Enhancing AI-Generated Image Detection: A Comparative Study of CNNs, Transformers, and Contrastive Learning

DL4DS Boston University

Viktoria Zruttova, Junhui Cho, Cordell Cheng

February 26, 2025

Abstract

AI has reached a point where it can generate highly realistic faces, scenes, and objects. This study addresses the problem of distinguishing AI-generated visuals from authentic photographs using a unique dataset, "AI vs. Human-Generated Images," from a Kaggle competition. Unlike conventional datasets, this dataset provides paired images where each real image has a corresponding AI-generated counterpart, allowing for direct comparative analysis. We leverage this structured pairing within a deep learning framework, incorporating convolutional neural networks (CNNs) and transformer-based architectures to develop robust classifiers. In addition, we explore contrastive learning to enhance feature discrimination, hypothesizing that it improves generalization by enforcing a more distinct separation between real and AI-generated images.

Code Repository: The complete implementation and experiments are available on GitHub at: `https://github.com/jh000107/AI-Image-Detector`

Introduction

AI-generated images – including deepfakes and other content created by generative adversarial networks (GANs) or diffusion models – have advanced to a point of being almost indistinguishable from real images. These technologies enable the creation of hyper-realistic faces, scenes, and objects, raising concerns about the spread of misinformation and the erosion of trust in digital media. Early research in this domain focused on deepfakes of human faces (e.g., face swaps or manipulations), but the challenge now extends to natural images generated by AI (such as artworks or scenery), which has only recently become a focus of studies [2]. Detecting AI-generated content reliably is difficult because modern fakes closely mimic the appearance of genuine images, often containing only subtle inconsistencies or artifacts [1]. Some detection approaches have exploited such artifacts – for example,

detecting unnatural blending, color inconsistencies, or frequency-domain irregularities [16]. However, as generation methods improve and remove obvious artifacts, detectors that rely on fixed cues may become fragile [8].

Another major challenge is the generalization of detectors to new or previously unseen types of AI-generated images. Many detection models perform well when evaluated on fake images similar to those they were trained on, but their accuracy degrades significantly on novel manipulation techniques [16, 18]. In practice, a detector must handle an open-set scenario, where the specific characteristics of fakes at test time may differ from the training data. Limited generalizability can hinder real-world deployment of deepfake detectors, as they may fail to recognize new attack methods. For instance, a model trained on faces manipulated by one GAN might not detect faces generated by a different, more advanced GAN [11]. This issue has been highlighted in studies showing that cross-dataset detection performance can drop sharply compared to in-dataset results. Developing techniques to make detectors more robust against unseen manipulations is therefore an active area of research.

In this context, we consider two prominent types of deep learning architectures for image classification: CNNs and vision transformers. CNNs have been the cornerstone of image recognition for years, excelling at learning hierarchical feature representations via convolution and pooling operations. Transformers, on the other hand, employ self-attention mechanisms to capture long-range dependencies in the image. Vision transformers have shown promise in image classification tasks, but they often require large training datasets and are computationally intensive, whereas CNNs come with strong inductive biases for locality that make them effective even with limited data [9]. In the realm of deepfake detection, CNN-based models (e.g. Xception, EfficientNets) initially dominated and achieved high accuracy on benchmark datasets, but researchers are now exploring transformer-based models (such as ViT or Swin Transformer) for potentially improved performance. Comparing these two architecture families in the specific context of AI-generated image detection is important for understanding their relative strengths. For example, CNNs might better capture fine-grained pixel artifacts, while transformers could detect inconsistencies in global image context. Recent work has indeed suggested that each may have unique advantages for deepfake detection.

Finally, contrastive learning has emerged as a promising technique to improve representation learning for classification tasks. In contrastive learning, models learn by pulling semantically similar examples closer and pushing dissimilar examples apart in the feature space. We hypothesize that contrastive learning can be especially beneficial for distinguishing real vs fake images, as it can encourage the model to develop a representation that maximally separates authentic images from AI-generated ones. Prior studies have begun to apply contrastive learning to deepfake detection to address generalization issues [16, 14, 10, 13].

"AI vs. Human-Generated Images" is a dataset introduced specifically for authenticity detection. The dataset consists of authentic images from Shutterstock paired with AI-generated counterparts created using cutting-edge generative models such as DeepMedia. In other words, for each real image, there is a closely matching synthetic image depicting the same or similar content. This structured pairing enables a direct comparison between real and AI-generated content, providing a robust foundation for model training and evaluation.

The insights from this work have significance for digital content verification, helping pave the way for tools that can automatically flag AI-generated images in news media, social networks, or other platforms – a key step toward mitigating the spread of misleading visual content.

Related Work

CNNs and Transformers for AI-Generated Image Detection

Deep learning has shown great promise in tackling the AI-image detection problem. In particular, convolutional neural networks (CNNs) have long been the backbone of image classification and have been applied successfully to discriminate between real vs. generated images. CNNs excel at learning local visual features and textures, which is advantageous when AI-generated images contain subtle pixel-level artifacts [6]. Indeed, recent work by Bird and Lotfi achieved about 93% classification accuracy distinguishing real photographs from diffusion-generated fakes using a CNN on the CIFAKE dataset [3]. Their experiments revealed that the CNN's attention was not on the main subjects of images but rather on background irregularities, suggesting these models identify imperceptible noise patterns left by generation processes [4].

