AI-Generated Synthetic Images Differentiation

Efe Eboreime
George R. Brown
School of Engineering
Rice University
Houston, Texas 77005
Email: ee37@rice.edu

Steve Hudgson
George R. Brown
School of Engineering
Rice University
Houston, Texas 77005
Email: sah27@rice.edu

Weixiao Liao
George R. Brown
School of Engineering
Rice University
Houston, Texas 77005
Email: wl80@rice.edu

Minh Nguyen
George R. Brown
School of Engineering
Rice University
Houston, Texas 77005
Email: mn78@rice.edu

Abstract—**This paper addresses the critical challenge of discerning authentic images from AI-generated ones, crucial in legal, societal, and cybersecurity contexts. Utilizing the CIFAKE dataset, a custom Convolutional Neural Network (CNN) and ResNet-50 model are implemented and rigorously compared. Both models showcase robust capabilities, with ResNet-50 achieving marginally superior accuracy. Precision, recall, and F1 score metrics offer nuanced insights, emphasizing the significance of specific error types in model evaluation. The findings underscore the potential of these models in navigating the intricate landscape of AI-generated imagery, with implications for bolstering cybersecurity and ensuring data trustworthiness. Future research avenues include exploring the integration of Generative Adversarial Networks (GANs) alongside the existing CNN architecture for potential performance enhancements.**

1. Introduction

1.1. Background

The ability to distinguish between authentic and AI-generated images is of utmost importance, marking a critical frontier in the realms of technological ethics and security. This is particularly significant due to the pervasive risks associated with the widespread use of AI-generated images, spanning societal, informational, and cybersecurity dimensions. Of specific concern is the potential creation of synthetic evidence for criminal activities, with cutting-edge models such as Stable Diffusion Models (SDMs) intricately fabricating deceptive visuals that introduce an element of doubt into legal proceedings [1].

The pervasive integration of machine-generated images into the landscape of fake news exacerbates the challenge of determining the veracity of information, thereby posing a formidable threat to public opinion [2]. Furthermore, in the realm of cybersecurity, the deployment of highly realistic synthetic personas in false acceptance attacks poses a direct threat to digital authentication processes, as evidenced by the work of Khosravy et al. [3]. Research also sheds light on the adaptability of generative models in overcoming signature verification systems, as highlighted by Bird et al. [4]. These multifaceted risks underscore the urgent need for the development and implementation of robust strategies to navigate the challenges posed by the ever-evolving landscape of AI-generated imagery.

In the landscape of image generation, Latent Diffusion Models (LDMs) present a novel approach, incorporating attention mechanisms and a U-Net structure. Models such as DALL-E, Imagen, SDM, and others contribute significantly to ongoing discussions concerning image quality, thereby intensifying debates surrounding professional, social, ethical, and legal considerations. As LDMs continue to emerge, their unique features contribute to the dynamic discourse surrounding the impact of generative models across diverse domains.

Within the domain of visual detection, various techniques are being developed to identify synthetic elements. Optical flow methods, as elucidated in the work of Amerini et al. [5], achieve an 81.61% accuracy in detecting synthetic human faces within the FaceForensics dataset. Another study by Wang et al. [6] proposes an efficient system utilizing EfficientNets and Vision Transformers, capable of detecting forged images with an impressive F1 score of 0.88 and an AUC of 0.95. Additionally, DE-FAKE [7] brings attention to digital fingerprints present in images generated by latent diffusion approaches, suggesting their inherently synthetic nature. Given the rapid pace of field development over the nearly five years since the initiation of this study, it becomes imperative to stay abreast of evolving methodologies and challenges in the realm of synthetic image detection.

1.2. Data

For our experimentation, we utilized the CIFAKE [8] (some examples of images shown in Figure 1) which comprises a total of 120,000 images, equally divided into two classes: REAL and FAKE. The REAL class includes 60,000 images sourced from the widely recognized CIFAR-10 dataset, originally developed by Krizhevsky, Nair, and Hinton. This set offers a diverse range of natural images, serving as a benchmark for real-world visual data.

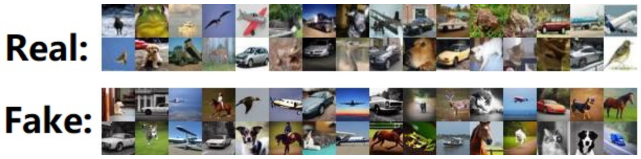

Figure 1. Examples of images from the CIFAKE dataset.

Conversely, the FAKE class contains 60,000 synthetically-generated images, created using the Stable Diffusion version 1.4 model, mirroring the CIFAR-10 equivalent but through AI-generated means.

