.81	848
macro avg	0.80	0.85	0.82	848
weighted avg	0.82	0.81	0.81	848

Figure 24: Classification Report

The images below also shows the actual prediction done by the model on test data as well as images taken from the internet.

Figure 25: Classification Report

```
Predicted artist = Titian
Prediction probability = 99.9320387840271 %
```

Figure 26: Classification Report

```
Predicted artist = Vincent van Gogh
Prediction probability = 37.23629713058472 %
```

Figure 27: Classification Report

```
Predicted artist = Pierre-Auguste Renoir
Prediction probability = 83.1238329410553 %
```

Figure 28: Classification Report

5.6 Implemented Model Comparison

The Table below shows the comparison between all the models implemented. From that the accuracy of Pretrained ResNet-50 model is the highest at 85.38 percent

Se. No.	Model	Optimizer	Activation Function	Accuracy
1	CNN	RMSprop	relu	38.98%
2	Alexnet	SGD	Softmax relu	62.39%
3	ResNet-50 with imagent Weights	adam	Softmax relu	85.38%
4	ResNet-50 With Xception layer	adam	relu	83.21%
5	Resnet-50 finetuned	Adam	relu	80.29%

Figure 29: Comparison of all the models implemented

5.7 Previously Implemented models comparison

The table below shows the comparison between all the models that are discussed in the paper:

Sl. No	Author	Method Implemented	Type of Classification	Number of Classes	Accuracy (in percentage)
1	Kelek et al. (2019)	GoogleNet, DenseNet, ResNet, Inceptionv3	Multi-class	17 classes, 46 images per class	78 percent
2	Narag and Soriano (2019)	SVM	Binary Classification	2 classes, 12 images per class	83 percent
3	Levy et al., (2014)	Genetic Algorithms and deep RBMs	Multi-class	3 classes, 40 images per class	90 percent
4	Cetinic and Grgic (2013)	MLP, SVM, Naïve Bayes, Random Forest	Multi-class	20 classes, 25 images per class	77 percent

Figure 30: Comparison of previoulsy implemented models

6 Conclusion and Future Work

The main goal of the project to use CNNs to identify painters based on their paintings was successfully completed by implementation of a resnet50 model with imagenet weights that gave an accuracy of 85.38 percent.
Furthermore, pretrained and non-pretrained models were compared and it was found that the pretrained models do a much better job at predicting paintings in this criterion. For evaluation of this model's accuracy was the main criteria. It was also seen that certain artists have clear styles and signatures and were better to predict than those artists who have worked in many different styles.
In this paper, the classification of paintings with their painters is a versatile task and many different techniques can be used to do this task. In this paper, CNNs were used but other techniques such as transfer learning and Template matching can also be used to do this task.
Currently, the data that was used had very clear images and the dataset has been properly labeled in future work not so complete dataset can be used. In addition to this users can input images to the program and directly get the artist should be an interesting project to work on.

References

Agarwal, S., Karnick, H., Pant, N. and Patel, U. (2015). Genre and style based painting classification, *2015 IEEE Winter Conference on Applications of Computer Vision*, IEEE, pp. 588–594.

Arora, R. S. and Elgammal, A. (2012). Towards automated classification of fine-art painting style: A comparative study, *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, IEEE, pp. 3541–3544.

Banerji, S. and Sinha, A. (2016). Painting classification using a pre-trained convolutional neural network, *International conference on computer vision, graphics, and image processing*, Springer, pp. 168–179.

Cetinic, E. and Grgic, S. (2013). Automated painter recognition based on image feature extraction, *Proceedings ELMAR-2013*, IEEE, pp. 19–22.

Chen, L., Chen, J., Zou, Q., Huang, K. and Li, Q. (2017). Multi-view feature combination for ancient paintings chronological classification, *Journal on Computing and Cultural Heritage (JOCCH)* **10**(2): 1–15.

Jangtjik, K. A., Ho, T.-T., Yeh, M.-C. and Hua, K.-L. (2017). A cnn-lstm framework for authorship classification of paintings, *2017 IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 2866–2870.

Johnson, C. R., Hendriks, E., Berezhnoy, I. J., Brevdo, E., Hughes, S. M., Daubechies, I., Li, J., Postma, E. and Wang, J. Z. (2008). Image processing for artist identification, *IEEE Signal Processing Magazine* **25**(4): 37–48.

Kelek, M. O., Calik, N. and Yildirim, T. (2019a). Painter classification over the novel art painting data set via the latest deep neural networks, *Procedia Computer Science* **154**: 369–376.

Kelek, M. O., Calik, N. and Yildirim, T. (2019b). Painter classification over the novel art painting data set via the latest deep neural networks, *Procedia Computer Science* **154**: 369–376.

Khan, A. I. and Al-Habsi, S. (2020). Machine learning in computer vision, *Procedia Computer Science* **167**: 1444–1451. International Conference on Computational Intelligence and Data Science.

Kim, J., Jun, J. Y., Hong, M., Shim, H. and Ahn, J. (2019). Classification of oil painting using machine learning with visualized depth information., *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences* .

Levy, E., David, O. E. and Netanyahu, N. S. (2014). Genetic algorithms and deep learning for automatic painter classification, *proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pp. 1143–1150.

Liu, C. and Jiang, H. (2014). Classification of traditional chinese paintings based on supervised learning methods, *2014 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, IEEE, pp. 641–644.

Nunez-Garcia, I., Lizarraga-Morales, R. A. and Hernandez-Gomez, G. (2018). Classification of paintings by artistic genre integrating color and texture descriptors, *Proceedings of the 2018 International Conference on Artificial Intelligence and Pattern Recognition*, pp. 66–70.

Pirrone, R., Cannella, V., Gambino, O., Pipitone, A. and Russo, G. (2009). Wikiart: An ontology-based information retrieval system for arts, *2009 Ninth International Conference on Intelligent Systems Design and Applications*, IEEE, pp. 913–918.

Saleh, B. and Elgammal, A. (2015). A unified framework for painting classification, pp. 1254–1261.

Seo, Y. and Shin, K.-s. (2018). Image classification for vehicle type dataset using state-of-the-art convolutional neural network architecture, *Proceedings of the 2018 Artificial Intelligence and Cloud Computing Conference*, pp. 139–144.

Sun, M., Zhang, D., Ren, J., Wang, Z. and Jin, J. S. (2015). Brushstroke based sparse hybrid convolutional neural networks for author classification of chinese ink-wash paintings, *2015 IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 626–630.

Zujovic, J., Gandy, L., Friedman, S., Pardo, B. and Pappas, T. N. (2009). Classifying paintings by artistic genre: An analysis of features & classifiers, *2009 IEEE International Workshop on Multimedia Signal Processing*, IEEE, pp. 1–5.