Complementing CNNs, transformer-based vision models (Vision Transformers, ViTs) have emerged as powerful image classifiers by capturing global context through self-attention mechanisms. Transformers consider the entire image patch relationships, potentially enabling the detection of more holistic anomalies in AI-generated images that might evade localized feature detectors [5]. Hossain et al. explored both CNN and ViT architectures for AI-generated image detection, finding an optimized CNN slightly outperformed the transformer, reaching 96.3% accuracy on a similar real-vs-fake image task [7]. This suggests that, despite transformers' success in general vision tasks, CNNs with their inductive bias for texture may still hold an edge in detecting the current generation of fakes. In practice, a hybrid or ensemble approach could leverage both local and global feature modeling to improve detection robustness [6].

CNNs and Transformers-Based for Deepfake Detection

CNNs have been the foundation of most early deepfake image detectors. Researchers have

applied off-the-shelf architectures (pretrained on ImageNet) and fine-tuned them to classify images as real or fake. For instance, the Xception network was used by Rößler et al. in the FaceForensics++ benchmark and achieved high accuracy in detecting manipulated facial images [15, 11]. In general, CNN detectors can achieve excellent in-dataset performance; one recent study reported accuracy above 97% (with AUC near 99%) on certain deepfake benchmarks using a CNN model [15]. Common CNN-based approaches often look for subtle artifacts left by the generation process – for example, unnatural textures, missing reflections, or eye and facial aberrations in deepfake faces. Some methods explicitly analyze frequency-domain patterns, operating on the observation that GAN-generated images may have spectral discrepancies from natural images [16]. Despite their success on known data, a well-documented drawback of these CNN classifiers is poor generalization. When evaluated on a different dataset or a new type of fake, their performance can drop drastically. This is because CNNs may overfit to particular artifacts or data characteristics present in the training set. For example, a model trained to detect FaceSwap manipulations might latch onto compression artifacts or blending errors specific to that method; if confronted with a NeuralTextures-generated fake (with different artifact signatures), it might fail to detect it. Thus, improving the general robustness of CNN-based detectors remains a key issue.

Transformers process images in a fundamentally different way, using global self-attention to relate patches of an image, which could be advantageous for detecting inconsistencies that span across an image (such as misaligned lighting or context that CNNs looking at local patches might miss). Vrizlynn Thing recently compared several CNN and transformer architectures on multiple deepfake image datasets. That study found that both kinds of models can attain high accuracy on deepfake detection given sufficient training data, with top performances exceeding 90% accuracy and 99% AUC on benchmarks like FF++ and Google's DeepFake Detection dataset [15]. Notably, a Swin Transformer (a type of vision transformer) was shown to perform on par with a strong CNN (ResNet) on the challenging DFDC dataset (Facebook's Deep Fake Detection Challenge). This suggests that transformers are a viable alternative to CNNs for this task. Nonetheless, transformers often require careful training and regularization when data is limited. A general observation is that CNNs bring strong local feature biases that are helpful for detecting pixel-level artifacts, whereas transformers might better capture global anomalies (e.g., a face that doesn't match its surroundings). Combining these strengths is an open research question.

Contrastive Learning for Fake Image Detection

In recent studies, contrastive learning has been applied to address the aforementioned generalization problem. The core idea is to learn an embedding where authentic vs. synthetic images are well-separated. Xu et al. employed a supervised contrastive learning (SupCon) loss to train a deepfake detector, forcing representations of real images to cluster together and away from representations of fake images [16]. By doing so, the model learned to emphasize features that consistently distinguish real from fake rather than features specific

to one fake type. In a true open-set evaluation (where the model was tested on an entirely novel fake type), their contrastive-learning-based model achieved about 78.7% accuracy, significantly outperforming a standard CNN, and further improved to 83.99% when combined with an Xception model in a fusion approach. This demonstrates that contrastive training can yield more generalizable detectors. Other works have explored combinations of unsupervised and supervised contrastive learning. One approach first learns features without labels by maximizing agreement between two augmented views of the same image (unsupervised contrastive pretraining) and then fine-tunes with supervised contrastive or classification loss [17].

Methodology

Our methodology involves a multi-stage deep learning pipeline comparing convolutional neural networks (CNNs), vision transformers (ViTs), and contrastive learning-based models for the task of AI-generated image detection.

Data Processing and Augmentation

We used a Kaggle-provided dataset of labeled real and AI-generated images. All images were resized and normalized using ImageNet mean and standard deviation. For training, we applied a variety of augmentations to enhance generalization, including:

- Random horizontal/vertical flips and rotations
- Color jittering and grayscale transformation
- RandAugment and RandomErasing for CNNs
- Perspective distortion for transformers

Train/validation splits were created using stratified sampling to preserve class balance.

