Forged Renaissance: Using Deep Learning to Differentiate Human-Produced Art from AI Art

Alexis Amoyo, Olivia Mei, Chelsea Wang

1 Introduction

AI art generators have been under heavy scrutiny because of ethical issues relating to training art generator models using art created by artists who have not given their consent, are not getting paid, and are not credited for their work. Artists' names will be fed into prompts in order to generate artwork with their style, creating works very similar to their original art, without having to hire them. Because of these concerns, there have been detectors developed to identify whether an artwork was generated by AI or created by an artist with varying degrees of accuracy. As new AI art generators are being developed and provided to the public, it is important to improve and develop new methods to accurately classify whether a given piece of art is machine or man-made. As a result, we chose 5 different machine learning models: ResNet50, ResNet101, VGG19, ViT-B32, and ViT-L32 to train on the same datasets and compare how well each of them performs.

2 Literature Survey

Extensive research in the field of image classification has established that Convolutional Neural Networks (CNN) and Visual Transformers (ViT) are the most effective architectures for our purpose.

Epstein et al. previously found that a straightforward CNN architecture could be used to classify photographs as either AI-generated or not with high accuracy. However, when the CNN models are applied to data from unseen AI image generators, it is no longer able to accurately differentiate the photos, especially with major architectural changes in the generators [5].

Another related study by Ngyuen et al. attempts to detect AI-generated artwork of human faces, created by a generative adversarial network (GAN), by using different CNN models. These included CDCN, a convolutional network created to detect image spoofing, and ConvNeXt, a modernized CNN that competes with Transformers for visual recognition. They found that these two models performed with high accuracy on their dataset and that combining them into an ensemble model further improved the results [12].

A common CNN architecture implements a residual learning framework for ease of training and overall increased performance. These Residual Networks (ResNets) use residual learning reformulation, which allows the model to introduce a shortcut or skip connection that allows the input to bypass one or more layers and directly contribute to the output. Multiple residual blocks are then stacked in multiple layers on top of each other, with depths varying from 18 layers to 152. With greater residual blocks, the model is able to capture more complex features of the data. These models demonstrate state-of-the-art performance and are demonstrative of foundational deep-learning models for image classification and segmentation [6].

Mascarenhas et al. conducted a comparison between VGG16, VGG19 and ResNet50 for image classification with a dataset of 6000 images and split it 70:30 for testing and training sets. It was found that ResNet50 had the best accuracy among the 3 models used [11].

Prior research includes other architectures that implement self-attention by dividing an image into patches which are then embedded as input into a standardized transformer, showing comparable results to CNN models. These Vision Transformers (ViT), which when compared to the performance of state-of-the-art models such as BiT-L composed of ResNet-based architectures, require less computational power for improved performance on both its "Large" and "Huge" models when pre-trained on large datasets. Thus, ViT offers an alternative architecture that could potentially outperform a state-of-the-art CNN due to its ability to better capture the context of an image as well as its potential for fine-tuning and patch extraction from CNN feature maps [4].

A more advanced Vision Transformer we investigated was Swin Transformer - a hierarchical Vision Transformer implementing a unique shifted windows technique that limits self-attention to non-overlapping "windows" that are later connected [10]. Feature maps are built similarly to ViT in that they divide an image into similar patch structures, but each successive layer should shift the partitioning of these patches to form windows of varying sizes to better capture key features of the image. This is further aided by the model's relative position bias, which can be utilized in fine-tuning processes to set the weights of the model in regard to window size. Through a combination of patch merging along with feature transformations, a hierarchical representation is produced with similar feature map resolutions to typical CNN architectures such as ResNet [1].

3 Methodology

3.1 Model Selection

We aimed to explore and compare two prominent approaches for image classification problems: Convolutional Neural Networks (CNNs) and Vision Transformers (ViT). Our goal was to leverage both these techniques, testing multiple models within each category. Specifically, we evaluated three distinct CNN models – ResNet50, ResNet101, and VGG19 – alongside two variations of ViT – ViT-B32 and ViT-L32.

