


Improving Art Style Classification through CNN Retraining and the Introduction of Non-Eurocentric Data

Sami Amer


samiamer@mit.edu

 sami-amer

 Sami Amer

Camila Miranda-Llovera

camilaml@mit.edu

 Camila Miranda-Llovera

Anita Podrug

apodrug@mit.edu

 Anita Podrug

Abstract

The amount of paintings in the world vastly surpasses the number of people who can properly classify them. Deep learning models have the potential to bridge the gap, but accuracy remains a concern, especially with little annotated data to train on. Through expansion of the popular WikiPaintings [11] dataset using more diverse painting styles, we shallowly re-trained three pre-trained image classification networks. These models (ResNet50 [5] trained on ImageNet [3], ResNet50 trained on Stylized ImageNet [4], and RASTA [7]) performed well, exhibiting results that aligned with our initial predictions. Despite the increases in overall performance, some per-class accuracies suffered, especially in harder to decipher classes (e.g. High Renaissance vs Late Renaissance). Results indicate that while the performance boosts were substantial, further exploration using more synthetically varied data could lead to even better results.

1. Introduction

In the world of art, there are hundreds, maybe thousands of different styles of paintings, each of which boasts tens of thousands of examples. With such a vast number of styles and an even larger quantity of paintings, it becomes intractable for museums and art galleries to correctly classify and organize their inventory – especially when this classification depends on a niche group with art history experience. The average person would be hard pressed to tell a baroque from a rococo, so the few who are equipped to make such decisions are left to tackle hundreds of thousands of paintings on their own. To alleviate this, there have been efforts to create neural networks that could bear some of the burden, automatically categorizing paintings with unambiguous classification, leaving the historians more time to debate the nuance in specific pieces.

Work on automatic art classification has seen a steady, if

minute, increase over time, with newer deep learning models breaking 50% accuracy. The most robust of these was the RASTA model [7], which achieved an accuracy of 62%. This model, however, was trained and tested only on art styles that had enough data in the WikiPaintings dataset. These proved to be mainly Euro-centric styles; genres from smaller countries or more obscure, foreign styles, have less than 200 paintings compared to the tens of thousands available for the likes of Renaissance art (Which itself is present as several sub-categories). Other papers on the identification of people in artwork faced a similar challenge, wherein the authors noted the level of difficulty in finding a more representative data set [6]. In the previous paper, the authors recognized that a large source of bias originated from a lack of racial representation, especially if the data set skewed toward Europe and the Americas – a finding that was evident from the neural network’s poor ability to locate people in Japanese art.

Our project set out to create a more robust art style classification model, focusing on introducing a more varied dataset to mitigate biases. Our main goal was to expand on the work of previous models and to include new, more representative data. We hypothesized that the inclusion of underrepresented styles will not only increase the styles the model can recognize, as is expected, but also enhance the model’s ability to recognize art from the original dataset.

1.1. Why Domain Transfer?

No matter how well a model transfers between different domains, one can always achieve a greater accuracy by training a CNN from the ground-up for a specific task. Thus, it naturally comes to mind: why not make an art style classification CNN? In addition to the obvious resource and time constraints that are inherent to a one-semester class project, a model created from the ground up for art style classification is only useful if there is an abundance of data, such that a golden standard can be created for comparable testing. As it currently stands, the largest publicly available annotated painting dataset is WikiPaintings, which stands at

about 80,000 images. Removing classes that are too small for any meaningful learning, and the number gets closer to 70,000, or an average of 2,700 per class (25 classes; also important to note that this average is not reflective of the actual dataset, which is imbalanced). Thus, while training a model from scratch will yield to better results, it is hard to justify the increased time and resource cost when any addition to the small dataset would drastically shift results. Instead, developing a robust retraining protocol allows us to take advantage of advances in the field of general image classification, and maintains the flexibility needed to incorporate new classes as datasets are released.