Model Architectures and Training Strategy

We evaluated three categories of models:

1. CNN-Based Models: We used ResNet50, ResNeXt50, and EfficientNetB3, each pretrained on ImageNet. We replaced the classification head with a custom multi-layer perceptron (MLP) featuring dropout and batch normalization. Models were trained with AdamW optimizer and cosine annealing learning rate scheduling. Training was monitored using wandb.

2. Vision Transformer Models: We experimented with various ViT models (e.g., ViT-Tiny/16, ViT-Small/8, ViT-B/32), implemented using the timm library. We customized the transformer head similarly with a deeper MLP. Augmentation strategies were tailored

to capture global context and discourage overfitting. Transformers were also trained with AdamW and CosineAnnealingLR. Training was monitored using wandb.

3. **Supervised Contrastive Learning (SupCon):** We implemented a two-stage pipeline:
   - **Stage 1: Contrastive Pretraining** — We trained an EfficientNetB3 encoder using the Supervised Contrastive Loss. Two augmented views of each image were passed through a shared encoder, and positive/negative pair relationships were formed using class labels.
   - **Stage 2: Linear Classification Head** — The frozen encoder's representations were used to train a separate linear classifier on top using cross-entropy loss.

The SupCon implementation relied on an mlp-based projection head and used SGD with momentum and cosine annealing for training.

Implementation Details

All models were implemented in PyTorch and trained on GPU-accelerated hardware using cuda or Apple mps. We tracked experiments and hyperparameters using Weights & Biases. Batch sizes ranged from 16 to 64 due to different strategies in implementation. We saved the best models based on validation accuracy.

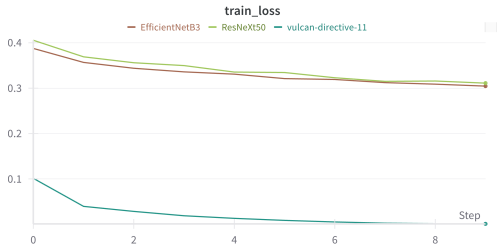
Model Selection for CNN-Based Experiments

In addition to the proposed CNN vs. Transformer comparison, we conducted a preliminary series of experiments focusing exclusively on CNN architectures. The objective was to quickly benchmark different convolutional backbones and head designs to assess their potential effectiveness in the task of AI-generated image detection. All experiments were performed during the early exploration phase, training the models for only 10 epochs with frozen backbone layers and training only the classification head.

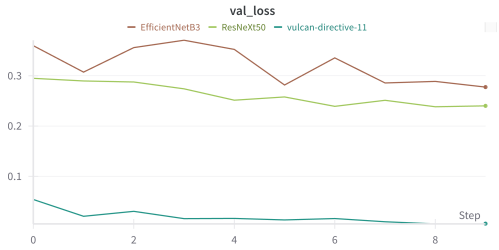
ResNet50 Baseline

For the initial benchmark, we employed ResNet50 as the backbone architecture. No data augmentation was applied in this setup to establish a clean baseline. The classification head consisted of a simple linear layer with no intermediate fully connected layers:

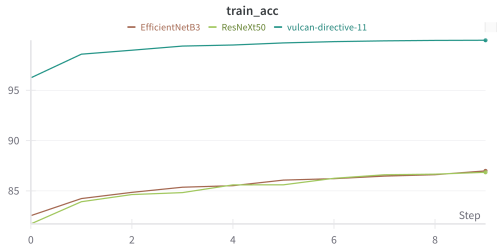
- **Head Architecture:** Single fully connected layer (Linear: in_features=2048, out_features=2).
- **Training Results:** Over 99% accuracy on both training and validation sets.
- **Public Leaderboard Score:** 0.30251.



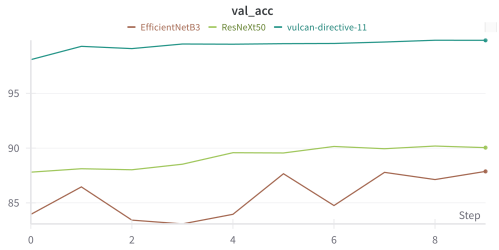
(a) Training Loss



(b) Validation Loss



(c) Training Accuracy



(d) Validation Accuracy

Figure 1: Comparison of CNN-based models across training and validation metrics. The subfigures display training loss, validation loss, training accuracy, and validation accuracy respectively for different model architectures.

Despite the impressive accuracy on the training and validation data, the low leaderboard score indicates significant overfitting and poor generalization to the test set.

ResNeXt50 with Augmentation and Custom Head

To mitigate overfitting and enhance feature extraction, we experimented with ResNeXt50. This variation included comprehensive data augmentation and a more sophisticated classification head with batch normalization and dropout layers.

- **Training Results:**
  - Training accuracy: 87%
  - Validation metrics:
    - Accuracy: 90.19%
    - F1 Score: 90.01%
- **Public Leaderboard Score:** 0.56871

Overall decrease in training and validation accuracy, but significant improvement in public leaderboard indicate that data a augmentation and a more complex head regularization contribute to better generalization.

EfficientNetB3 with Enhanced Augmentation

Building on the previous experiment, we tested EfficientNetB3 with even more aggressive data augmentation while keeping the same custom head design as ResNeXt50.

