Final Report: Team 72

Word Count: 2479 (Excluding figure words, cover page, and references)
Word Count Penalty: 0%

Name	**Student #**
Gaurav Ranganath	1004877573
Sowrov Talukder	1004832470
Gabriel Paulos	1005231973
Saeed Khan	1004862561

Introduction

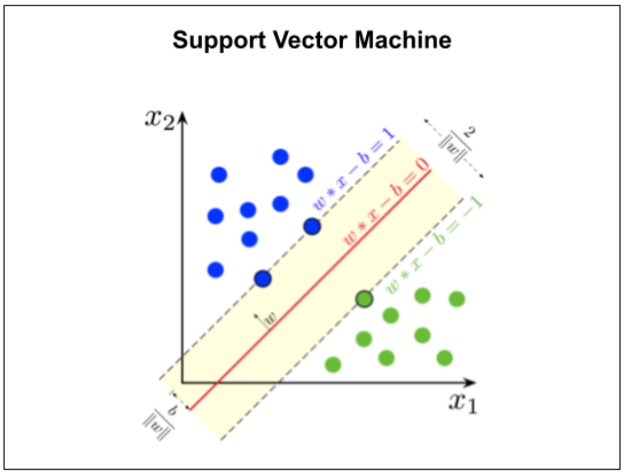
The intersection between art and technology can not only inspire new cultural and artistic movements, but it has the potential to advance machine learning as well, especially with regards to computer vision and deep learning[1]. The goal of the project is to create a machine learning model that is able to classify a painting in real time with an artistic style and to understand its relationship with that style. Machine learning is ideal to solve this problem as "[Deep Learning] excels at solving closed-end classification problems, in which a wide range of potential signals must be mapped onto a limited number of categories"[2]. As such, machine learning, specifically Deep Learning, would be the ideal method in developing a model that is able to classify paintings based on qualitative features.

Illustration / Figure

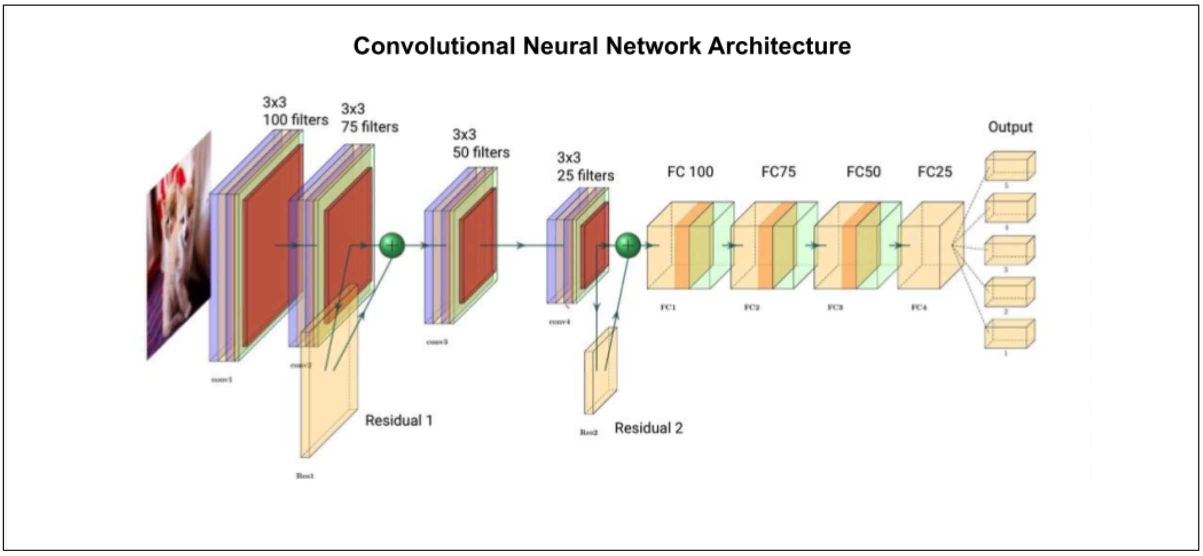
Project Workflow

1. *Research, collect, and clean dataset*

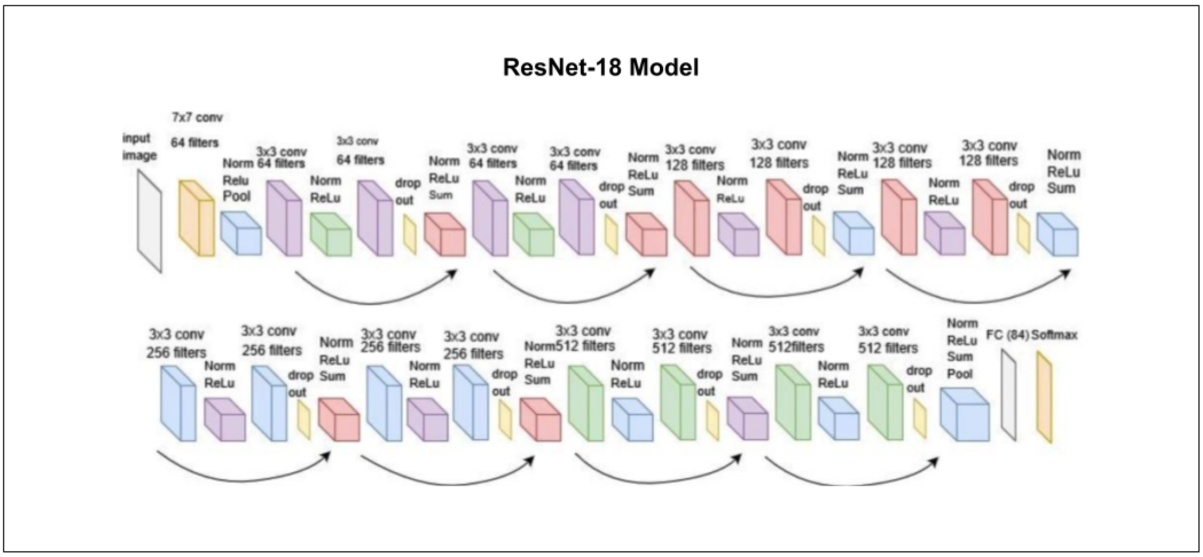
2. *Create a support vector machine (SVM) to evaluate data as a baseline model*



