IE 7615 Project Report
Detection of AI-Generated Image

Shuhan Li(li.shuhan1@northeastern.edu)

Weiqi Sun(sun.weiq@northeastern.edu)

Abstract

As AI technology continues to advance, its capabilities become increasingly refined, and the films or artworks created by AI are becoming indistinguishable to the naked eye. In this project, we are dedicated to developing a CNN-based model to effectively help people distinguish between AI-generated images and real images. To further validate the reliability of this CNN model, we utilize ResNet, a pre-trained model with exceptional performance in image classification, for a horizontal comparison. Additionally, we employ data augmentation techniques to enhance the model's generalizability and utilize Bayesian hyperparameter optimization strategies for the final parameter tuning. Ultimately, this model, with its robust and reliable accuracy, can assist users in effectively distinguishing the authenticity of images when necessary.

Catalog

Introduction

At the end of 2022, the launch of ChatGPT significantly heightened global interest in AI technologies. Since then, major tech companies have developed powerful models to meet the needs of various domains, greatly facilitating everyday life and, to some extent, radically transforming it. With just simple commands, once unimaginable results can now be achieved instantaneously. However, this progress has not come without its challenges, particularly for artists who invest their time and passion into creating photographic works or paintings, embedding their love for art into their creations and using them as a medium to convey their thoughts to the world. This dedication is not only admirable but also deserving of protection. Therefore, this project is dedicated to developing a deep learning model based on Convolutional Neural Networks (CNN) to effectively distinguish between AI-generated images and real photographs, thereby supporting content verification and copyright protection applications.

The deep convolutional neural network (CNN) is the state-of-the-art solution for large-scale visual recognition [1]. Naturally, it was our first choice, as CNNs, by simulating the mechanism of the human visual system, are capable of autonomously learning and recognizing hierarchical features in images, thus achieving high precision in image classification and recognition tasks. To assess the performance of our developed model and ensure its applicability, we have incorporated the Residual Network (ResNet) as a benchmark model. ResNet addresses the issue of performance degradation during the training of deep networks through the concept of residual learning, and its superior classification performance in various image processing benchmarks was a key reason for its selection.

Additionally, to enhance the model's adaptability to different environments and conditions, we have employed data augmentation techniques. By performing operations such as rotation, scaling, and cropping on the training dataset, we increased the diversity and quantity of the data. Concurrently, Bayesian hyperparameter optimization methods were utilized to fine-tune the CNN model. This strategy, based on probabilistic models, optimizes parameters iteratively to achieve optimal learning outcomes and generalization capabilities.

Even the most powerful models require appropriate data support. Our project utilizes a dataset comprising 60,000 pairs of various content images and AI-generated images. Of these, 45,000 pairs are used for training the model, 5,000 pairs for validation, and 10,000 pairs for final testing.

By integrating advanced technologies such as CNN architecture, data augmentation, and Bayesian optimization, the model developed in this project not only provides high accuracy in determining the authenticity of images but also boasts considerable reliability and practicality. This will offer robust technical support for ensuring the authenticity and security of digital content.

Background

CNN was initially known as LeNet, which described the foundational components of CNNs and can be considered the origin of CNN technology. Due to the lack of hardware, especially GPUs (Graphics Processing Units), LeNet-5 was not well-known. Consequently, from 1990 to 2000, there was minimal research on CNNs. The success of AlexNet in 2012 opened the door for computer vision applications, leading to the development of many different forms of CNNs, such as the R-CNN series. Although today's CNN models are quite different from LeNet, they all evolved from it.[2]

In the rapidly evolving field of digital image analysis, the current state of the art in detecting AI-generated images showcases significant advancements in machine learning and computational techniques. As of now, deep learning models, particularly Convolutional Neural Networks (CNNs) stand at the forefront of this technology, offering unprecedented accuracy in distinguishing between AI-generated and authentic human or digitally created images. Based on the research of [3], *they proposed a model that implements explainable AI via Gradient Class Activation Mapping to explore which features within the images are useful for classification to reach a relatively high accuracy at 92.98%.*

Approach

CNN

In our project, we deploy a Convolutional Neural Network (CNN) using a sequential architecture to differentiate real images from AI-generated ones. The CNN is meticulously designed to capture the hierarchical patterns in the images, from simple edges to complex structures through multiple layers. We use 3 convolutional layers with a 3x3 filer to extract 32,64,128 feature layers and each layer is followed by a 2x2 resampling layer using MaxPooling. The multi-dimensional output of previous layers is transformed into a one-

dimensional array through a flattening process. To combat overfitting, a dropout layer is applied after the dense layer, randomly dropping out 50% of the nodes in the training process, which encourages the network to learn redundant representations and increases its generalizability.