- **Training Results:**
  - Training accuracy: 87%
  - Validation metrics:
    - Accuracy: 87.87%
    - F1 Score: 88.61%
- **Public Leaderboard Score:** 0.68717

This model achieved the highest leaderboard score among the tested CNN architectures, indicating promising potential for further development.

Model Selection for Transformer-Based Experiments

Following CNN experiments, we explored Vision Transformers (ViTs) to leverage their global context capabilities, potentially advantageous in detecting subtle inconsistencies in AI-generated images.

Initial Experiments with ViT Variants

Initially, we explored smaller architectures such as ViT-Tiny/16 and ViT-Small/8, but these models did not yield satisfactory results, exhibiting poor generalization and significant overfitting. Subsequently, we tested ViT-B/16, given its established performance in image classification tasks:

- **Training Results:**
  - Training accuracy: 99.47%
  - Validation metrics:
    - Accuracy: 98.67%
    - F1 Score: 98.69%
- **Public Leaderboard Score:** 0.327

Despite high validation metrics, the relatively low leaderboard score indicated considerable overfitting and inadequate generalization on the test dataset.

Transition to ViT-B/32

Switching to ViT-B/32 effectively reduced model complexity and better captured global image structures:

- **Training Results:**
  - Training accuracy: 94.79%
  - Validation metrics:
    - Accuracy: 92.93%
    - F1 Score: 92.8%
- **Public Leaderboard Score:** 0.36537

ViT-B/32 with Augmentation and Regularization

Further enhancing ViT-B/32 with advanced data augmentation techniques and regularization methods significantly improved model robustness and generalization:

- **Training Results (with augmentation and regularization):**
  - Training accuracy: 84.842%
  - Validation metrics:
    - Accuracy: 89.68%
    - F1 Score: 89.69%
- **Public Leaderboard Score:** 0.471

Implementing these strategies proved effective in reducing overfitting and enhancing the performance of the model on unseen data.

Effectiveness of ViT-B/32

ViT-B/32 is better primarily due to its **improved generalization**, as the larger patches emphasize global image features, enabling the model to better generalize to unseen data. Moreover, the model benefits from **reduced complexity**, where fewer patches lead to lower computational complexity and a reduced parameter count, significantly mitigating overfitting risks.

Additionally, ViT-B/32 exhibited **enhanced stability** by responding effectively to regularization techniques such as dropout, label smoothing, and data augmentation. This responsiveness indicates that ViT-B/32 is particularly suitable for training enhancements aimed at improving robustness and stability on unseen images.

Further optimization of ViT-B/32 will be pursued through advanced regularization methods.

Model Selection for Supervised Contrastive Learning

Contrastive learning is a representation learning technique that trains a model to distinguish between similar and dissimilar data points. Rather than simply predicting a label, contrastive learning focuses on learning relationships between examples. The fundamental goal is to map similar instances closer together in the feature space, while pushing dissimilar instances farther apart. Mathematically, for two inputs x_i and x_j with corresponding feature embeddings $z_i = f(x_i)$ and $z_j = f(x_j)$ generated by an encoder $f(\cdot)$, the model minimizes a contrastive loss based on their similarity.

In our project, we apply **Supervised Contrastive Learning (SupCon)**, an extension of contrastive learning that utilizes class labels. Images belonging to the same class are

treated as positive pairs, while images from different classes are treated as negative pairs. The supervised contrastive loss we minimize is:

$$\mathcal{L}_{\text{SupCon}} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(z_i \cdot z_p / \tau)}{\sum_{a \in A(i)} \exp(z_i \cdot z_a / \tau)}$$

where I is the set of indices in the batch, $P(i)$ denotes positive examples for anchor i, $A(i)$ represents all positive and negative examples excluding i, and τ is a temperature hyperparameter that controls separation strength. The similarity between embeddings is computed using the dot product $z_i \cdot z_j$.

We initially trained a model using a **ResNet-50 encoder** with a multilayer perceptron (MLP) projection head under the SupCon framework. After freezing the encoder and fine-tuning a linear classifier, we achieved f1 score approximately **66.57%** on the Kaggle competition test set. While this already outperformed our previous baselines using a conventional convolutional neural network (CNN) and a vision transformer (ViT), we sought to further enhance performance.

Consequently, we retrained the encoder using a more powerful backbone: the **EfficientNet-B3** architecture. In this setup, each image underwent more diverse augmentation strategies, including random cropping, flipping, rotation, color jitter, and RandAugment policies. Using the `TwoCropTransform` technique, two augmented views of each image were generated and passed through the encoder. The supervised contrastive loss encouraged embeddings from the same class to cluster together and embeddings from different classes to be separated.

Following contrastive pretraining, we froze the EfficientNet-based encoder and trained a **linear classifier** on top of the frozen feature embeddings using the standard cross-entropy loss:

$$\mathcal{L}_{\text{CE}} = -\sum_{i} y_i \log(\hat{y}_i)$$

where y_i is the true label and $\hat{y}_i$ is the predicted probability for class i. For classifier training, we utilized stochastic gradient descent with momentum and weight decay for regularization. Additionally, a **Cosine Annealing Learning Rate Scheduler** was applied to progressively decay the learning rate across epochs, encouraging smoother optimization. This led to a significant improvement on our Kaggle testing set with **78.36% f1 score**.