ResNet50 is a residual network that uses 50 layers with 3-layer bottleneck blocks to reduce the computational cost [6]. This model uses a 50-layer plain network architecture inspired by the VGG-19 model and applies shortcut connections, which skip layers without reducing performance; this is generally an effective way to deal with the vanishing gradient problem that is encountered when training deep neural networks. We chose this model as it is known for image-related tasks, including image classification.

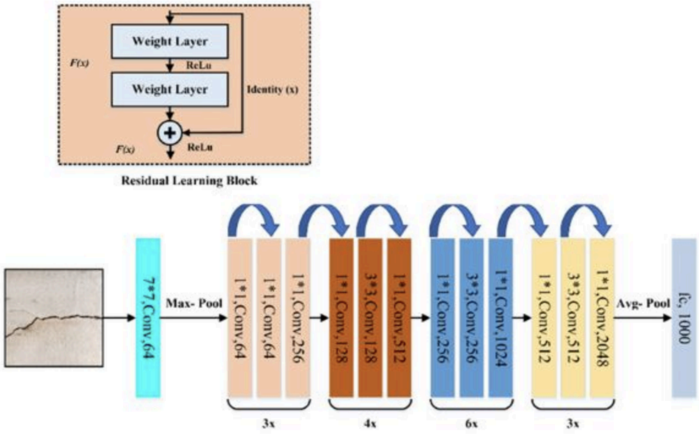


Figure 1: ResNet50 architecture [8]

ResNet101 is similar to ResNet50, with the difference being that ResNet101 uses 101 layers in contrast to the latter's 50. This increased depth allows ResNet101 to capture more complex features in data compared to ResNet50 generalizing it to a broader range of data and allowing deeper layers to better conceptualize higher-level features as well as the more complex aspects that a 50-layer ResNet model may not be able to do.

VGG19 is a standard deep convolutional neural network that uses 19 layers to classify images, with 16 convolutional layers and 3 fully connected layers [6]. These convolutional layers use a minimal receptive field of 3x3 followed by a max pooling layer that extracts the maximum value to reduce computational complexity while the fully connected layers act as classifiers. We selected this model as it is known for its simplicity to compare its performance to the other models.

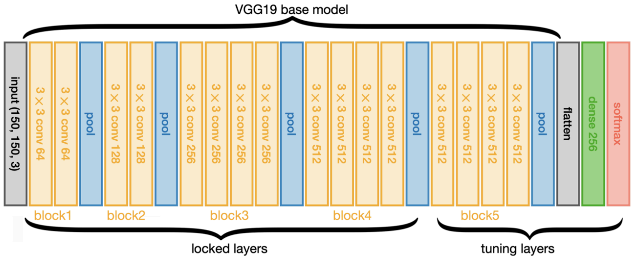


Figure 2: VGG19 architecture [3]

ViT splits a given image sample into equally-sized patches, these patches are then flattened and put through a transformer that contains processing elements of layer normalization, multi-head attention network, and multi-layer perceptrons. The application of these elements allows the model to adapt to images during training, focus on regions in an image, and provide classification labels respectively. It is to be noted that the ViT model's accuracy will be low if it is trained on a small dataset. Compared to ResNet50, ViT retains more spatial information and may be able to better capture key feature relational positions within an image. We chose this model due to how it parses a given image sample into patches to compare to the performance of the ResNet50 model.