2. Methods

2.1. Image Dataset

To keep the work manageable yet impactful, we focused on extending the publicly available WikiPaintings [11] dataset. We achieved this by using The Metropolitan Museum of Art’s Open Access API [8], creating our own “fake” categories out of commonalities between paintings. We decided to synthesize our own categories for two reasons: first, it allowed us to add un-classified data to our dataset, which is crucial when our goal is to add less-represented styles. Second, the actual name of the category does not matter, and can be changed at any time, as long as the images within the category share some defining features. To this end, we were able to create 3 new datasets from the Met Open Access: Islamic Art, Islamic Textiles, and Ukiyo-e. The artworks that were categorized by the Islamic Art department of the Met are varied enough that they could be split into two different categories within the dataset. Islamic Art and Islamic Textiles are brand new, while Ukiyo-e is actually already present in the WikiPaintings dataset, but to a much lesser extent (the data we found doubled the original 1,000 images, bolstering the smallest class in the dataset). We also found a set of about 2000 paintings that depict Chinese Landscapes [13] in a consistent style, which we also included.

Overall, we extended the dataset by about 8000 images, an increase of about 10%. While on the lower side, the smallest of our added data was still in the ballpark of the smallest of the original data.

As a side note: the training, validation, and testing splits were not modified from the original RASTA repository, which downloads a .tar file with the data already split. This ensures that our retraining and retesting of the model was not skewed by the model testing on data it had already trained on.

2.2. Image Transformation

For model training, all images were resized to 224 by 224. To increase robustness in models that were not pre-

viously trained on art style classification, random flipping, resizing, and cropping was introduced (note that the resizing still resulted in a 224 by 224 image).

2.3. Models

To avoid reinventing the wheel, we chose 3 pre-trained models. The first of these is the RASTA model [7], which is a ResNet50 [5] model originally trained on ImageNet, and then deeply retrained (20 layers) for art style classification. We chose this because it was the best model that could do art style classification “out of the box”, and would allow us to most accurately define the effects of our new data. Our second model was also a ResNet50 model trained on ImageNet [3] (henceforth referred to as the ImageNet model), but without any deep retraining. Our last model is a ResNet50 trained on Stylized ImageNet (henceforth referred to as the StyleNet Model), the Bethge Lab’s solution to texture-dependency in classic ImageNet models [4]. We chose this last model to better help us understand how texture and shape influence art style classification.

2.3.1 Training

The ImageNet and StyleNet models were trained on both the original and the extended datasets, while the RASTA model was only trained on the extended dataset (retraining on the original dataset would have been redundant).

2.3.2 Pytorch Models (ImageNet and StyleNet)

For the Pytorch [9] models, images in the training set had a random resized crop, a random horizontal flip, and were then normalized. Images in the validation and testing were simply center cropped and resized. All resulting images were 224 by 224 by 3. A batch size of 32 was used, and the training/validation/testing splits were maintained from the original RASTA paper; all additional data was distributed between the three splits in proportions mirroring the original distribution (about 80:10:10).

Both models had the final layer (which output ImageNet predictions) replaced with a fully connected layer that output to a number nodes equal to the number of classes, with a softmax activation. All weights outside of this final fully connected layer were frozen, leading to about 51,000 trainable parameters for the original dataset and about 57,000 for the extended dataset. The models were trained for 25 epochs, using a Stochastic Gradient Descent optimizer with learning rate of 0.001, momentum of 0.9. The learning rate was decayed by a factor of 0.1 after every 7 epochs.

As mentioned before, the ImageNet model was a pre-trained Wide ResNet 50 loaded from Pytorch’s list of pre-trained models. The StyleNet model was loaded using the model loader included in the Bethge Lab’s github repo.

2.3.3 Tensorflow Model (RASTA)

For the Tensorflow [1] models, images in the training, validation, and testing sets were resized, while the training set had an added random horizontal flip. Images also had to be converted from RGB to BGR to be compatible with the RASTA model. This was done using Keras's [2] built in utilities. Similar to the Pytorch models, a fully connected layer was added in place of the final layer, again with a softmax. Weights were again frozen in such a way that the model had the same number of trainable parameters as above (note: no modifications were done to the original RASTA model, which was evaluated as is).