Our project follows a two-stage contrastive learning pipeline:

1. First, we learn generalizable and transferable representations without explicit classification objectives through supervised contrastive learning.
2. Then, we fine-tune a lightweight linear classifier on top of the frozen encoder representations.

This method significantly boosts generalization, particularly for challenging tasks such as distinguishing real from AI-generated images.

Evaluation Setup

While we defer full performance analysis to the Evaluation section, we used the Kaggle competition's test set to submit final predictions. Model outputs were converted to class labels and saved in a submission-ready CSV file for leaderboard evaluation.

Datasets

We will use a curated dataset of AI-generated and Authentic Images obtained from Kaggle [12].

Authentic Images: Sourced from the Shutterstock platform, the dataset includes a diverse array of genuine images spanning various categories. Notably, approximately one-third of these images prominently feature human subjects, ensuring a balanced representation.

AI-Generated Images: Each authentic image is paired with an AI-generated counterpart. These synthetic images are created using a state-of-the-art generative model from DeepMedia, providing a rich set of examples for training and evaluation.

Exploratory Data Analysis

We performed exploratory data analysis to better understand the structure, distribution, and potential biases in the data set. Due to the large number of images in the dataset, when exploring image characteristics, we took samples of 500 images instead.

We first looked at the data to find missing values or outliers and see the distribution of AI vs Human generated images. We found the dataset to be clean with no missing or duplicate images. Additionally, there are 39,975 human-generated and 39,975 AI-generated images, which is as expected since each image has both an AI and human created example as seen below.

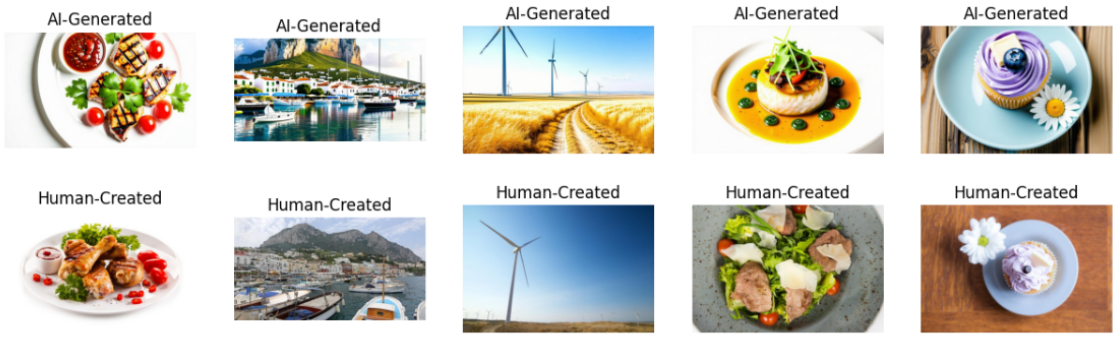


Figure 2: Example of Image Pairings

We then looked at the dimensions of the images to compare AI vs human-generated images. From our analysis. it shows that the dimensions of the images are the same.

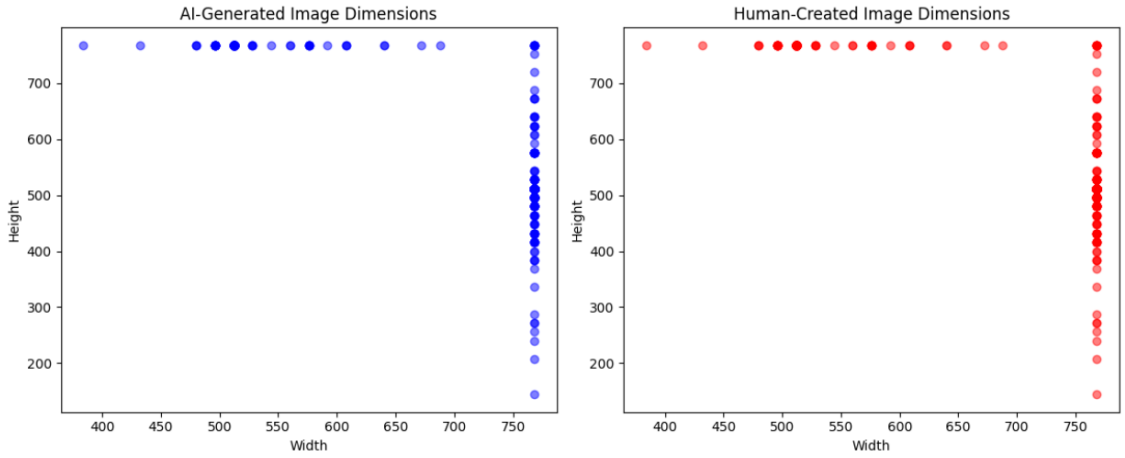


Figure 3: Comparing Image Dimensions Between AI and Human Created Images

We then looked at the color intensity of the images. Color intensity refers to the brightness or how saturated an image is. From this, we can see that on average, AI generated images have a higher standard deviation than human generated images. This indicates that AI generated images have more color variation whereas human generated images have more consistent colors overall.