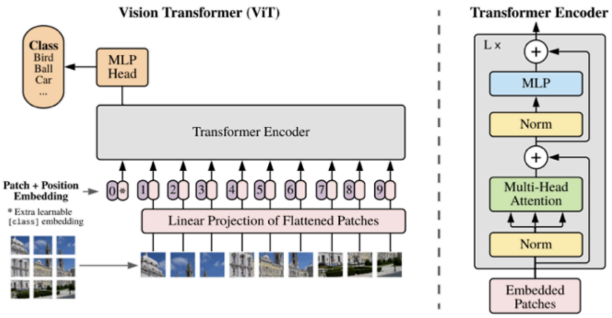


Figure 3: ViT Transformer architecture [4]

3.2 Selection of Evaluation Metrics

To evaluate our model, we decided to look at the metrics: accuracy, precision, recall, and F1 score. Because we selected our dataset so that it was balanced, accuracy is both a reliable measure and intuitive to understand in its ability to assess the models' classification abilities. To better assess the trends in the models' learning, training and validation accuracy were tracked over the total number of epochs, as were their respective losses. However, it is important to utilize additional metrics to achieve a more complete picture of how well the models are performing, so F1 scores were measured to more comprehensively evaluate the models' effectiveness on false positives and false negatives. Precision and recall are also essential in determining a model's potential for errors and the potential impacts of misclassification. These measures are all commonly used and allow us to compare our performance with previous studies.

4 Implementation

4.1 Data Collection and Preprocessing

The first dataset we chose for this study contains 250k AI-generated art samples from MidJourney [13]. We chose to use MidJourney because it is one of the most popular AI art generators and because there was a Kaggle dataset containing both the prompts and the artwork generated from a MidJourney bot on a Discord Server. Out of these 250k images, we extracted approximately 2000 samples that were landscape-related. The samples obtained from this dataset represent our AI-generated class. Only 500 samples from the extracted data were used for the AI art dataset.

The second dataset was from WikiArt, an online encyclopedia that aims to make visual arts accessible to anyone for free [9]. We decided to obtain half of our human-created art class from this dataset because it contains a diverse array of traditional art. In addition, the dataset was available on Kaggle, as someone had previously scraped the website from various Discord prompts. As a result, we randomly selected 500 landscape samples to represent traditional human-made art.

The third part of our dataset includes human-created art from ArtStation, a platform for artists to showcase their art [2]. We specifically chose ArtStation because it is the platform that many professional game artists will use to display their portfolios, so we know the quality of the artwork will be high. In addition, it contains a free API that allows for the filtering out of AI-generated art and non-digital art mediums. We collected around 500 samples of digital human-created landscape art by utilizing a scraping tool for ArtStation available on GitHub [7]. By selecting 250 samples from our selected WikiArt images and 250 samples from our selected ArtStation images to create the human art dataset, we ensured that the 500 samples of human art contained a good mix of traditional and digital media. By using 500 AI art samples and 500 human-created art samples as our dataset, we hope to avoid class imbalance.

After collecting the data, extensive preprocessing steps were performed. We manually looked through our samples to ensure that the images were of landscapes because the presence of the word in a prompt or search term does not guarantee that the artwork is of a landscape. Once some of the unsatisfactory samples were filtered out, the MidJourney dataset had to be reformatted because each sample contained four different AI-generated images for the same prompt. We then resized all RGB images to 224 x 224 pixels before converting them to Numpy arrays, normalized the arrays to values between 0 and 1, and randomized the samples. By taking these steps, we were able to maintain consistency across the data from the three different sources and limit the amount of noise in our dataset. After our data was preprocessed, the data was randomly split into 20% for testing and 80% for training.

4.2 Model Details

The models ResNet50 and ResNet101 were trained by implementing the Keras library in Python. We set up our ResNet50 Base by pre-training the model on the ImageNet dataset and froze the weights of the pre-trained layers to better retain the learned features. Upon this basis, the output was then flattened to a one-dimensional tensor before adding a final dense layer for binary classification using the sigmoid activation function. The model was then compiled with the Adam optimizer with a learning rate of 0.001 using Binary Cross-Entropy as the loss function.

The ResNet50 Augmented model was built similarly to our base model, except there was an additional dense layer of 256 units and ReLU activation function so the model could better learn the patterns from the flattened layer, as well as a dropout layer of 0.5 to prevent over-reliance on specific neurons.