3. *Create our own convolutional neural network (CNN)*



4. *Use and improve upon transfer learning model (ResNet)[15]*



[15]

Background & Related Work

Many researchers have worked to tackle the problem of matching a painting to its art style. Dr. Florian Yger[1] tested multiple machine learning models against each other when trying to classify the art styles of painting. The research found that Residual Neural Networks were the most effective in classifying art styles and further analysis talked about the use of bagging to improve the accuracy of the model further.

Dr. Akshay Joshi and Dr. Ankit Agrawal's paper[7] took a different approach from Dr. Yger and developed a machine learning model based on Auto-Encoding Transformations to classify the art style of paintings. The researchers developed their own semi-supervised model and would utilize transformation of the paintings to classify the art styles. The researchers developed their own framework for the model as well, which included an encoder, classifier, and decoder. Each image was transformed with the following transformations: Projective, Affine, Similarity, Euclidean, and a combination of Color, Contrast, Brightness, Sharpness.

Data Processing

The principal source of our data that we will use will be the refined version of Chee Seng Chan's WikiArt Dataset, originally used for his ICIP2016 paper[8].

In addition to Chan's dataset, another resource that we will use is WikiArt itself[9], scraping images from our choses art styles for classification.

Two supplementary datasets that we will use will be Pedro Torres' Kaggle dataset for Cubism, Expressionism, and Romanticism paintings[10], and BryanB's Abstract Art Gallery dataset for Abstract paintings[11].

The data processing steps will be as follows:

1. Aggregate images from different datasets (as JPG files) into labelled folders which express their associated classifications (Abstract, Baroque, etc.)
2. Resize images to 128x128 pixels
3. Split the data into train (70%), validation (15%) and test (15%) sets
4. Compute mean of all pixel values of Training set, and then subtract the mean from all pixel values in all sets

Architecture

The architecture that our project used for the Convolutional Neural Network (CNN) was as follows:

- Input: 128x128 RGB images
- 4 Convolutional layers, each using 3x3 kernels with 100 channels, 75 channels, 50 channels, and 25 channels respectively
- 4 Max-Pooling layers between every convolutional layer using 2x2 kernels
- 4 Fully-Connected layers with Softmax output and with 100, 75, 50, and 25 hidden units respectively.
- 2 Residual layers after the 2nd and 4th convolutional layer.
- Output: Classification of art-style based on Softmax output for 3, 5, and 7 styles.
- Cross-Entropy loss function
- Adam optimizer

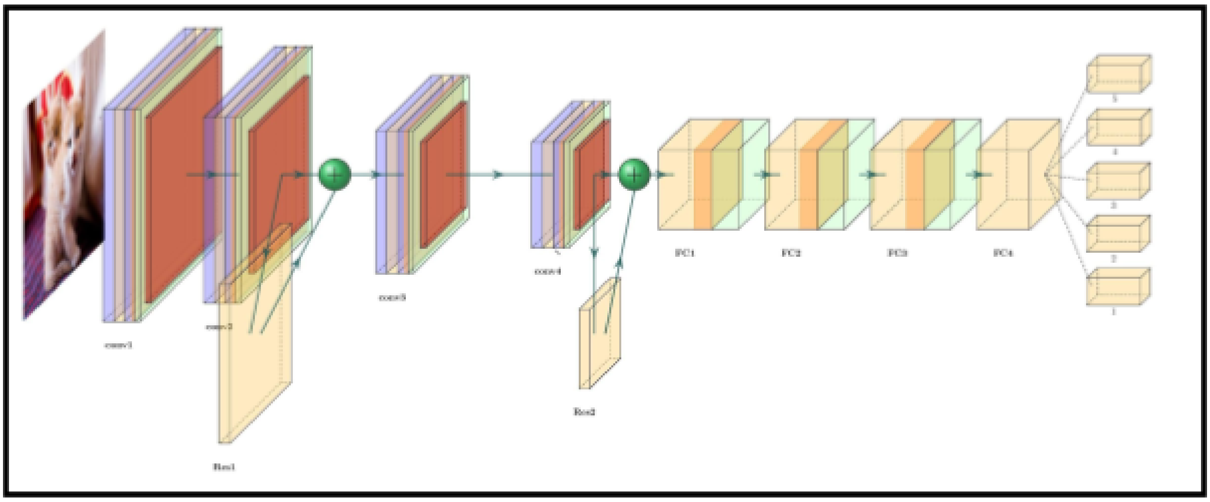


Figure 1: CNN model

We also developed a transfer learning model with ResNet18 with the following architecture:

- Input: 224x224 RGB images.
- First has a convolutional layer with 7x7 kernel with 64 channels followed by a 3x3 max pool
- 16 convolutional layers, with each using a 3x3 kernel with 64 channels for the first 4 layers, 128 for the next 4, 256 channels for the next 4, and 512 channels for the last 4 layers
- A 7x7 average pool after the last convolutional layer
- 3 fully connected layer of 1000, 200, 50 hidden layers respectively for each layer
- Output: Classification of art-style based on Softmax output for 3,5, and 7 styles.