The models again used a batch size of 32 across 25 epochs, this time with an RMSprop optimizer to accurately replicate the RASTA paper.

2.4. Model Testing

Outside of the training and validation accuracies reported during model fitting, we calculated Top-k accuracy and per-class accuracy (the former script was modified from the community, the latter from the official Pytorch documentation). The script relied purely on tensor mathematics, which made it possible to port over to Tensorflow with some changes.

2.5. Human Testing

Our experiment was designed using PsychoPy [10] and consisted of a match-to-sample task for 3 images per art style category, totalling to 84 stimulus images. Each experiment ran through one iteration of all the stimuli, where each image was chosen at random and flashed for 400 ms, with experiments lasting around 4 minutes. After the stimulus image was flashed, two sample images were shown on the screen: one random image in the same art style of the sample image, and one random image of a different art style. The participant was asked to choose the image that seems most similar in art style to the stimulus image. All three images were recorded, as well as the participant's answer of either the left or right key. An example of the match-to-sample task is shown in Figure 1 and Figure 2. 20 participants from the MIT community were recorded as they completed the experiment alone with no access to other resources.

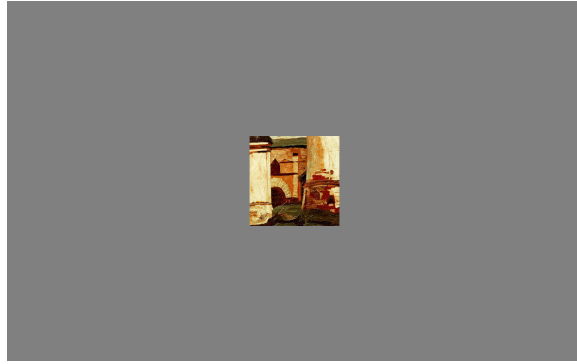


Figure 1. Example of stimulus image, flashed for 400 ms

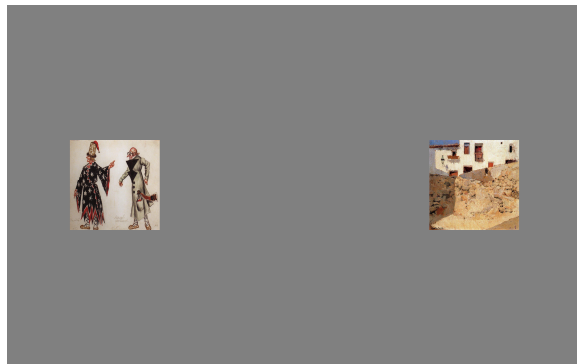


Figure 2. Example of sample images. A participant would now choose between the left and right image

3. Results

3.1. Model Results

3.1.1 ImageNet vs StyleNet

We were interested in seeing how the ImageNet model stacked up against the less texture-reliant StyleNet. The Top-k accuracies were surprisingly close, with the ImageNet model boasting 44.1%, 72.5%, and 84.1% Top-1, Top-3 and Top-5 on the original dataset, respectively. In these same circumstances, the StyleNet model had 43.9%, 72.2%, and 84.7%. While these differences are negligible, the data gets more interesting when calculated per class, as seen in Figure 3. The black line denotes the RASTA model's overall top-1 accuracy.