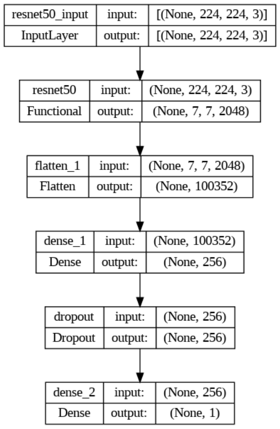

resnet50_input | input: | [(None, 224, 224, 3)]
InputLayer | output: | [(None, 224, 224, 3)]
resnet50 | input: | (None, 224, 224, 3)
Functional | output: | (None, 7, 7, 2048)
flatten_1 | input: | (None, 7, 7, 2048)
Flatten | output: | (None, 100352)
dense_1 | input: | (None, 100352)
Dense | output: | (None, 256)
dropout | input: | (None, 256)
Dropout | output: | (None, 256)
dense_2 | input: | (None, 256)
Dense | output: | (None, 1)

Figure 4: ResNet50 Augmented

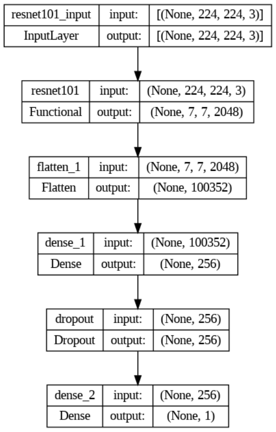

resnet101_input | input: | [(None, 224, 224, 3)]
InputLayer | output: | [(None, 224, 224, 3)]
resnet101 | input: | (None, 224, 224, 3)
Functional | output: | (None, 7, 7, 2048)
flatten_1 | input: | (None, 7, 7, 2048)
Flatten | output: | (None, 100352)
dense_1 | input: | (None, 100352)
Dense | output: | (None, 256)
dropout | input: | (None, 256)
Dropout | output: | (None, 256)
dense_2 | input: | (None, 256)
Dense | output: | (None, 1)

Figure 5: ResNet101 Augmented

The ResNet101 Base and the ResNet101 Augmented models were constructed using the same configuration, where the additional layers, functions, and compilation parameters are identical. The only difference in our implementation is the Keras model that we used.

The VGG19 model was implemented with the Keras library and follows the same setup as our ResNet50 Base, with the addition of early stopping dependent on the validation loss and if there is no improvement over 3 epochs to avoid overfitting.

ViT-B32 and ViT-L32 were created using the Keras-ViT library, which implements a Keras-based ViT model based on the paper "An image is worth 16x16 words: transformers for image recognition at scale" [4]. Both of these models were pre-trained on imagenet21K weights with the sigmoid activation function.

For each model we trained and tested, we used the actual values of the test set and the predicted values the model provided as parameters for the precision_recall_fscore_support function from Scikit-learn to obtain the precision, recall, and F1-score metrics to evaluate each model.

5 Results

The VGG19 model's implementation of early stopping resulted in VGG19 being trained for 10 epochs while the other models were trained for 50 epochs.

Of the models attempted, results showed consistently high precision, recall, and F1 scores across both human-created and AI-generated art (Table 1). For the ResNet50 models attempted, human-created art had a lower rate of false positives than AI-generated art, but this was found to be lower for both cases in the augmented version of the model. The ViT models also offered high precision values for human-created art. As for the ResNet101 models, precision was significantly higher for AI-generated art. The VGG19 model provides high precision values for AI-generated art and human-created art, with the precision for the human-created art being slightly higher than the precision for the AI-generated art. A higher precision indicates that the model predicts a low occurrence of false positives for the class and is more likely to be correct in its classification; in this case, a false positive for AI-generated art is classified as the model mistakenly identifying human-created art as AI-generated. On the other end of the spectrum, a false positive for human-created art occurs when the model incorrectly classifies AI-generated art as human-created.

As for recall, a higher value for this metric implies that the model predicts a low rate of false negatives, defined as when the model fails to recognize AI-generated art and attributes it to media created by humans or vice versa. Given that recall generally has an inverse relationship to precision, it can be seen in

Table 1 that recall was significantly greater in the ResNet101 model for human-created art than AI-generated art. ResNet50, contrastingly, has a higher recall for AI-generated art; similarly, the ViT models showed similar results. The recall for AI-generated art using the VGG19 model is slightly greater than the recall for human-created art.