The dataset is partitioned into training and testing subsets. There are 100,000 for training, with an equal split of 50,000 images per class. The testing subset contains 20,000 images and is similarly divided into 10,000 images per class. The table below summarizes the CIFAKE dataset.

	# of REAL images	# of FAKE images
Train	50,000	50,000
Test	10,000	10,000

2. Experimentation

2.1. Data Augmentation and Transformation

We employ the `torchvision`'s `transform` technique to apply a series of transformations to the dataset. Our first transformation is resizing the images into a uniform dimension of 32x32 pixels using `transforms.Resize((32, 32))`. Even though the images are already set to this size in the CIFAKE dataset, the resizing step will ensure future-proof flexibility with new images of varying sizes. The next necessary transformation is to convert images to PyTorch sensors. Such transformations are critical in ensuring consistency in image size and format, which can heavily impact our model's learning efficiency and performance.

2.2. Methods

In our study, we implemented two distinct methods for distinguishing between AI-generated synthetic images and real images. The first method involved developing a custom Convolutional Neural Network (CNN) based on the suggested framework outlined in Bird and Lotfi's research on the CIFAKE dataset [1]. we utilized the ResNet-50 model, a well-established neural network renowned for its deep architecture and efficiency in image classification tasks. These two approaches were then compared and contrasted in terms of their performance metrics, such as accuracy, precision, recall, and F-1 score to evaluate their effectiveness in differentiating synthetic and real images. This comparison allowed us to assess the strengths and limitations of our custom-designed models versus established architectures in the field of image classification.

2.2.1. Custom CNN. Our CNN model is structured to leverage the nuances of image data efficiently, incorporating a sequence of layers, each with a distinct role in the image classification task. The structure of our CNN is demonstrated through Figure 2 below.

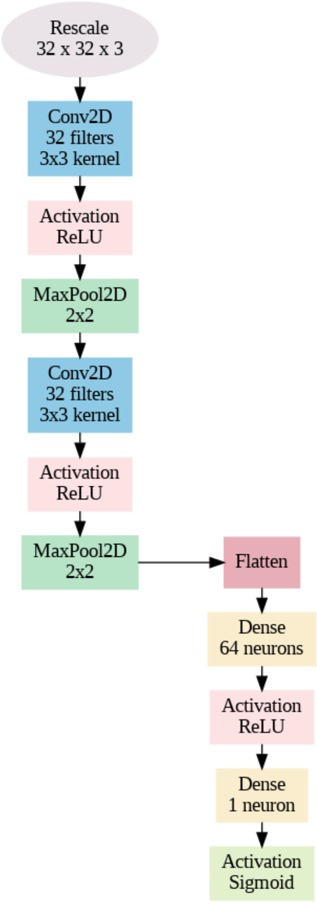

Figure 2. Custom CNN structure.

- **Convolutional Layers (Conv2d):** The model begins with convolutional layers (Conv2d), employing 32 filters of size 3x3. These layers are instrumental in scanning the 32x32 pixel input image, preserving spatial dimensions due to padding of 1, while enhancing depth for feature detection.
- **ReLU Activation Function:** Following the convolutional layers, the ReLU (Rectified Linear Unit) function introduces non-linearity. This is crucial for learning complex patterns, as it converts negative values in the feature map to zero, aiding in complex input-output mapping.
- **Max Pooling (MaxPool2d):** Max Pooling layers follow each ReLU activation. These layers reduce spatial dimensions by half, first to 16x16, then to 8x8, while maintaining a depth of 32. This step is essential for managing data representation and mitigating overfitting.

- **Flattening and Fully Connected Layers:** The network employs a flattening operation to transform 2D feature maps (8x8x32) into a 1D vector, enabling input into fully connected layers. The first of these layers reduces the 2048 elements to 64 features, leading to the final output layer.
- **Output Layer and Sigmoid Activation:** The final layer of the CNN uses a Sigmoid function to provide a probability score, indicating the likelihood of the image being real or synthetic.
- **Optimization and Training:** The model employs the Adam optimizer for its training process. This optimizer is known for its adaptability in adjusting learning rates, ensuring efficient training of the network.

The architecture of this CNN is grounded in the principles of feature extraction and dimensionality reduction. The convolutional layers detect intricate patterns within images, while the pooling layers reduce computational complexity and prevent overfitting. The fully connected layers then interpret these patterns to classify the images. This architecture is particularly novel for its balance of depth and computational efficiency, making it an optimal choice for distinguishing between AI-generated and real images.

Throughout the development process, we experimented with varying the number and size of filters and layers. Some configurations led to overfitting or underfitting, underscoring the importance of architectural balance. These trials, although not successful, provided valuable insights into the model's sensitivity and the critical role of each layer in image classification.