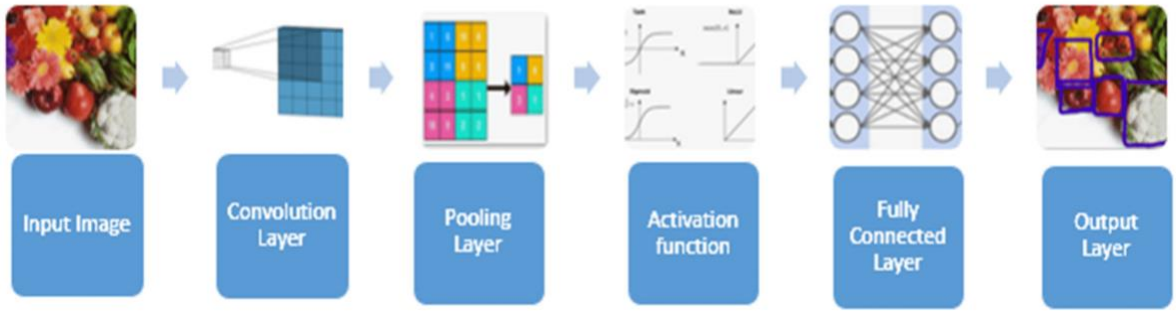


Figure1: Model Architecture For CNN

ResNet

Our second method is the ResNet50 architecture, a residual learning framework renowned for enabling the training of exceedingly deep neural networks as our base model. The "50" in ResNet50 denotes the fifty layers that make up this network, consisting of a series of residual blocks that utilize skip connections to jump over certain layers. These connections address the vanishing gradient problem by allowing the gradient to be directly backpropagated to earlier layers. Mathematically, the operation performed by a residual block can be represented as F(x)+x where x is the input to the block, and F(x) is the output of the residual mapping (the stacked non-linear layers). The sum F(x)+x is then passed through a ReLU activation function before proceeding to the next block.

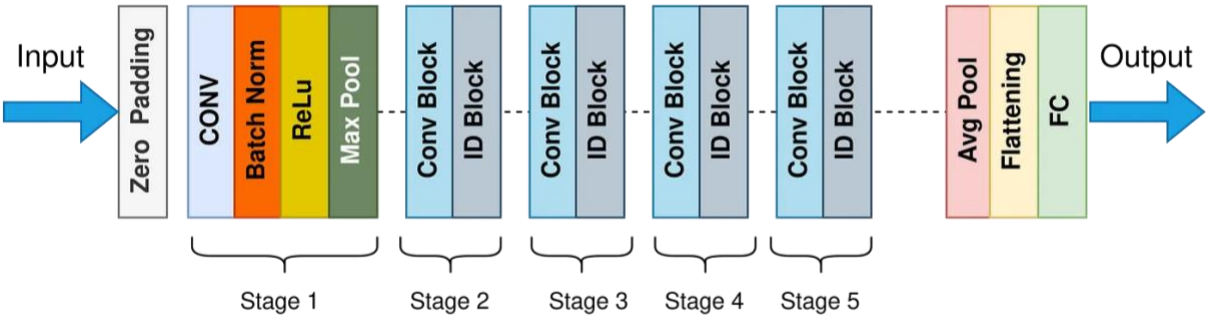


Figure 2：ResNet50 Model Architecture

Data Augmentation

In our comprehensive approach to developing a robust classifier, we have incorporated a strategic data augmentation phase. Data augmentation is a critical step in our methodology, aimed at bolstering the learning efficacy of our model and enhancing its ability to generalize from the training data to unseen images. We used geometric transformation to enhance model variability and complexity, resulting in a more rigorous and comprehensive learning experience.