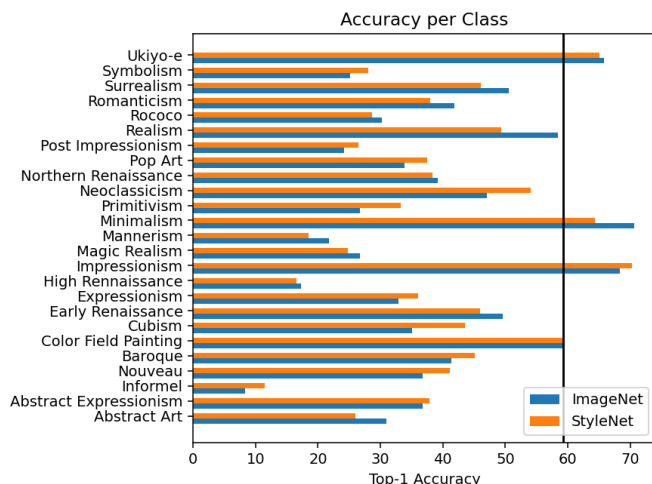


Figure 3. Bar Graph Comparing StyleNet and ImageNet model accuracies per class. Black line represents RASTA top-1 accuracy on the same dataset

Looking at this graph, we notice that the accuracies vary per-class, going as low as 18.5% on Mannerism and as high as 70.1% on Impressionism. We can better view the direct differences between the two models in Figure 4

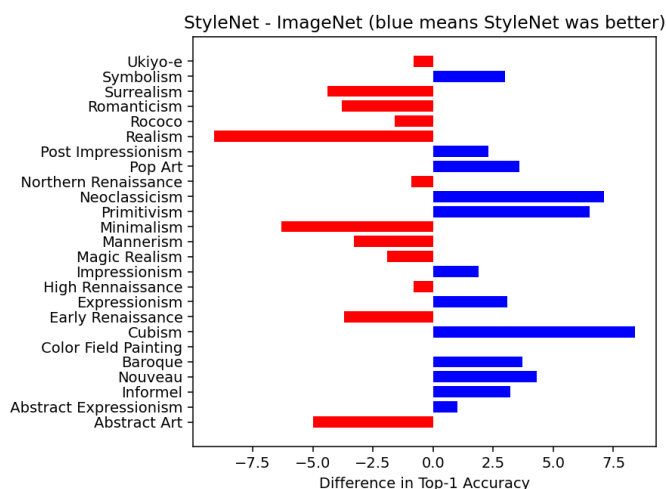


Figure 4. Difference in top-1 accuracy between the StyleNet and ImageNet models (blue means StyleNet is better)

The results are as expected, with ImageNet performing better on more texture based styles like Impressionism and Realism, and StyleNet performing better on more shape based styles like Cubism and Primitivism. Interestingly, Minimalism, which is defined by minimal colors and shapes, is classified much more accurately by the ImageNet

model.

Using the extended dataset leads to some interesting changes, with some accuracy differences increasing in magnitude and others flipping. Looking at Figure 5, we see the ImageNet model widen its gap in Northern Renaissance and Minimalism the most, with 8 and 7 percent gaps, respectively.

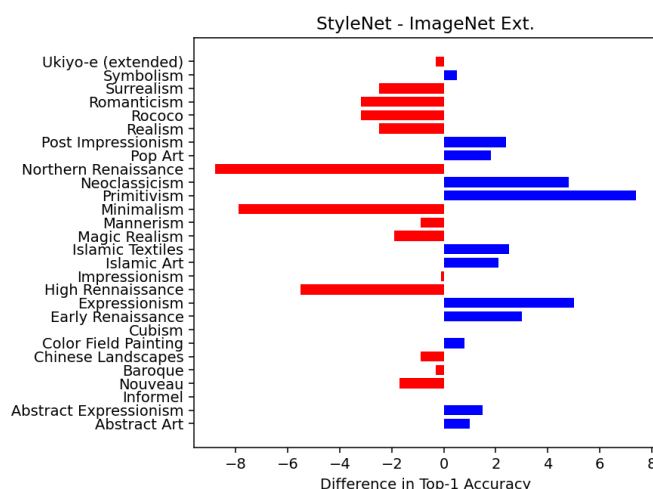


Figure 5. Difference in top-1 accuracy between the StyleNet and ImageNet models (blue means StyleNet is better). Using the Extended Dataset.