Layer Name	Output Size	ResNet-18
conv1	$112 \times 112 \times 64$	7×7, 64, stride 2
conv2_x	$56 \times 56 \times 64$	3×3 max pool, stride 2 $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$
conv3_x	$28 \times 28 \times 128$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$
conv4_x	$14 \times 14 \times 256$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$
conv5_x	$7 \times 7 \times 512$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$
average pool	$1 \times 1 \times 512$	7×7 average pool
fully connected	1000	512×1000 fully connections
softmax	1000	

Figure 2: ResNet Model [16]

Baseline Model

A reasonable baseline to compare our model's performance to would be a Support Vector Machine (SVM) with a linear kernel. This baseline model would be implemented using the

Scikit-learn machine learning library and would operate on the same cleaned data we feed into our CNN model. We would compare the accuracy of the predictions made by the two models to get a sense of how our model is performing.

The LinearSVC class would be used for classification (it uses a linear kernel by default). The parameters for the LinearSVC would be kept as all default (e.g. L2 penalty, Squared-Hinge loss, Dual optimization, One-vs-Rest multi-class classification, etc.), as these are the most simple, straightforward, and likely performant choices for the baseline model in our case.

Quantitative Results

We created three output variants for each model we used: one for 3, 5, and 7 art styles. The SVM's testing accuracy was only approximately 25% for 3/5/7 art-styles. The tuned CNN model's accuracy results for all three variances are shown below:

Model	Training Accuracy	Validation Accuracy	Testing Accuracy
CNN-3 styles	74%	66%	69%
CNN-5 styles	55%	52%	51%
CNN-7 styles	48%	40%	38%

Figure 3: Table for accuracies of CNN variants (See appendix for all results)

The three style CNN variant was able to achieve the highest testing accuracy. Our CNN model is a great success when compared to the baseline model as all the variants' final testing accuracy surpasses its SVM counterpart. We decided to also compare our CNN against a ResNet-18 model that we slightly modified in order to see how the CNN model compares to a state of the art transfer learning model. The tuned ResNet model's final accuracy results for all three variants are shown below:

Model	Training Accuracy	Validation Accuracy	Testing Accuracy
ResNet-3 styles	95%	79%	77%
ResNet-5 styles	92%	69%	66%
ResNet-7 styles	89%	58%	55%

Figure 4: Table for the accuracies of ResNet variants (See appendix for all results)

The three style ResNet variant was able to achieve the highest testing accuracy of all models and their variants. All of our models overfit after 6 epochs, and their validation accuracies plateaus significantly for all variants of both models. This proves to be a problem as the model has issues generalizing the more similar art styles in our dataset.

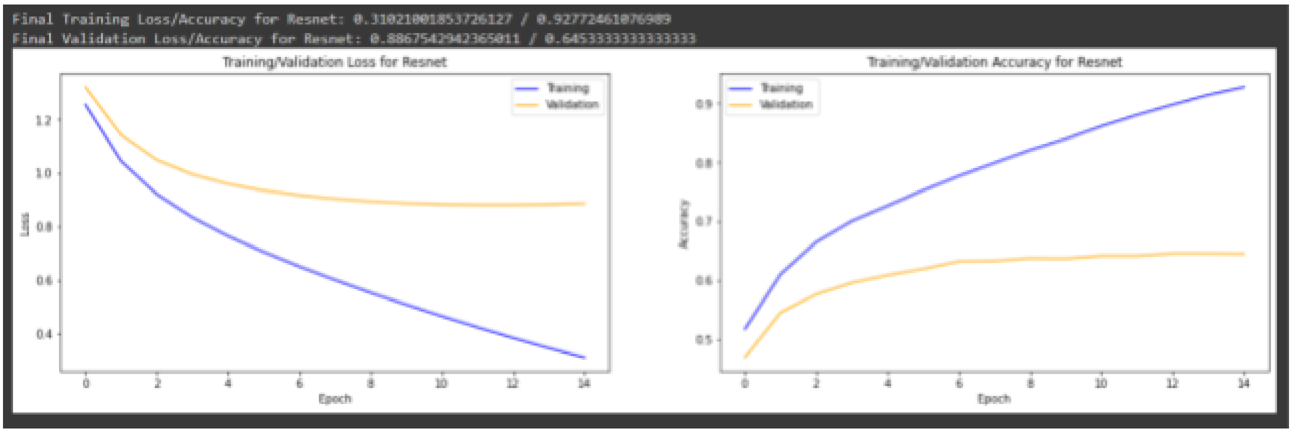


Figure 5: Resnet training and validation accuracy and loss for 5 style variant

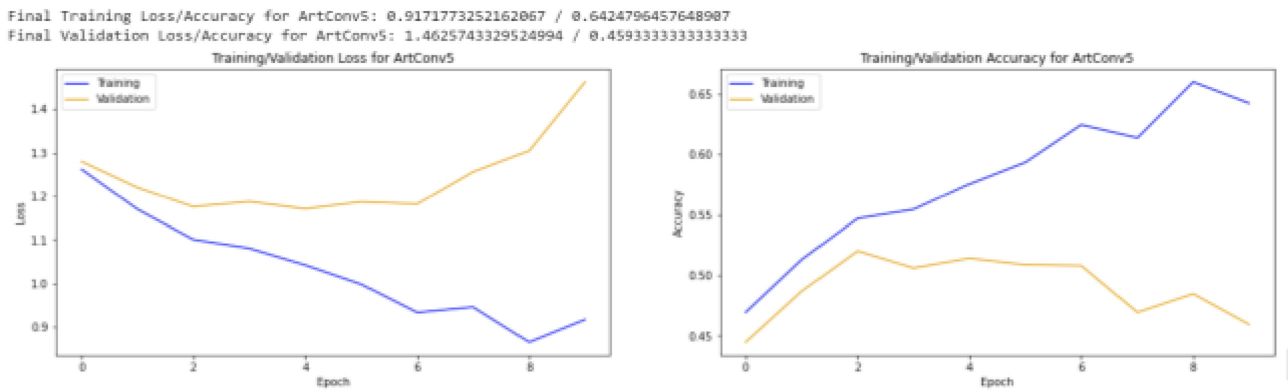


Figure 6: Training and Validation accuracy for our 5 style CNN model variant

Qualitative Results

When we look deeper into the results we see that both our CNN and ResNet models find it difficult to distinguish between some of the art styles that are visually similar or historically related. We can use the confusion matrices for the CNN and the ResNet model to extract how well the models classified the styles from the testing dataset. Cubism and Romanticism were the most easily distinguishable art styles for both models, as they have the most distinct features of all the styles we used in training and testing the models. Both models had trouble identifying Symbolism and Expressionism correctly, as shown by the recall in both models in the figures below. The models confuse Expressionism with Cubism as Expressionism shares similar elements, like color or line use, with each other.