Metrics	**Precision**		**Recall**		**F1 Score**		**Accuracy**
Dataset	**Human Created Art**	**AI Generated Art**	**Human Created Art**	**AI Generated Art**	**Human Created Art**	**AI Generated Art**	**Cumulative**
ResNet50 Base	93.94%	74.62%	64.58%	96.15%	76.54%	84.03%	81.00%
ResNet50 Augmented	85.06%	80.53%	77.08%	87.50%	80.87%	83.87%	82.50%
ResNet101 Base	75.42%	91.46%	92.71%	72.12%	83.18%	80.65%	82.00%
ResNet101 Augmented	68.89%	95.38%	96.88%	59.62%	80.52%	73.37%	77.50%
VGG19	92.55%	91.51%	90.63%	93.27%	91.58%	92.38%	92.00%
ViT-B32	88.89%	75.00%	66.67%	92.31%	76.19%	82.76%	80.00%
ViT-L32	85.53%	75.00%	67.71%	89.42%	75.58%	81.58%	79.00%

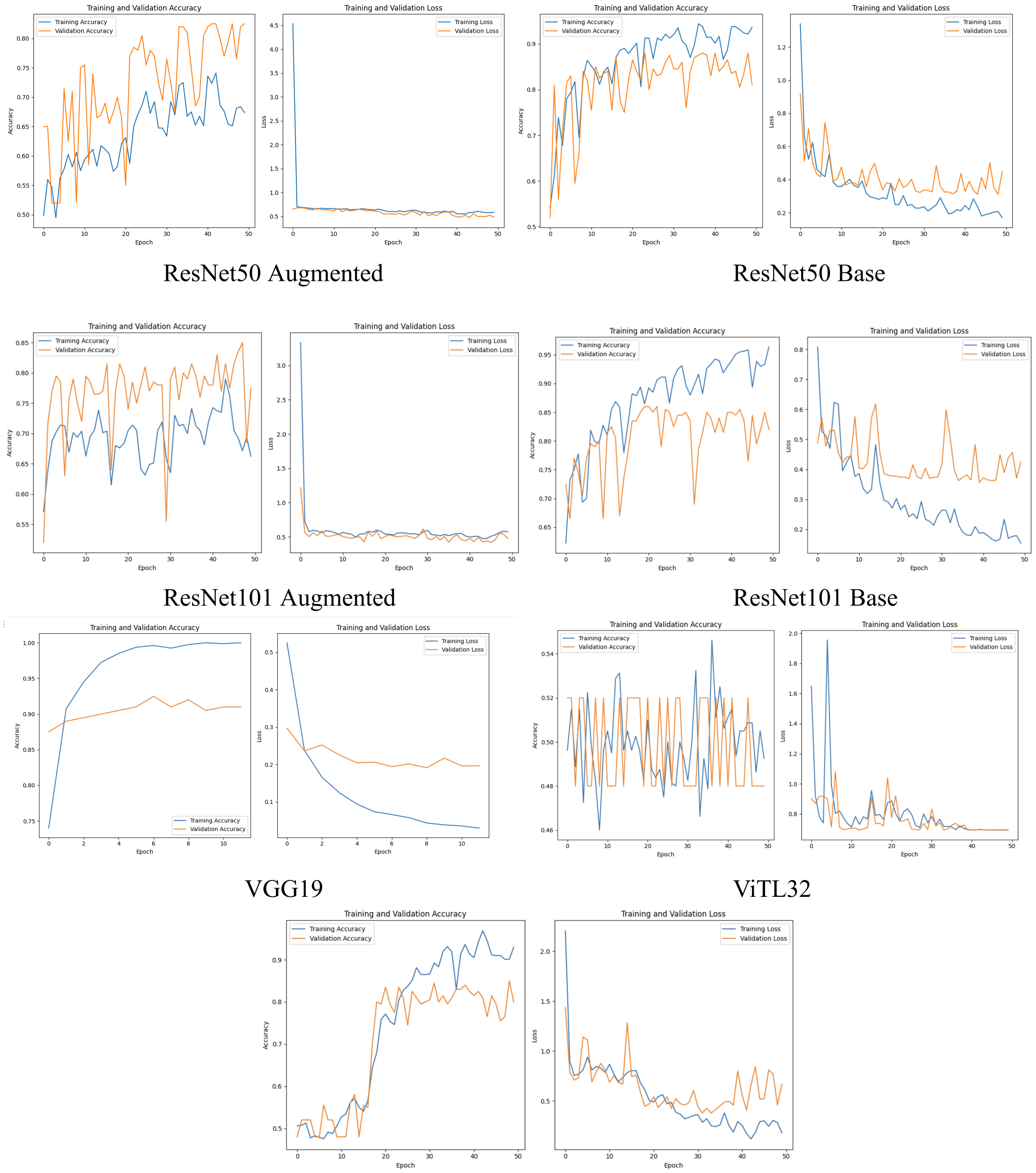
Table 1: Precision, Recall, F1-Score, and Accuracy across all models attempted.