On the other hand, we see some gaps shrink, with Cubism becoming much more equal between the two models and Abstract Art now leaning more heavily towards the StyleNet model. One theory is that the ImageNet model that was used in this study is more advanced than the one that was originally retrained on Stylized ImageNet. We also see some gaps in accuracy on the new classes, but these are better understood by looking at Figure 6, where we can see that both models performed exceptionally well on the new classes, breaking 80% accuracy on the now-larger Ukiyo-e, as well as Islamic Textiles and Chinese Landscapes, and hovering just below 80% on Islamic Art. We hypothesize that this exceptional performance is due to the distinctness of the added and extended classes.

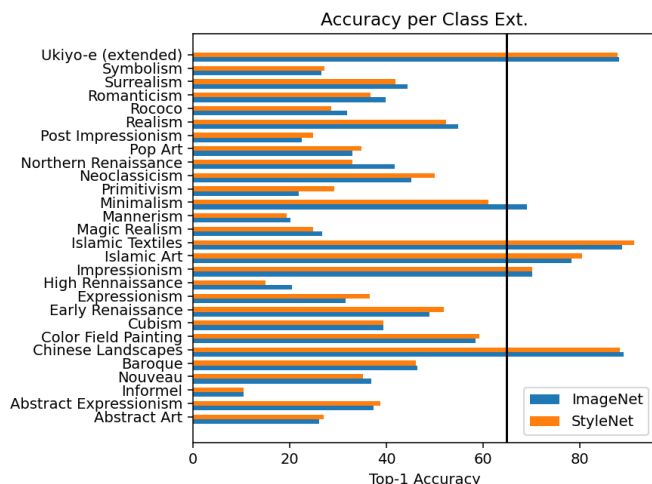


Figure 6. Bar Graph Comparing StyleNet and ImageNet model accuracies per class. Black line represents RASTA top-1 accuracy on the same dataset

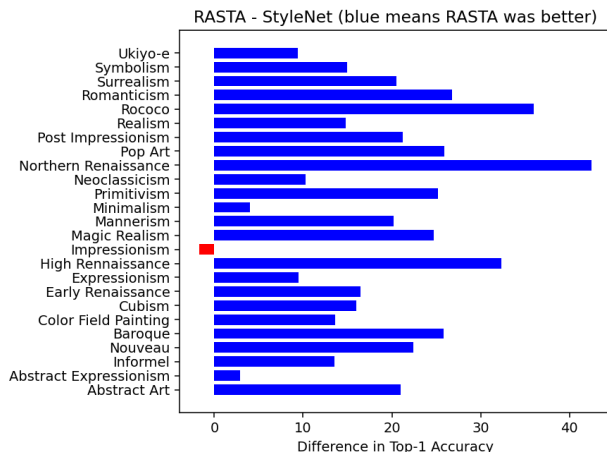


Figure 7. Difference in top-1 accuracy between the StyleNet and RASTA models (blue means RASTA is better)

3.1.2 StyleNet vs RASTA

For the sake of simplicity, moving forward we will focus our comparisons between the StyleNet model and the RASTA model. As expected and as seen above, the RASTA model benefited greatly from the expanded dataset, with a 5.6% jump in Top-1 Accuracy. Looking at how the models performed on the original dataset, we see improvements across the board(Figure 9). These improvements become even more pronounced when training off of the extended dataset, as seen in Figure 10. Notice, however, how the smallest differences were in the extended datasets. While this is not necessarily expected, it is explainable: we specifically chose to add classes that were unique and distinct from the rest of the original dataset, and as such the non-RASTA models performing nearly as well on these classes is not completely unexpected.