However the depth of the ResNet model allowed it to distinguish Impressionism with a marked improvement from the CNN model, as the recall increased by 27%. Impressionism has many subtle elements that make it difficult to identify such as thin brush lines, which can be hard for the more shallow CNN model to identify. Below are the confusion matrices for both models that specify the recall and precision for the 5 style variant of both the CNN and the ResNet model. Despite the significant parameter disparity between both models, each model still had trouble misclassifying the more subtle art styles and had success in classifying with the more visibly distinguishable elements.

Confusion Matrix -- Testing:

	actual_Impressionism_2000	actual_Expressionism_2000	actual_Symbolism_2000	actual_Cubism_2000	actual_Romanticism_2000
predicted_Impressionism_2000	239.0	17.0	46.0	3.0	37.0
predicted_Expressionism_2000	26.0	161.0	56.0	46.0	18.0
predicted_Symbolism_2000	22.0	20.0	153.0	8.0	28.0
predicted_Cubism_2000	6.0	47.0	12.0	228.0	1.0
predicted_Romanticism_2000	28.0	33.0	60.0	0.0	205.0

Class Metric Scores -- Testing:

	Impressionism_2000	Expressionism_2000	Symbolism_2000	Cubism_2000	Romanticism_2000
Recall	0.744548	0.579137	0.467890	0.800000	0.709343
Precision	0.698830	0.524430	0.662338	0.775510	0.628834
F1-Score	0.720965	0.550427	0.548387	0.787565	0.666667

Figure 7: Confusion Matrix for the ResNet 5 style

Confusion Matrix -- Testing:

	a_Cubism_2000	a_Symbolism_2000	a_Expressionism_2000	a_Impressionism_2000	a_Romanticism_2000
p_Cubism_2000	194.0	9.0	62.0	15.0	5.0
p_Symbolism_2000	22.0	92.0	32.0	33.0	17.0
p_Expressionism_2000	101.0	40.0	109.0	34.0	18.0
p_Impressionism_2000	13.0	34.0	29.0	156.0	43.0
p_Romanticism_2000	6.0	90.0	47.0	92.0	207.0

Class Metric Scores -- Testing:

	Cubism_2000	Symbolism_2000	Expressionism_2000	Impressionism_2000	Romanticism_2000
Recall	0.577381	0.347170	0.390681	0.472727	0.713793
Precision	0.680702	0.469388	0.360927	0.567273	0.468326
F1-Score	0.624799	0.399132	0.375215	0.515702	0.565574

Macro F1-Score -- Testing: 0.5
Weighted F1-Score -- Testing: 0.5
Final Testing Accuracy for ArtConv5: 51.0%

Figure 8: Confusion Matrix for the CNN model

Evaluate Model on New Data

For evaluating the model on new data, we took several steps to make sure that the results were a good measure of the model's generalization capabilities.

The first important step was to make sure that our testing data was not impacted in any way by the data processing steps we took prior to training the model. We randomly shuffled and sampled 2000 images from our dataset for each art-style to make sure that the dataset was balanced (no art-style had more training data than another).

The next step in our data processing was resizing, and then computing the mean value of all the images in our training set and then subtracted it from the pixels in all of the sets; we maintained the integrity of our test (and validation) data by making sure that those values were not used in producing the mean pixel value, which would be data-snooping.

Our next step was to train our model on the training data, and evaluate its performance on the validation data. Here, we never referred to the test data at any point while tuning our hyperparameters. Heavy overfitting would occur during training, which would make our validation loss/accuracy plateau after a few epochs for both our CNN and ResNet-18 models. Since our accuracies (given the scale of our models and the amount of training data we had) were comparable to some estimates given by research into the same problem[1], we were satisfied with our performance after some hyperparameter tuning and model modifications.

Our final step was to evaluate our model on the test data, which until this point had not been seen by the model. To ensure proper generalization performance, we only used our test data once to verify good generalization from the validation performance. Our results after testing were as expected, with test accuracy within a few percent of our validation accuracy, showing that our model was generalizing as expected.

The last model evaluation we did for the project was for our presentation demonstration, where we evaluated our 5-style CNN and ResNet-18 models on 5 randomly selected images found online, each of a different art-style. We processed them the same way as testing data and let our model perform inference on them, and found the results to match our testing results quite well (40% for CNN, 80% for ResNet-18), meaning our model can be properly evaluated on new data.

Discussion

Support Vector Machine (SVM)

Our SVM performance was 27% which is not good compared to the 62% final testing accuracy achieved by the research papers we reviewed[1]. As a baseline the low accuracy confirms that the problem we are trying to solve is intrinsically difficult. The reason for our low SVM accuracy is because the model classifies points in an extremely high dimensional space and has no way to account for correlations between distinguishable features in each art style. Something interesting about our SVM results to note was that it had extremely high training accuracy due to heavy overfitting.

Convolutional Neural Network (CNN)

Our CNN showed noticeable improvement over the baseline with a final testing accuracy of 51% which is reasonably good considering the research papers we reviewed had a final testing accuracy of 62%[1]. Something surprising about our CNN results was that they were much closer to our ResNet model results than expected. We predicted that our CNN results would have been much worse.