Enhancing data through scaling, cropping, and horizontally flipping images can improve the accuracy and stability of models, and also serves as an ideal solution when there is a lack of data. Although this can often lead to negative effects such as increased testing time, as long as the size of the augmented data is planned properly, the improvements it brings are significant and effective. [4]

Bayes Hyperparameter Tuning

In our pursuit to fine-tune the CNN for optimal performance, we have utilized Bayesian optimization for hyperparameter tuning. This probabilistic model-based approach offers a more intelligent search for the hyperparameter space compared to random or grid search methods. The optimization algorithm iteratively selects the next set of hyperparameters by analyzing the previous results, effectively narrowing down the search space to the most promising regions. Once the trials are complete, we extract the best hyperparameter set that the model suggests yields the highest classification accuracy.

$$\boldsymbol{P}(\boldsymbol{Z}|\boldsymbol{Y}) = (\boldsymbol{P}(\boldsymbol{Y}|\boldsymbol{Z})\,\boldsymbol{P}(\boldsymbol{Z}))/\boldsymbol{P}(\boldsymbol{Z})$$

Figure 3: Bayesian optimization

The optimization procedure is based on Baye's Theorem in the equation which states for a model Z and observation Y where P(Z|Y) is the posterior probability of Z given Y, P(Y|Z) is the likelihood of Y given Z, P(Z) is the prior probability of Z, and P(Z) is the marginal probability of Z. Bayesian optimization is used to find the minimum of a function, f(y), on a bounded set, Y. [5]

Early Stopping

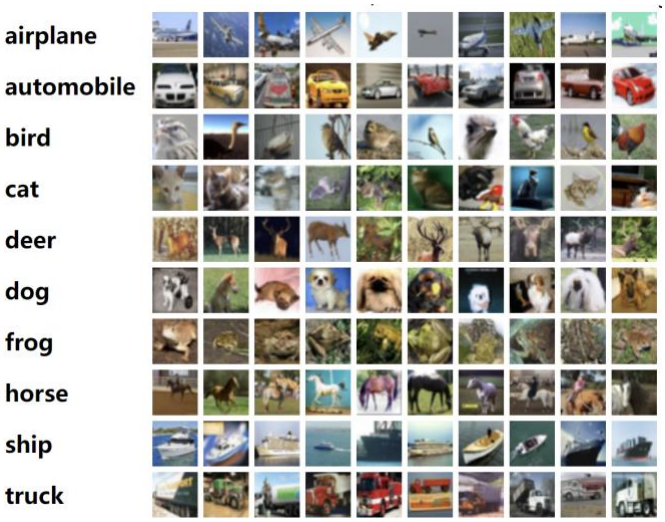
Understanding that the number of training epochs can significantly influence a model's ability to generalize, we plan to establish an optimal balance. Too few epochs could

lead to underfitting, while too many can lead to overfitting. By monitoring validation loss and employing early stopping, we can halt training as soon as the model's performance on the validation set begins to degrade.

Results

Dataset

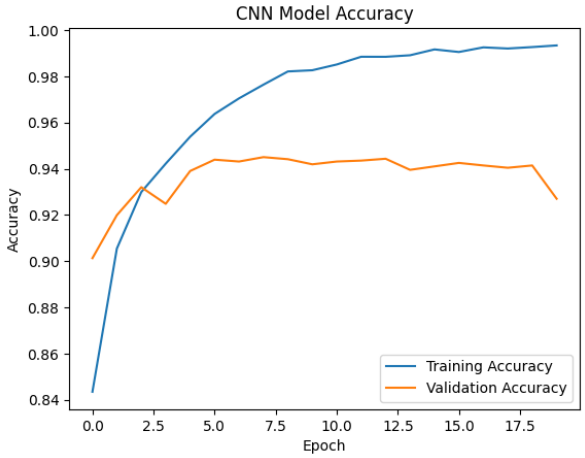
The dataset employed in our study is separated into two parts: The first part incorporates the CIFAR-10 dataset, which constitutes the authentic images. The second segment encompasses synthetic images, produced using Stable Diffusion version 1.4, to create the artificial or 'fake' image category. The training subset contains a total of 100,000 images, evenly distributed with 50,000 images for each class, while the testing subset is comprised of 20,000 images, with 10,000 images per class for evaluation purposes.