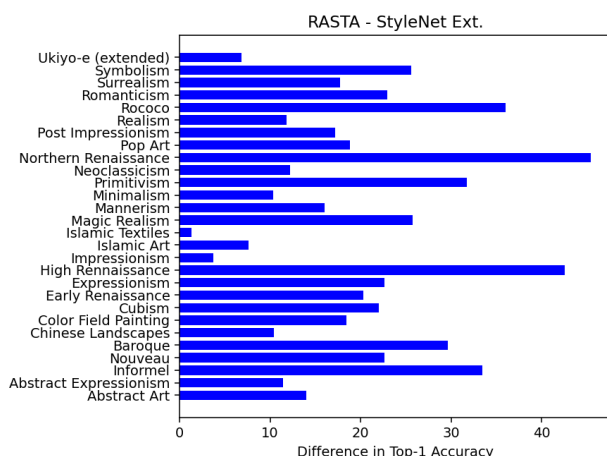


Figure 8. Difference in top-1 accuracy between the StyleNet and RASTA models (blue means RASTA is better). Extended Dataset is used.

3.1.3 RASTA vs RASTA Ext.

To fully isolate the effects of the extended dataset, we compare the differences between the RASTA model trained on both the base and extended datasets, as seen in Figure 11.

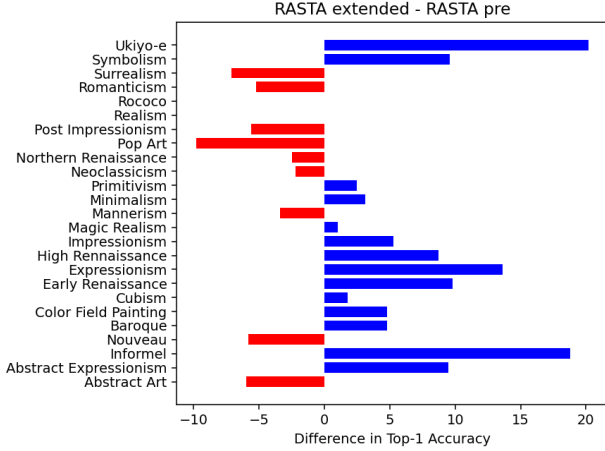


Figure 9. Difference in top-1 accuracy between the RASTA models (blue means the Extended Dataset Model is better). Keep in mind Ukiyo-e dataset is an extended dataset.

While the extended dataset did increase accuracy in many classes, there were some classes that suffered. These classes tended to be ones that are more varied, with multiple different kinds of textures and object styles. It is possible that with the increased diversity in classes, the model had more difficulty assigning multiple features to one class, and thus the accuracy suffered. This could be alleviated by deeper training, as well as more data points.

3.2. Discrepancies from the Original Paper

Throughout work for this paper, some discrepancies between our results and the original RASTA paper were found. Most of these discrepancies probably resulted from lack of access to the full dataset that was used in the original paper, which is not publicly available. We hypothesize that the effects of this were actually minimized due to general advances in classification models. To test this hypothesis, we trained another pre-trained Image Net model, this time a ResNext [12] model. This model outperformed our first ImageNet model as well as the ResNet model from the original paper.

3.3. Human Results

We collected data from 20 participants around the MIT community where each participant completed a match-to-sample task for 3 images per art style.

3.3.1 Art Style Categorization Accuracy

Using the data from our experiment, we were able to compute the sample human accuracy and standard deviation for each art style, shown in Figure 12. Averaging all art styles together, we get a human classification accuracy of 71%

with a standard deviation of 25%. We recognize that the variability of this data is quite high: increasing the amount of stimulus used per art style and increasing the amount of participants would help decrease this variance. As seen in Figure 12, the underrepresented art styles that were added to the data set have some of the highest classification accuracy and lowest standard deviation. Again, we believe this is due to the distinctness of the added styles. We also see that other unique art styles like Cubism were classified fairly well, while styles with broader classifications had low accuracy and high standard deviation.