ResNet-18

Our ResNet model gave us the best results and even came close to the 62% final testing accuracy from the research literature we reviewed[1]. However, it is important to note that the literature we referenced used many more classes than we did and because we lack the resources (hardware and time) to include additional classes we cannot compare our model on the same scale. Something interesting to note about our ResNet model was that it was particularly very good at identifying cubist style paintings and we believe this is because of their extremely distinct features.

Final Thoughts

Even though our model had weaknesses, we can see that different models actually learned features for different styles and did not learn random aspects that made no sense. For example, our model was able to predict cubist paintings much better than other styles, showing how our model with its limitations still learned the cubist style. If we were to improve our model even further, we would be able to learn the other features of other art styles.

Ethical Considerations

Classifying paintings into specific styles, especially those found in fine arts is not an objective task[12]. Art critics use qualitative features to identify a painting's style: color, line, depth, composition, light, space, shape, etc. As such, the ground truths in our model will be subjective[13]. The AI model will extrapolate this bias when trained, which will affect the grouping of a painting with an art style. In the case where that painting is closely related to the viewer's culture, the outputs of the model can affect their understanding of their culture, either positively or negatively[13][14]. Classifying art using a model could receive backlash from the fine arts community as some don't want their art defined by someone else's technology.

Project Difficulty / Quality

Project Difficulty

The difficulty of our project can be separated into two major categories, hardware limitations and data availability.

1. *Hardware Limitations*

   Due to the memory constraints of Google Colab we were not able to train larger models such as Resnet-50 or a 12-layer CNN, which may have increased our final accuracy. Time was also a large difficulty to work around when developing our model. We had to select our data carefully since the training time increased by a large amount when we increased our data. For example, for one training session we had to wait for 15 hours for the training to complete on ResNet.

2. *Availability of Data*

   Theoretically there could be an infinite amount of variations between different paintings of the same art style. For example, both the images below are part of the Symbolism art style, yet look very different.

Death and the Grave Digger by Carlos Schwabe: Symbolism Art Style[17]

The Three Brides by Jan Toorop: Symbolism Art Style[18]

Due to hardware limitations we had to manually select images that we believed best represented their respective art style. The difficulty in knowing which styles best represented their art style and then discarding other paintings which we thought least represented their art style made it difficult for us to create a model that was flexible in classifying paintings that were very different from one another.

Project Quality

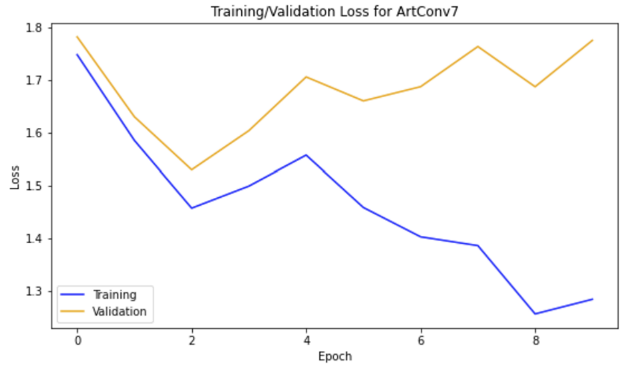
Even with many difficulties with the project, we were able to create solid models using our own custom CNN as well as transfer learning with ResNet-18. The research papers we reviewed achieved about a 62%[1] testing accuracy for 25 styles. Even though we couldn't create a model for 25 art styles due to the difficulties mentioned above, we did create multiple "scaled down" models which could predict for 3,5, and 7 styles. Both our CNN and ResNet-18 models had results which were in a similar range as researched models, which is a large success.

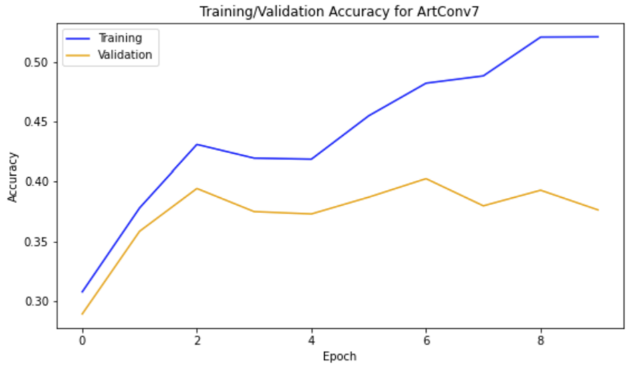
Appendix

A. Colab Links:
   - https://colab.research.google.com/drive/1q1D7tNVwPcXQX6NGGC_g4zD2IloxzCc9?usp=sharing
   - https://colab.research.google.com/drive/1mO1BXTw14-Ubk-I91K0aybVyCvpwzJto?usp=sharing
   - https://colab.research.google.com/drive/1xLrcuIhch6fEuEsojE-1ZlvhahD8omr7?usp=sharing

B. Collected Data:

```
Final Training Loss/Accuracy for ArtConv7: 1.284558510148762 / 0.5211018711018711
Final Validation Loss/Accuracy for ArtConv7: 1.7751600236603708 / 0.3763336566440349
```





Confusion Matrix -- Validation:

	a_Impressionism_2000	a_Expressionism_2000	a_Cubism_2000	a_Romanticism_2000	a_Realism_2000	a_Symbolism_2000	a_Baroque_2000
p_Impressionism_2000	189.0	77.0	55.0	32.0	72.0	66.0	19.0
p_Expressionism_2000	6.0	54.0	71.0	15.0	7.0	14.0	16.0
p_Cubism_2000	4.0	27.0	119.0	2.0	8.0	7.0	0.0
p_Romanticism_2000	29.0	28.0	10.0	134.0	68.0	56.0	59.0
p_Realism_2000	35.0	20.0	17.0	31.0	70.0	22.0	19.0
p_Symbolism_2000	30.0	58.0	32.0	28.0	24.0	134.0	13.0
p_Baroque_2000	9.0	11.0	9.0	68.0	42.0	16.0	130.0