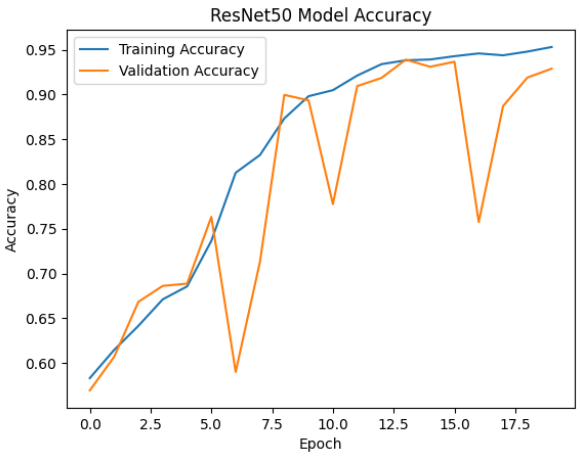


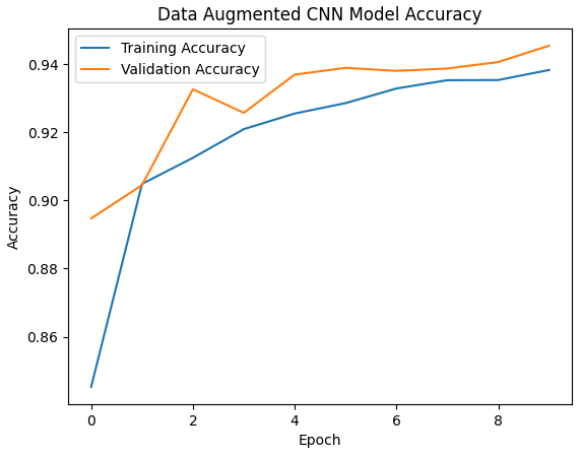
Real and AI-Generated Synthetic Images

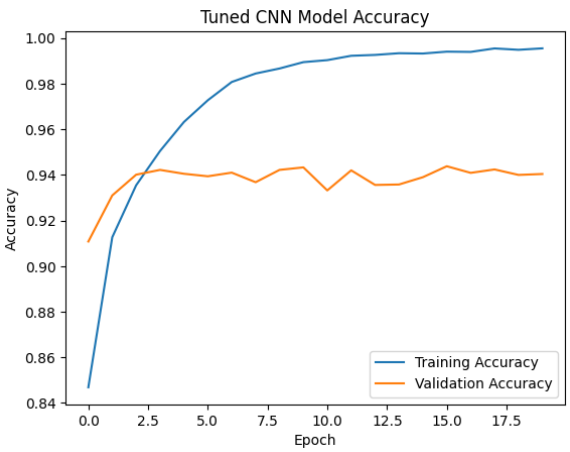
Model Accuracy

To maintain the consistency of our training process, we used the same 20 training epochs for each model and the same optimizer, “Adam”, to perform our classification task.









Model	Test Accuracy
CNN	0.9258
ResNet 50	0.9235
CNN with Data Augmentation	0.9450
CNN with Hyperparameter Tuning	0.9401

The first plot shows the training and validation accuracy of the CNN model. The training accuracy increases consistently with each epoch and appears to plateau around 98%, indicating strong learning on the training set. The validation accuracy, while lower than the training accuracy, also shows an upward trend, suggesting that the model is generalizing well. There's a noticeable gap between training and validation accuracy, which could be indicative of some overfitting.

The second graph, the ResNet50 model, shows more volatility, particularly in the validation accuracy. The training accuracy exhibits an upward trend but with fluctuations, reaching roughly 95% by the final epoch. The validation accuracy, however, has a significant variance, peaking and dipping dramatically, which may suggest that the model is not stable and could benefit from further tuning or regularization.

In the third graph, both the training and validation accuracies demonstrate a consistent increase. The validation accuracy is the highest one at 94.50%. The closer convergence of the two lines as compared to the first plot indicates that data augmentation has helped the model generalize better, reducing the gap between training and validation accuracy.

The last hyper-parameter tuned model, demonstrates an improvement in test accuracy by approximately 2% over the baseline CNN model. Moreover, this refined model achieves its highest validation accuracy at an earlier stage, specifically by the third epoch, in contrast to the initial CNN model's longer time to reach peak performance.