F1-Score, which is the harmonic mean between precision and recall, offers a balance between precision and recall to assess the model's effectiveness in classification. VGG19 has the highest F1 scores for AI-generated art. As for human-created art, the VGG19 model had the highest F1-Score. The F1-Scores using ResNet50, ResNet101, and ViT models are relatively close together.

The accuracy of the VGG19 model used was the most consistent across training and validation, achieving a final accuracy of 92.00% on the test set. The other models achieved significantly lower accuracies upon evaluation, of which the next highest accuracies are listed in the following: ResNet50 Augmented, ResNet101 Base, ResNet50 Base, and ViT-B32. The simplest model of the models we used had the best accuracy.

It may also be noted that, in our experimental results, we observed erratic behavior in both training and testing accuracy throughout the training process for all models except for VGG19. The models' accuracy exhibited pronounced fluctuations, deviating from a consistent upward trend. This erratic pattern could be attributed to the inherent complexity of the task, leading to challenges in achieving stable convergence.

This may be related to how the data was processed and trained, or it may be related to the nature of the dataset used. The MidJourney dataset contains multiple copies of similar images due to the nature of using the same prompt to generate images. It was also manually parsed for landscape images by excluding prompts that included certain keywords, such as "people," "man," or "woman." This may be related to the fluctuations of the graphs in Figure 6 due to a lack of diversity in the dataset that could have been selected more strictly.



ResNet50 Augmented

ResNet50 Base

ResNet101 Augmented

ResNet101 Base

VGG19

ViTL32

ViTB32

Figure 6: Training and Validation Accuracy and Loss graphs of models

6 Conclusion

Each of the models used for experimentation were able to characterize the differences between machine-made art and man-made art with better-than-average accuracy. Each model was able to achieve accuracy between 77% and 92%, and the high values for precision and recall indicate a relatively balanced performance as seen in the relatively high F1 scores across both classes. At first glance, these results show that the models are able to distinguish between the various features of the different classes, and therefore have a generally robust ability to correctly predict common trends within AI-generated and human-created art.