Class Metric Scores -- Validation:

	Impressionism_2000	Expressionism_2000	Cubism_2000	Romanticism_2000	Realism_2000	Symbolism_2000	Baroque_2000
Recall	0.625828	0.196364	0.380192	0.432258	0.240550	0.425397	0.507812
Precision	0.370588	0.295082	0.712575	0.348958	0.327103	0.420063	0.456140
F1-Score	0.465517	0.235808	0.495833	0.386167	0.277228	0.422713	0.480591

```
Macro F1-Score -- Validation: 0.39
Weighted F1-Score -- Validation: 0.4
Final Validation Accuracy for ArtConv7: 40.0%
```

Confusion Matrix -- Testing:

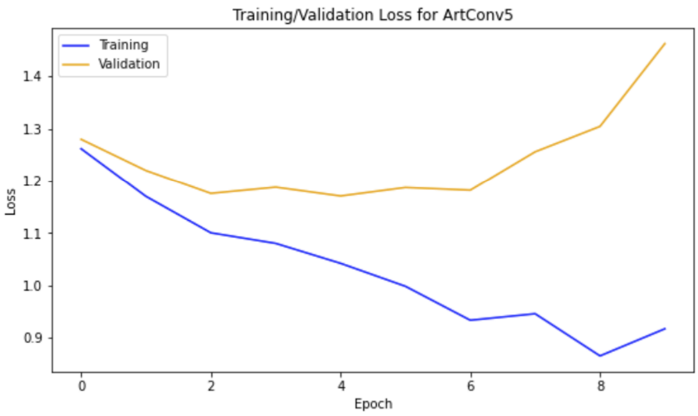
	a_Impressionism_2000	a_Expressionism_2000	a_Cubism_2000	a_Romanticism_2000	a_Realism_2000	a_Symbolism_2000	a_Baroque_2000
p_Impressionism_2000	166.0	79.0	58.0	46.0	69.0	72.0	21.0
p_Expressionism_2000	8.0	60.0	63.0	9.0	11.0	15.0	6.0
p_Cubism_2000	2.0	37.0	109.0	3.0	7.0	10.0	0.0
p_Romanticism_2000	37.0	24.0	11.0	124.0	60.0	55.0	52.0
p_Realism_2000	50.0	30.0	18.0	35.0	64.0	22.0	17.0
p_Symbolism_2000	43.0	58.0	53.0	25.0	20.0	111.0	11.0
p_Baroque_2000	7.0	15.0	6.0	42.0	51.0	19.0	151.0

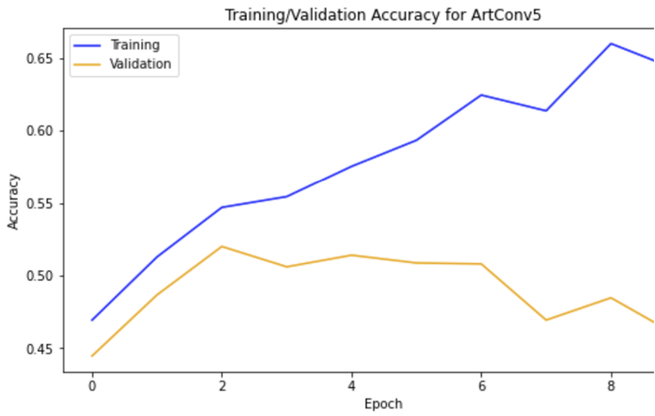
Class Metric Scores -- Testing:

	Impressionism_2000	Expressionism_2000	Cubism_2000	Romanticism_2000	Realism_2000	Symbolism_2000	Baroque_2000
Recall	0.530351	0.198020	0.342767	0.436620	0.226950	0.365132	0.585271
Precision	0.324853	0.348837	0.648810	0.341598	0.271186	0.345794	0.518900
F1-Score	0.402913	0.252632	0.448560	0.383308	0.247104	0.355200	0.550091

```
Macro F1-Score -- Testing: 0.38
Weighted F1-Score -- Testing: 0.38
Final Testing Accuracy for ArtConv7: 38.0%

Final Training Loss/Accuracy for ArtConv5: 0.9171773252162067 / 0.6424796457648907
Final Validation Loss/Accuracy for ArtConv5: 1.4625743329524994 / 0.4593333333333333
```





```
Confusion Matrix -- Training:
```

	a_Cubism_2000	a_Symbolism_2000	a_Expressionism_2000	a_Impressionism_2000	a_Romanticism_2000
p_Cubism_2000	851.0	124.0	300.0	27.0	16.0
p_Symbolism_2000	95.0	545.0	156.0	137.0	69.0
p_Expressionism_2000	329.0	169.0	594.0	122.0	80.0
p_Impressionism_2000	39.0	208.0	141.0	744.0	144.0
p_Romanticism_2000	51.0	399.0	212.0	355.0	1094.0

```
Class Metric Scores -- Training:
```

	Cubism_2000	Symbolism_2000	Expressionism_2000	Impressionism_2000	Romanticism_2000
Recall	0.623443	0.377163	0.423378	0.537184	0.779758
Precision	0.645675	0.543912	0.459042	0.583072	0.518238
F1-Score	0.634365	0.445443	0.440489	0.559188	0.622652

```
Macro F1-Score -- Training: 0.54
Weighted F1-Score -- Training: 0.54
Final Training Accuracy for ArtConv5: 55.00000000000001%
```

```
Confusion Matrix -- Validation:
```

	a_Cubism_2000	a_Symbolism_2000	a_Expressionism_2000	a_Impressionism_2000	a_Romanticism_2000
p_Cubism_2000	178.0	22.0	77.0	9.0	3.0
p_Symbolism_2000	19.0	100.0	34.0	33.0	14.0
p_Expressionism_2000	78.0	34.0	136.0	31.0	32.0
p_Impressionism_2000	9.0	39.0	29.0	137.0	32.0
p_Romanticism_2000	15.0	95.0	41.0	74.0	229.0

```
Class Metric Scores -- Validation:
```