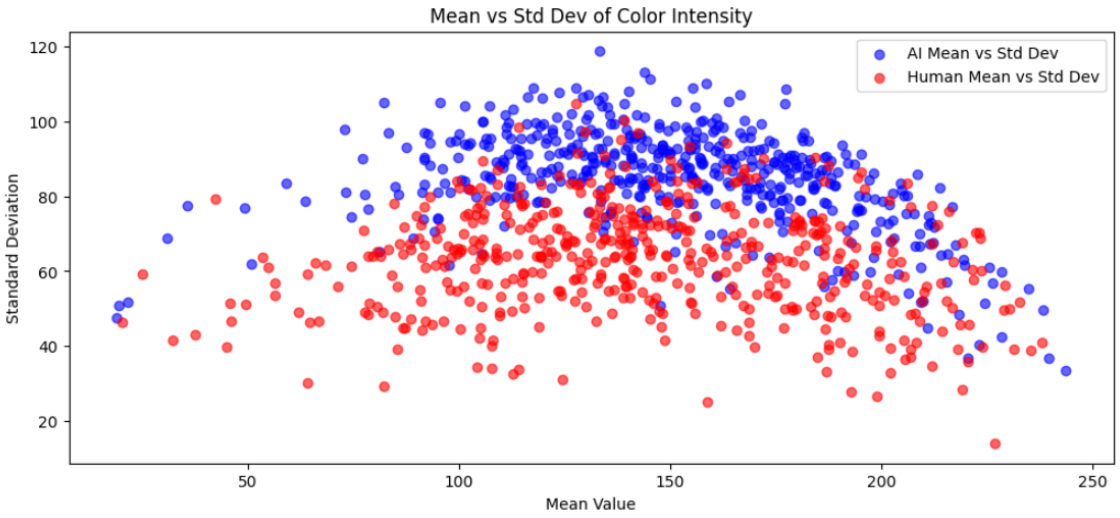


Figure 4: Comparing Color Intensity Between AI and Human Created Images

Lastly, we looked at contrast, dissimilarity, and homogeneity. These three measurements describe the smoothness of the texture of the images with higher values indicating less smooth transitions between pixels and a rougher texture. These graphs show that human-generated images seem to have lower contrast and dissimilarity scores indicating smoother transitions between pixels. The distribution of homogeneity is more uniform for human-created than ai-generated images indicating these images are either smoother or purposefully textured.

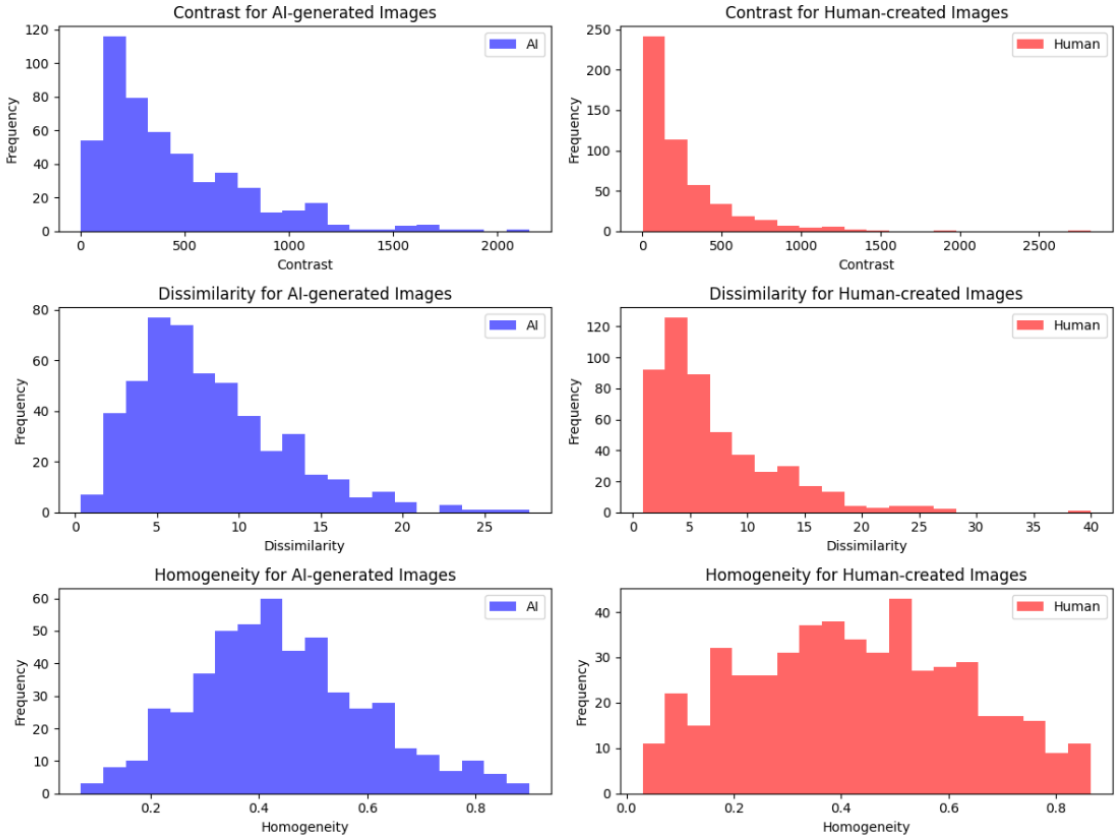


Figure 5: Comparing Image Characteristics Between AI and Human Created Images

Evaluation

Our model's performance will be assessed using the following metrics:

- **F1-score**: The harmonic mean of precision and recall, providing a single metric for performance evaluation:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Log Loss (Cross-Entropy Loss)**: Used when the model outputs probabilities rather than hard classifications, penalizing false classifications more heavily:

$$\text{Log Loss} = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i \log(p_i) + (1 - y_i)\log(1 - p_i)\right)$$

where y_i is the true label and p_i is the predicted probability.

After training three separate models, we then ran our model on a held out test set which was then submitted to the Kaggle leaderboard and measured on F1-Score. We first started off with a vision transformer based model using ViT-B/16 and then ViT-B/32 with augmentations and regularization. Using ViT-B/16 with limited augmentations, we achieved an F1-Score of 32.7% After adding augmentations, we increased the model's performance by 14.4 percentage points up to 47.1%. We also used a CNN based model starting with ResNet50 as a baseline. This model achieved an F1-score of 30.%. After augmentations and using ResNext50, the model significantly increased 26.6 percentage points up to 56.9%. After adding contrastive learning to this model, it increased an additional 9.7 percentage points up to 66.6%. We then found that EfficientNetB3 gave better results than ResNet50 and changed our model to use this architecture. Implementing EfficientNetB3 with augmentations gave us a F1-score of 68.7%. After adding contrastive learning to this model, it increased by 10.1 percentage points up to our highest performing model at 78.8%.

When evaluating our models, we found that the CNN based models performed better than the vision transformer based model. We also noticed two big changes led to great improvements in our F1-score. The first is using augmentations and regularizations which significantly improved all of our model. The second and more important change is the implementation of contrastive learning. After implementing contrastive learning, ResNet50 and EfficientNetB3 both had significant improvements at 9.7 and 10.1 percentage points respectively. This demonstrates that using contrastive learning is an effective tool to help classify ai versus human generated images and could achieve a relatively high F1-score of 78.8%. While there were no official baselines, the winning student score was 89% but used a completely different model. However, we were more focused on implementing a new technique rather than focused on achieving as high of an F1-score as possible but still performed relatively well.

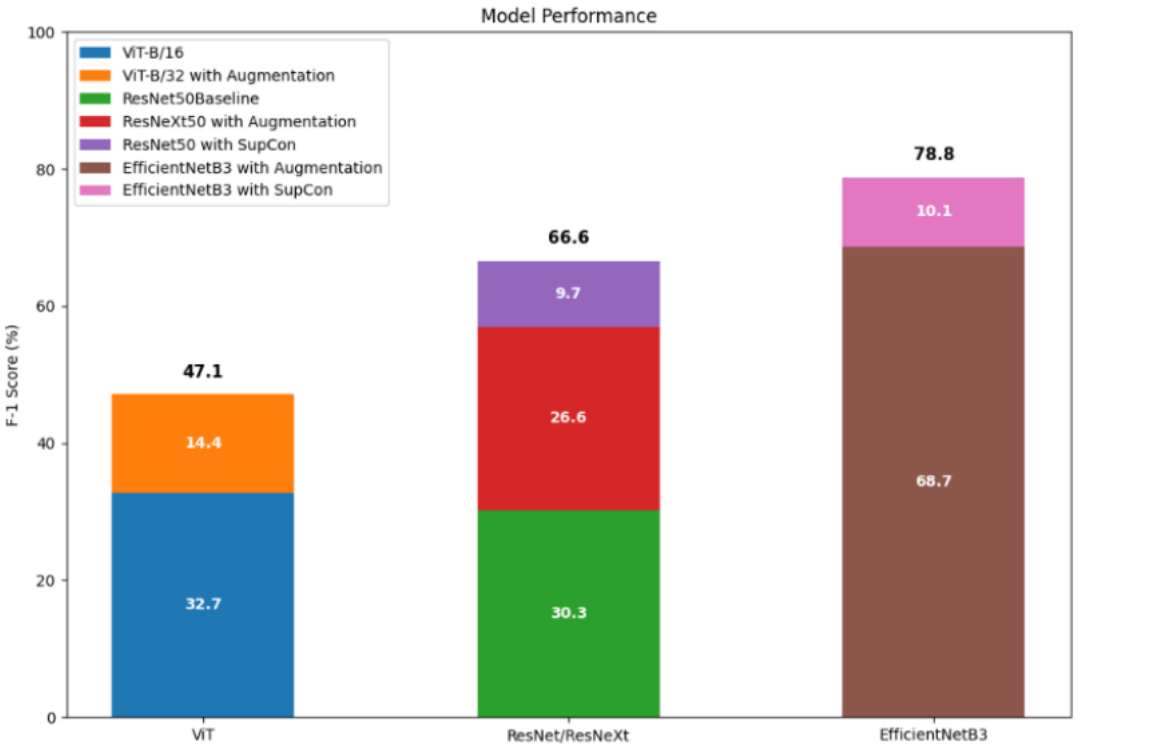


Figure 6: Comparing Model Performance F-1 Score (%)