In our future research endeavors, we aim to delve deeper into refining the performance and generalizability of our models, with a particular focus on the following strategic areas: regularization techniques and advanced hyperparameter tuning.

Discussion

In our initial results, we were very satisfied with the test accuracy of 92.58% achieved by our initial CNN model. This result confirmed that CNN models indeed possess strong performance and accuracy in the field of visual recognition and classification. After undergoing data augmentation and Bayesian hyperparameter optimization, the accuracy of the CNN model improved by approximately 2%. Although this improvement might not seem significant, considering that the primary goal of data augmentation is to enhance the model's generalizability and that of Bayesian optimization is to enhance model performance, such outcomes are completely acceptable.

In this project, we aimed to compare the performance of our CNN model against the powerful pre-trained ResNet50 model. Using 45,000 data pairs and 20 epochs, we achieved an accuracy nearly identical to that of the pre-trained ResNet model and demonstrated stability superior to that of ResNet. We conducted a deeper investigation and discussion of these results, leading to the following conclusions:

1. **Network Depth Doesn't Mean Everything**: ResNet50 has a deeper network structure than our CNN model. Typically, a deeper network can capture more

complex features and patterns, which may perform better on some complex tasks. However, in some simple classification tasks, an overly deep network could lead to overfitting or difficulties in training, thereby reducing accuracy. Therefore, the selection of network depth should be balanced with the specific tasks and characteristics of the dataset.

2. **More Parameters Aren't Always Better**: ResNet50 has a larger number of parameters, meaning it can learn more features and provide more complex representations. However, more parameters also increase the risk of overfitting. On certain datasets, a smaller ResNet18 might perform better due to having an adequate number of parameters to handle the task's features and being less prone to overfitting. Therefore, when tuning models, it is crucial to choose the appropriate amount of parameters based on the size of the dataset and the complexity of the task.

These findings illustrate that no single model is perfect for all problems and highlight the importance of model tuning and optimization. Optimal results can be achieved by adjusting the model parameters according to different data and experimental purposes.

In our ongoing research and study, we are committed to learning and researching model-tuning strategies to find effective and efficient ways to identify suitable model parameters. This requires a deeper understanding of the field of deep learning and a comprehensive grasp of the computational principles of various models.

Conclusion

In this project, we conducted comprehensive experiments on the CNN model within the domain of image classification, continually optimizing our model through data augmentation and Bayesian optimization. Ultimately, we used the ResNet50 as a comparative model to study the differences between models on the same problem. Our research showed that the powerful pre-trained ResNet50 model did not demonstrate overwhelming superiority without any adjustments and also highlighted the excellent advantages of CNN due to its unique network structure in this field. This project research deepened our understanding of the CNN model and emphasized that there is no ultimate solution in the field of deep learning. A profound understanding of the problem at hand and the dataset, coupled with appropriate model tuning, can enable even simple and basic models to achieve excellent results

References

[1]: Xie, Lingxi, and Alan Yuille. "Genetic cnn." *Proceedings of the IEEE international conference on computer vision*. 2017.
[2]: Bhatt, Dulari, et al. "CNN variants for computer vision: History, architecture, application, challenges and future scope." *Electronics* 10.20 (2021): 2470.
[3]: Bird, J. J., and A. Lotfi. "CIFAKE: Image Classification and Explainable Identification of AI-Generated Synthetic Images." IEEE Access, vol. 12, 2024, pp. 15642-15650. IEEE, doi:10.1109/ACCESS.2024.3356122.
[4]: Shorten, Connor, and Taghi M. Khoshgoftaar. "A survey on image data augmentation for deep learning." *Journal of big data* 6.1 (2019): 1-48
[5]: Victoria, A. Helen, and Ganesh Maragatham. "Automatic tuning of hyperparameters using Bayesian optimization." *Evolving Systems* 12.1 (2021): 217-223.

Figure1:Bhatt, Dulari, et al. "CNN variants for computer vision: History, architecture, application, challenges and future scope." *Electronics* 10.20 (2021): 2470.
Figure 2：Mukherjee, Suvaditya. “The Annotated ResNet-50.” *Medium.Com*, 18 Aug. 2022, https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758.