	Cubism_2000	Symbolism_2000	Expressionism_2000	Impressionism_2000	Romanticism_2000
Recall	0.595318	0.344828	0.429022	0.482394	0.738710
Precision	0.615917	0.500000	0.437299	0.556911	0.504405
F1-Score	0.605442	0.408163	0.433121	0.516981	0.599476

```
Macro F1-Score -- Validation: 0.51
Weighted F1-Score -- Validation: 0.51
Final Validation Accuracy for ArtConv5: 52.0%
```

```
Confusion Matrix -- Testing:
```

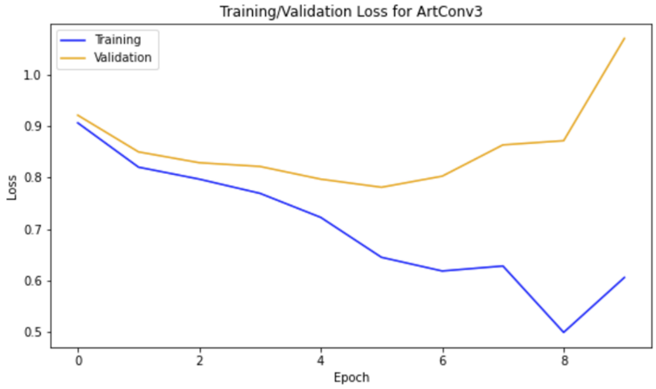
	a_Cubism_2000	a_Symbolism_2000	a_Expressionism_2000	a_Impressionism_2000	a_Romanticism_2000
p_Cubism_2000	194.0	9.0	62.0	15.0	5.0
p_Symbolism_2000	22.0	92.0	32.0	33.0	17.0
p_Expressionism_2000	101.0	40.0	109.0	34.0	18.0
p_Impressionism_2000	13.0	34.0	29.0	156.0	43.0
p_Romanticism_2000	6.0	90.0	47.0	92.0	207.0

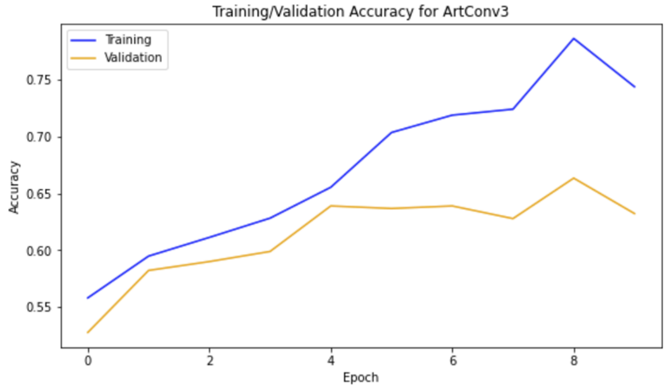
```
Class Metric Scores -- Testing:
```

	Cubism_2000	Symbolism_2000	Expressionism_2000	Impressionism_2000	Romanticism_2000
Recall	0.577381	0.347170	0.390681	0.472727	0.713793
Precision	0.680702	0.469388	0.360927	0.567273	0.468326
F1-Score	0.624799	0.399132	0.375215	0.515702	0.565574

```
Macro F1-Score -- Testing: 0.5
Weighted F1-Score -- Testing: 0.5
Final Testing Accuracy for ArtConv5: 51.0%
```

```
Final Training Loss/Accuracy for ArtConv3: 0.6060431518337943 / 0.7436874702239161
Final Validation Loss/Accuracy for ArtConv3: 1.0698586543401083 / 0.6322222222222222
```





Confusion Matrix -- Training:

	a_Expressionism_2000	a_Cubism_2000	a_Impressionism_2000
p_Expressionism_2000	791.0	121.0	93.0
p_Cubism_2000	388.0	1222.0	14.0
p_Impressionism_2000	232.0	50.0	1287.0

Class Metric Scores -- Training:

	Expressionism_2000	Cubism_2000	Impressionism_2000
Recall	0.560595	0.877243	0.923242
Precision	0.787065	0.752463	0.820268
F1-Score	0.654801	0.810076	0.868714

```
Macro F1-Score -- Training: 0.78
Weighted F1-Score -- Training: 0.78
Final Training Accuracy for ArtConv3: 79.0%
```

Confusion Matrix -- Validation:

	a_Expressionism_2000	a_Cubism_2000	a_Impressionism_2000
p_Expressionism_2000	113.0	41.0	38.0
p_Cubism_2000	114.0	237.0	24.0
p_Impressionism_2000	60.0	26.0	247.0

Class Metric Scores -- Validation:

	Expressionism_2000	Cubism_2000	Impressionism_2000
Recall	0.393728	0.779605	0.799353
Precision	0.588542	0.632000	0.741742
F1-Score	0.471816	0.698085	0.769470

Macro F1-Score -- Validation: 0.65
Weighted F1-Score -- Validation: 0.65
Final Validation Accuracy for ArtConv3: 66.0%

Confusion Matrix -- Testing:

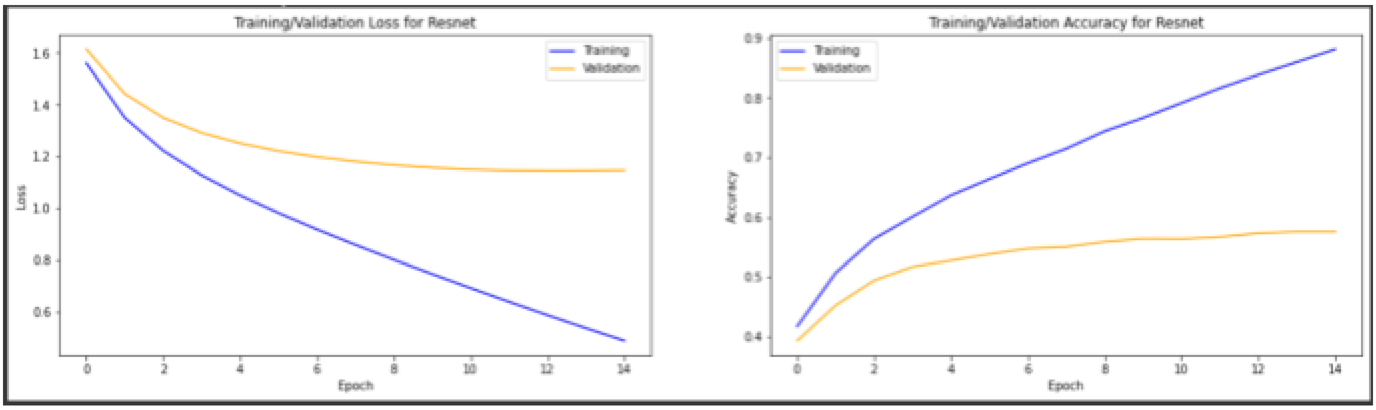
	a_Expressionism_2000	a_Cubism_2000	a_Impressionism_2000
p_Expressionism_2000	130.0	45.0	33.0
p_Cubism_2000	109.0	248.0	18.0
p_Impressionism_2000	62.0	10.0	245.0

Class Metric Scores -- Testing:

	Expressionism_2000	Cubism_2000	Impressionism_2000
Recall	0.431894	0.818482	0.827703
Precision	0.625000	0.661333	0.772871
F1-Score	0.510806	0.731563	0.799347

Macro F1-Score -- Testing: 0.68
Weighted F1-Score -- Testing: 0.68
Final Testing Accuracy for ArtConv3: 69.0%

ResNet 3 style variant



References

[1] Lecoutre, A., Negrevergne, B., Yger, F., “Recognizing Art Style Automatically with deep learning,” in *Proceedings of Machine Learning Research: Asian Conference on Machine Learning*, Seoul, Korea, 15-17 November 2017 N. Lawrence, M. Reid, Eds. pp. 327–342.

[2] N. O'Mahony, S. Campbell, A. Carvalho, S. Harapanahalli, G. V. Hernandez, L. Krpalkova, D. Riordan, and J. Walsh, "Deep Learning vs. Traditional Computer Vision," *Advances in Intelligent Systems and Computing Advances in Computer Vision*, pp. 128–144, 2019.

[3] Marino, M., "*Boats on the Seine*". 2021. [image]. Available: https://s31531.pcdn.co/wp-content/uploads/2018/06/MMarino_Boats-on-the-Seine-1024x674.jpg

[4] Raphael, "*School of Athens*". [Image]. Available: https://www.artyfactory.com/art_appreciation/art_movements/italian-renaissance/italian-renaissance/school-of-athens.jpg

[5] "*Abstract Art*". [Image]. Available: https://digitaldefynd.com/wp-content/uploads/2020/07/Best-Abstract-Art-course-tutorial-class-certification-training-online-scaled.jpg

[6] Rubens, P. "Garden of Love". [Image]. Available: https://dam-13749.kxcdn.com/wp-content/uploads/2017/09/ogrod_z5.jpg

[7] Joshi, A. et al. "Art Style Classification with Self-Trained Ensemble of AutoEncoding Transformations." *ArXiv abs/2012.0337*, 2020.

[8] Chan, C. "*WikiArt Dataset*": Github Repository, 2021. [Dataset]. Available: https://github.com/cs-chan/ArtGAN/tree/master/WikiArt%20Dataset

[9] "*Wikiart*". Wikiart Repository, 2021. [Dataset]. Available: https://www.wikiart.org/

[10] Torres,P. "*art movements*": Kaggle Repository, 2019. [Dataset]. Available: https://www.kaggle.com/torres07/art-movements

[11] Bryan, B. "*Abstract Art Gallery*": Kaggle Repository, 2021. [Dataset]. Available: https://www.kaggle.com/bryanb/abstract-art-gallery

[12] Lombardi, T, Pace University. "*The Classification of style in Fine-Art Painting*,". Accessed on: February 12, 2021. [Online]. Available: https://csis.pace.edu/~lombardi/docs/classification_of_style.pdf

[13] "*But Is It really Art? The Classification of Images as "Art"/"Not Art" and Correlation with Appraisal and Viewer Interpersonal Differences,*". Accessed on: February 12, 2021. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fpsyg.2017.01729/full

[14] TED Talks, "Thelma Golden How art gives shape to cultural change". 2009. Accessed on: February 12,2021. [Online]. Available:https://www.ted.com/talks/thelma_golden_how_art_gives_shape_to_cultural_change/transcript?language=en#t-555436

[15] M. A. Hasan, ResearchGate. [Online]. Available: https://www.researchgate.net/figure/Proposed-Modified-ResNet-18-architecture-for-Bangla-HCR-In-the-diagram-conv-stands-for_fig1_323063171. [Accessed: 04-Apr-2021].

[16] P. Napoletano, F. Piccoli, R. Schettini, K. Xu, T. Sun, and X. Jiang, “Table 1 . ResNet-18 Architecture.,” *ResearchGate*, 25-Sep-2020. [Online]. Available: https://www.researchgate.net/figure/ResNet-18-Architecture_tbl1_322476121. [Accessed: 10-Apr-2021].

[17] “Symbolism (arts),” *Wikipedia*, 05-Apr-2021. [Online]. Available: https://en.wikipedia.org/wiki/Symbolism_(arts). [Accessed: 10-Apr-2021].

[18] “Symbolism - Important Paintings,” *The Art Story*. [Online]. Available: https://www.theartstory.org/movement/symbolism/artworks/#pnt_4. [Accessed: 10-Apr-2021].