Conclusion

We explored the problem of AI-generated image detection by comparing convolutional neural networks (CNNs), vision transformers (ViTs), and supervised contrastive learning (SupCon) approaches. Our experiments demonstrated that while CNNs and transformers can achieve high training and validation accuracy, their generalization to unseen test data remains a significant challenge. Initial models based purely on standard classification objectives often exhibited overfitting, as evidenced by the gap between validation performance and Kaggle competition leaderboard scores.

To address this, we incorporated supervised contrastive learning. By first learning generalizable feature representations without relying on explicit classification, and subsequently fine-tuning a lightweight linear classifier, we achieved improved separation between real and AI-generated images. Our initial SupCon training with a ResNet-50 encoder achieved approximately **66.57%** test f1 score, outperforming earlier CNN and transformer baselines. Building on this, switching to a more powerful EfficientNet-B3 encoder, combined with stronger data augmentation and a cosine annealing learning rate schedule, further

boosted our test f1 score to approximately **78.8%**.

Our findings support that contrastive learning enhances generalization in the context of AI image forensics. Future work could extend this study by incorporating semi-supervised contrastive learning, testing larger transformer backbones, or leveraging synthetic data augmentation to further close the gap in detecting increasingly realistic AI-generated content.

References

[1] Samah S. Baraheem and Tam V. Nguyen. Ai vs. ai: Can ai detect ai-generated images? *Journal of Imaging*, 9, 10 2023.

[2] Lorenzo Baraldi, Federico Cocchi, Marcella Cornia, Lorenzo Baraldi, Alessandro Nicolosi, and Rita Cucchiara. Contrasting deepfakes diffusion via contrastive learning and global-local similarities. 7 2024.

[3] Jordan J. Bird and Ahmad Lotfi. Cifake: Image classification and explainable identification of ai-generated synthetic images. *IEEE Access*, 12:15642–15650, 2024.

[4] Druthik Sai Chinta, Sindhusha Kamineni, Ratna Pujitha Chatragadda, and Sujatha Kamepalli. Analyzing image classification on ai-generated art vs human created art using deep learning models. In *2024 3rd International Conference on Electrical, Electronics, Information and Communication Technologies, ICEEICT 2024*. Institute of Electrical and Electronics Engineers Inc., 2024.

[5] Stefano Filipazzi, Christopher D. Hacon, and Roberto Svaldi. Boundedness of elliptic calabi-yau threefolds. 12 2021.

[6] FlyPix AI. Image recognition models: Cnns and the future of ai vision, 2025. Accessed: 2025-02-27.

[7] Md Zahid Hossain, Farhad Uz Zaman, and Md Rakibul Islam. Advancing ai-generated image detection: Enhanced accuracy through cnn and vision transformer models with explainable ai insights. In *2023 26th International Conference on Computer and Information Technology, ICCIT 2023*. Institute of Electrical and Electronics Engineers Inc., 2023.

[8] Fernando Martin-Rodriguez, Rocio Garcia-Mojon, and Monica Fernandez-Barciela. Detection of ai-created images using pixel-wise feature extraction and convolutional neural networks. *Sensors*, 23, 11 2023.

[9] José Maurício, Inês Domingues, and Jorge Bernardino. Comparing vision transformers and convolutional neural networks for image classification: A literature review, 5 2023.

[10] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. 12 2017.

[11] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. 1 2019.

[12] Alessandra Sala, Harshika, Manuela Jeyaraj, Margarita Pitsiani, Niamh Belton, and Toma Ijatomi. Detect ai vs. human-generated images. `https://kaggle.com/competitions/detect-ai-vs-human-generated-images`, 2025.

[13] Connor Shorten and Taghi M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6, 12 2019.

[14] Ke Sun, Taiping Yao, Shen Chen, Shouhong Ding, Jilin L, and Rongrong Ji. Dual contrastive learning for general face forgery detection. 12 2021.

[15] Vrizlynn L L Thing. Deepfake detection with deep learning: Convolutional neural networks versus transformers. Technical report.

[16] Ying Xu, Kiran Raja, and Marius Pedersen. Supervised contrastive learning for generalizable and explainable deepfakes detection. Technical report.

[17] Jun Shuai Zheng, Yi Chao Zhou, Xi Yuan Hu, and Zhen Min Tang. Deepfake detection with combined unsupervised-supervised contrastive learning. In *Proceedings - International Conference on Image Processing, ICIP*, pages 787–793. IEEE Computer Society, 2024.

[18] Mingjian Zhu, Hanting Chen, Qiangyu Yan, Xudong Huang, Guanyu Lin, Wei Li, Zhijun Tu, Hailin Hu, Jie Hu, and Yunhe Wang. Genimage: A million-scale benchmark for detecting ai-generated image. Technical report.