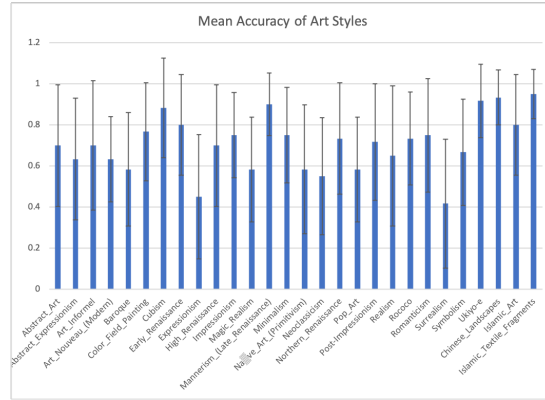


Figure 10. Accuracy and standard deviations for each art style using the human experiment data.

3.3.2 Multidimensional Scaling

We performed Multidimensional Scaling (MDS) to analyze the similarity between art styles based on the data collected in our human experiment, shown in Figure 13. We created a symmetric confusion matrix of dissimilarity for all art styles in our human experiment, with both the columns (X) and rows (Y) being each art style. The value between art style X and art style Y is the percentage that X was classified correctly when being compared to Y. This gave us the dissimilarity for each pair of art styles, where a higher value represents a higher dissimilarity. Using these dissimilarities, we used MDS to create a map where the distances between art styles is representative of how similar the art styles seemed to the participants. Art styles that are closer in distance seemed more similar to the participants and were confused more often, while farther art styles were not confused as often. This analysis follows our understanding of the art styles used. The underrepresented data we added is fairly close together, as well as art styles that are classified similarly (e.g. Northern and Early Renaissance).

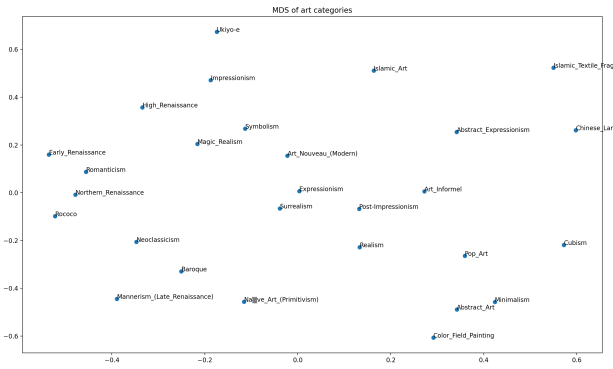


Figure 11. 2D representation of the similarity between art styles, computed using Multidimensional Scaling.

3.4. Comparing Model to Human Accuracy

In order to compare the classification accuracy of models and humans, we tested the re-trained RASTA model on the 84 images used in the human experiment. We calculated classification accuracy of the model for each art style by averaging the confidence percentage of the correct classification for the 3 images per style. We then compared the model's accuracy to the human accuracy computed in Section 3.3.1, shown in Figure 14. The model seems to outperform our participant data in several art styles, including most of the underrepresented styles. However, due to the few art styles the model struggles with (e.g. Abstract Art, Mannerism), the overall accuracy does not outperform the human data. The model has an overall accuracy of 63% on this data set, while the participants had an overall accuracy of 71%. (Keep in mind that our sample size is fairly small, with 3 images per class and 20 participants. We do not necessarily expect untrained participants to perform similarly over a larger set of stimuli. Additionally, our re-trained ResNext model performs more similarly to the subjects.)

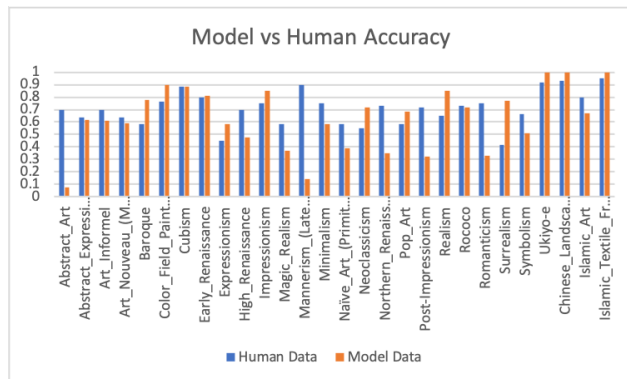


Figure 12. Art Style Classification Accuracy of the model vs Human.