2.2.2. ResNet-50. ResNet-50, a variant of the ResNet architecture with 50 layers, is a deep convolutional network known for its efficient handling of vanishing gradients in deep networks. This is achieved through the use of skip connections or shortcuts that jump over some layers, allowing for the training of deeper networks without loss of performance. In your model adaptation, the final fully connected layer of ResNet-50 has been modified to output a single feature, using a sigmoid activation function for binary classification, which is ideal for tasks like distinguishing AI-generated images from real ones. This modification, combined with the use of the Adam optimizer, highlights an emphasis on adaptability and efficient training tailored to your specific classification task. The simplified architecture of our ResNet-50 is demonstrated through Figure 3 below.

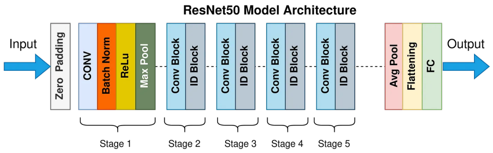

Figure 3. ResNet-50 architecture [9].

2.3. Metrics

In assessing the performance of classification models, we utilize three key metrics: Precision, Recall, and F1-Score. Precision measures the accuracy of positive predictions, Recall assesses the coverage of actual positive cases, and F1-Score provides a balance between Precision and Recall, offering a single measure of test accuracy.

- **Precision**: This is the ratio of true positive predictions to the total number of positive predictions made. It is a measure of the accuracy of the positive predictions, accounting for false positives. The formula for precision is given by:

$$\text{Precision} = \frac{\text{True positives}}{\text{True positives} + \text{False positives}}$$

- **Recall**: This is the ratio of true positive predictions to the actual number of positive cases. It measures how many actual positive cases were correctly identified and is crucial for analyzing false negatives. The formula for recall is:

$$\text{Recall} = \frac{\text{True positives}}{\text{True positives} + \text{False negatives}}$$

- **F1-Score**: This score is a harmonic mean of precision and recall, providing a balance between them. It is especially important in situations like fraud detection, where a false negative could have serious implications. The F1-score is calculated as:

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

3. Observation & Results

The analysis of the image classification task was conducted with the primary objective of evaluating the 'task-learning ability' inherent in the employed models. Our findings are derived from a comprehensive examination of a substantial training set, comprising 100,000 images, and a separate test set, which included 20,000 images.

Examining the performance of our custom CNN implementation, we observed a consistent reduction in loss during both training and testing, accompanied by a significant rise in accuracy, peaking at around 95% (see Figure 4) during testing. The evaluation of the pre-trained ResNet-50 model showcased similar trends, with a consistent decrease in loss during training and testing and a distinct increase in accuracy, reaching an impressive peak of approximately 97% (see Figure 5) during testing. In both cases, these models demonstrated robust capabilities in effectively distinguishing between AI-generated (fake) images and real ones.

While Accuracy remains a widely used metric in evaluating models, especially in classification tasks, its limitation lies in its inability to discern specific types of errors, such as false positives and false negatives. This becomes crucial because, in various applications, certain types of errors

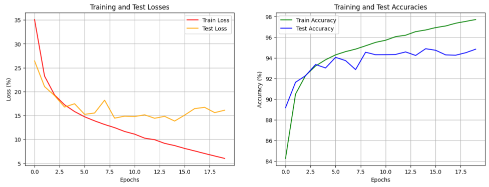

Figure 4. Loss and Accuracy (Custom CNN).

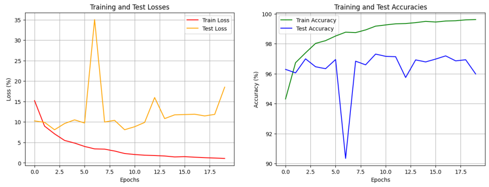

Figure 5. Loss and Accuracy ResNet-50.

may be more significant than others. Metrics like precision, recall, and F1 score may be more informative.

To expand our evaluation, we augment our findings through a thorough analysis incorporating two essential components: confusion matrices (Figure 6 & 7) and precision/recall/F1 metrics (Table 1). These two elements collectively paint a more nuanced picture, revealing not only the overall accuracy but also the model's precision in correctly identifying positive predictions and its recall in capturing all relevant predictions. This approach allows for a much deeper understanding of both models' efficacy, offering insights into their ability to differentiate between the two classes.

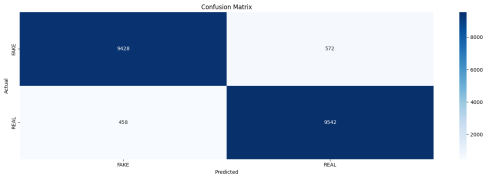

Figure 6. Confusion Matrix for Custom CNN.