While our overall accuracy and F1 scores seem favorable for the CNN (ResNet50, ResNet101, and VGG19) models, these values may be misleading. When we observe the training and validation accuracy graphs for the models, the accuracy exhibits erratic fluctuation across epochs for all models except for VGG19. This could be a result of the model overfitting. When we decreased the complexity of our model, the irregularity of the accuracies also appeared to decrease. This can be seen in VGG19's training and validation accuracy graph in Figure 6. When we compare ResNet50 to ResNet101, the data doesn't seem to vary as dramatically for ResNet50. Similarly, when we compare the Augmentation to the Base models, the Base models with fewer layers seem to perform more closely to our expectations. In order to prevent overfitting in the future, we could find ways to improve our dataset. One of the issues we currently have is the suboptimal quality of the MidJourney samples. This data was originally pulled indiscriminately from a few public Discord servers, resulting in some samples containing duplicate or similar images. In addition, we selected the data by choosing all images that contained the keyword "landscape" in the prompt, resulting in some samples where the landscape wasn't the focus of the artwork. When these issues are combined with our small dataset size, the dataset lacks the diversity required for the model to learn the underlying patterns of the data. Instead, the model may be detecting attributes that are specific to our dataset in particular, causing overfitting. With the ever-growing range of AI-generated art available, encompassing a larger spectrum of this kind of art outside of MidJourney would be beneficial in generalizing the data to be more robust. In future work, it would be best to increase the number of AI-generated art samples, either from different platforms or more selective prompts.

Despite the constraints when obtaining a diverse and large set of relevant data, our ability to train a precise and non-overfitted model by using simpler architecture reveals the potential for models to accurately differentiate art created by AI and by humans. With a better, more diverse dataset, we may be able to train on the relatively more complicated algorithms presented in this paper, as well as other state-of-the-art image classifiers such as the Swin Transformer. Such models have had great success in both classification and segmentation, and this may show promising results in the future. There is also potential in training models such as CLIP, which specializes in understanding and generating visual content from text-based descriptions. Once we can apply more complex models, we may be able to generalize the samples better resulting in models that can better detect art generated by AI.

References

[1] Aburass, S., & Dorgham, O. (2023, October). Performance Evaluation of Swin Vision Transformer Model using Gradient Accumulation Optimization Technique. In Proceedings of the Future Technologies Conference (pp. 56-64). Cham: Springer Nature Switzerland. https://arxiv.org/pdf/2308.00197.pdf

[2] ArtStation. (2014). Home. ArtStation. https://www.artstation.com/

[3] Chachra, G., Kong, Q., Huang, J. et al. Detecting damaged buildings using real-time crowdsourced images and transfer learning. Sci Rep 12, 8968 (2022). https://doi.org/10.1038/s41598-022-12965-0

[4] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

[5] Epstein, D. C., Jain, I., Wang, O., & Zhang, R. (2023). Online detection of ai-generated images. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp.382-392).

[6] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778). https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html

[7] Hueyning. (2020). Art Station Scraper [Software]. GitHub. https://github.com/hueyning/art-station-scraper

[8] Javaid, S., Rizvi, S., Ubaid, M., Darboe, A., Mayo, S., “Interpretation of Expressions through Hand Signs Using Deep Learning Techniques,” Int. J. Innov. Sci. Technol., vol. 4, no. 2, pp. 596–611, 2022. https://journal.50sea.com/index.php/IJIST/Deep-Learning-Techniques

[9] Kaggle. (2022). WikiArt - GANgogh: Creating Art with GAN [Data set]. Kaggle. https://www.kaggle.com/datasets/ipythonx/wikiart-gangogh-creating-art-gan

[10] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., ... & Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 10012-10022).https://arxiv.org/abs/2103.14030

[11] Mascarenhas, S., and Agarwal, M., "A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for Image Classification," 2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON), Bengaluru, India, 2021, pp. 96-99, doi: 10.1109/CENTCON52345.2021.9687944.

[12] Nguyen, M. Q., Ho, K. D., Nguyen, H. M., Tu, C. M., Tran, M. T., & Do, T. L. (2023, October). Unmasking The Artist: Discriminating Human-Drawn And AI-Generated Human Face Art Through Facial Feature Analysis. In 2023 International Conference on Multimedia Analysis and Pattern Recognition (MAPR) (pp. 1-6). IEEE. https://ieeexplore.ieee.org/document/10289113

[13] Turc, I., & Nemade, G. (2022). Midjourney User Prompts & Generated Images (250k) [Data set]. Kaggle. https://doi.org/10.34740/KAGGLE/DS/2349267