4. Conclusions

Through our results, we see that the ImageNet and StyleNet models both performed decently well, but in different capacities. While there is not a clear winner between the two, the differences in accuracies are worth exploring. A model that is trained on stylized versions of the dataset, similar to the Bethge Lab paper, could learn to pickup more kinds of features and thus get the best of both models.

Additionally, our experiment with ResNext proved that newer Image Net trained models with better accuracy transferred this better performance to the art style classification task. Both the RASTA paper and the Bethge Lab paper are quite dated in terms of machine learning, releasing in 2017 and 2018, respectively. A one-to-one recreation of these papers with more modern models could boast improvements with no further tweaking.

Finally, our comparison of the model to human subjects showed that using models for art style classification shows promise. In Section 3.4, we saw the model significantly outperform the average participant in 12 out of the 28 categories, which bodes well for automated art style classification.

5. Future Work

Future work would be defined by a more faithful recreation of the papers. A possible workflow, which we foresee having sizable improvements, could be training a newer and better pre-trained Image Net model on Stylized ImageNet, and then deeply retraining on both the art style images and stylized art style images.

Additionally, during the course of this work we decided not to use Style Transfer to artificially expand any datasets. This was mostly due to lack of time as well as expertise in terms of Style Transfer calibration. In the future, using Style Transfer to artificially create more data points in the training set could help balance out the sizes of the classes and thus lead to better performance.

5.1. Difficulties

The biggest difficulties specific to the machine learning portion of this paper were resources. Training models is expensive, especially when there are 5 different models that need to be trained and tested, sometimes twice to validate results. We were luckily able to leverage the IBM-MIT Satori cluster, which gives anyone on campus access to NVIDIA Tesla V100s, which were powerful enough to train the RASTA models in 90 minutes and the non-RASTA models in 180 minutes. Of course, using this cluster was not without its own issues: the cluster architecture is PowerPC64 (more accurately IBM's Power9 chips), which lead to severe restrictions in terms of the software we were able to run. Due to a lack of binaries, we were only able to run

GPU-supported Tensorflow and Pytorch through IBM provided containers, both of which were not fully up-to-date. Solving this issue would both make training and testing easier, and allow for more experimentation.

More generally, data collection for paintings remains very challenging, and until a large and well-funded initiative sets out to fix that, it will remain that way. While images ARE available, they are not annotated. Until software engineers become art historians, we are at the mercy of other organizations and their efforts in mass art style classification.

5.2. Code Availability

Code is available at [Sami's Github](#). Note that the code is shared for transparency, and no attempt was made to make it easy to reproduce: despite this, the code is mostly self-documenting.

References

- [1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. [3](#)
- [2] François Chollet et al. Keras. <https://keras.io>, 2015. [3](#)
- [3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. [1](#), [2](#)
- [4] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019. [1](#), [2](#)
- [5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. [1](#), [2](#)
- [6] David Kadish, Sebastian Risi, and Anders Sundnes Løvlie. Improving object detection in art images using only style transfer. *CoRR*, abs/2102.06529, 2021. [1](#)
- [7] Adrian Lecoutre, Benjamin Negrevergne, and Florian Yger. Recognizing art style automatically in painting with deep learning. 2017. [1](#), [2](#)
- [8] The Met. The met dataset, 2020. [2](#)
- [9] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. [2](#)
- [10] Jonathan Peirce, Jeremy R. Gray, Sol Simpson, Michael MacAskill, Richard Höchenberger, Hiroyuki Sogo, Erik Kastman, and Jonas Kristoffer Lindeløv. PsychoPy2: Experiments in behavior made easy. *Behavior Research Methods*, 51(1):195–203, Feb. 2019. [3](#)
- [11] wikiart.org. Visual art encyclopedia, 2022. [1](#), [2](#)
- [12] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. *CoRR*, abs/1611.05431, 2016. [6](#)
- [13] Alice Xue. End-to-end chinese landscape painting creation using generative adversarial networks, 2020. [2](#)