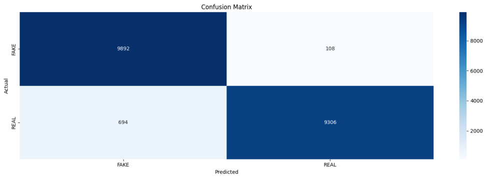

Figure 7. Confusion Matrix for Resnet-50.

Comparing the results of both models, it is evident that our custom implementation struggles a lot more in identifying real images by misclassifying more fake images as real than in the pre-trained Resnet-50 model. As previously mentioned, this can have varying impacts depending on the gravity of these types of errors in that application domain. The novelty of our approach here is our ambition to outperform already established models that succeed in a wide variety of image classification tasks. Arguably, more finetuning could be done to achieve this, however, given the limited timeline coming this close is proof of our ability to potentially achieve this.

Model	Precision	Recall	F1-Score
Custom CNN	0.9434	0.9542	0.9488
Resnet-50	0.9885	0.9306	0.9587

TABLE 1. Performance Metrics

4. Conclusion

This study has proposed the possibility of differentiating synthetically AI-generated images from real ones. Utilizing the CIFAKE dataset, we employed a Convolutional Neural Network (CNN) architecture to classify images into two categories: authentic and fabricated. The accuracy of the classification was approximately 94%, utilizing a total of 120,000 images, with 20,000 allocated for testing and 100,000 dedicated for training. Notably, in comparison, the ResNet-50 model demonstrated a slightly higher accuracy, which could be attributed to its deeper architecture and sophisticated feature extraction capabilities.

Future work for this project could involve implementing Generative Adversarial Networks (GANs) alongside our current CNN architecture in order to see if there will be a difference in performance and scalability.

This type of work will help different areas in computing, such as cyber security and data management, by ensuring data authenticity and trustworthiness when dealing with different datasets that one is not sure what origin they are from.

5. Code Availability

Our code is available on GitHub at https://github.com/minhnguyen1406/AI-Generated-Synthetic-Images-Differentiation.

6. Group Members Roles

In our collaborative research, Weixiao, serving as the team captain, delved into the existing literature, conducting a thorough investigation to provide a solid foundation for the research. Minh takes charge of coding and network construction, leveraging technical proficiency to implement the envisioned models. Efe assumes the crucial role of scrutinizing and analyzing data outputs, ensuring a cohesive synthesis of findings. Steve, responsible for drawing conclusions and managing references, synthesizes overarching insights from our findings. It's noteworthy that the collaborative nature of our team extended to all members participating in both the coding and writing processes, with each member contributing to different sections of the paper and playing a role in the creation of our project poster.

References

[1] J. J. Bird and A. Lotfi, ”CIFAKE: Image Classification and Explainable Identification of AI-Generated Synthetic Images,” *arXiv preprint arXiv:2303.14126*, 2023. [Online]. Available: https://ar5iv.org/abs/2303.14126

[2] Singh, B., and D. K. Sharma. ”Predicting Image Credibility in Fake News over Social Media Using a Multi-Modal Approach.” *Neural Computing and Applications*, vol. 34, no. 24, 2022, pp. 21503-21517.

[3] Khosravy, M., et al. ”Model Inversion Attack: Analysis under Gray-Box Scenario on Deep Learning-Based Face Recognition System.” *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 15, no. 3, 2021, pp. 1100-1118.

[4] Bird, J. J., A. Naser, and A. Lotfi. ”Writer-Independent Signature Verification; Evaluation of Robotic and Generative Adversarial Attacks.” *Information Sciences*, vol. 633, 2023, pp. 170-181.

[5] Amerini, I., et al. ”Deepfake Video Detection through Optical Flow-Based CNN.” *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019.

[6] Wang, J., et al. ”M2TR: Multi-Modal Multi-Scale Transformers for Deepfake Detection.” *Proceedings of the 2022 International Conference on Multimedia Retrieval*, 2022, pp. 615-623.

[7] Sha, Z., et al. ”De-Fake: Detection and Attribution of Fake Images Generated by Text-to-Image Diffusion Models.” *arXiv preprint arXiv:2210.06998*, 2022.

[8] Bird, J. J. ”CIFAKE: Real and AI-Generated Synthetic Images.” *Kaggle*, 2023, www.kaggle.com/datasets/birdy654/cifake-real-and-ai-generated-synthetic-images.

[9] Wikipedia contributors, ”ResNet-50 architecture,” *Wikimedia Commons, the free media repository*. [Online]. Available: https://commons.wikimedia.org/wiki/File:ResNet50.png. [Accessed: insert-date-here].