Multi-dataset AI Image Forgery Detection Using CNNs

Caleb Hsu
University of Washington
Seattle, WA
chsu14@uw.edu

Catrina Yeh
University of Washington
Seattle, WA
catrinay@uw.edu

Hanna Ngo
University of Washington
Seattle, WA
hhn9@uw.edu

Jack Scott
University of Washington
Seattle, WA
jscott26@uw.edu

Abstract

As image editing software and AI generation tools become more accessible, distinguishing real images from manipulated ones has become a pressing challenge. This paper investigates the efficacy of a standard Convolutional Neural Network (CNN) for detecting different types of image manipulations. We evaluate the model's performance on two types of forgeries: traditional manipulations, such as splicing and copy-move (using the CASIA 2.0 dataset), and modern AI-generated images (using the CIFAKE dataset). Our experiments reveal a stark performance divergence: the CNN model is highly effective at identifying AI-generated images, achieving a peak F1-score of 0.939, while struggling with traditional forgeries, reaching only 0.720. More critically, we demonstrate that models trained on one forgery type fail to reliably detect the other, achieving a best F1-score of only 0.751 in this cross-dataset test. These findings suggest that the visual artifacts of AI generation are fundamentally different from those of manual manipulation, and that a single CNN architecture is insufficient as a universal image forgery detection solution. Our work highlights the need for more adaptable detection systems that can recognize both traditional editing techniques and modern AI-generated content as digital deception continues to evolve.

1 Introduction

In the modern digital age, images are a primary source of information, but the integrity of that information is increasingly under threat. These forgeries can be used to spread misinformation in journalism and social media, create false evidence in legal cases, and erode public trust. As a result, the ability to reliably detect image forgeries has become a critical area of research.

This challenge has recently become more complex, splitting into two distinct problems. The first is the "traditional" problem: detecting manual manipulations like splicing (pasting a part of one image onto another) and copy-move (cloning a part of an image to another location). Datasets like CASIA 2.0 were created to test a model's ability to find these specific, often low-level artifacts. The second is the "new" problem: detecting images that are entirely synthetic, created by modern generative AI. The CIFAKE dataset, for example, was developed to test a model's ability to distinguish these high-quality, AI-generated images from real photographs.

To combat both of these problems, many researchers have turned to deep learning, and specifically Convolutional Neural Networks (CNNs). CNNs are a powerful tool because they can analyze complex visual features and detect subtle signs of manipulation that are often invisible to the human eye.

Preprint. Under review.

Researchers have shown their effectiveness for both tasks: Patel et al. (1) demonstrated a CNN's ability to find traditional forgeries, while Bird and Lotfi (2) showed that a similar CNN architecture could be trained to detect AI-generated images with high accuracy.

This use of a common tool for two different problems leads to our central research question: while we know CNNs can be specialized for each task, can a single CNN generalize across them? Are the indications of a traditional splice and the indications of an AI-generated image similar enough for one model to learn both? This question of model generalization is the central focus of our investigation.

In this paper, we investigate the efficacy and generalization capabilities of a standard CNN architecture across these two distinct forgery domains. We conduct a series of experiments using the CASIA 2.0 and CIFAKE datasets. We first establish baseline performance by training and testing the model on each dataset individually. We then perform critical cross-dataset evaluations—training on traditional fakes and testing on AI fakes, and vice-versa—to measure the model's ability to generalize its findings.

Our findings reveal a stark performance divergence. This suggests that the visual artifacts of manual manipulation and AI generation are fundamentally different and that a universal forgery detector is a significant challenge.

2 Related Work

Deep-Learning Methods for Image Forensics Castillo Camacho and Wang (3) provide a comprehensive review of deep-learning-based methods in the field of image forensics, including image manipulation detection, deepfake image detection, and image falsification detection. They discuss the ability of CNNs to detect alterations that may not even be able to be seen with the human eye, supporting our method choice. However, while various models exist for various types of image forensics, there is not one able to generalize across the types.

CNN Architectures for Traditional Image Forgery Detection Patel et al. (1) use a CNN to a determine if an image has been manipulated, from retouching to splicing. The CASIA v2.0 dataset contains approximately 7,000 authentic and 5,000 tampered images created through copy-move and splicing techniques. Their algorithm consisted of a Error Level Analysis with deep-learning techniques in order to achieve a detection accuracy of 93%. Singh and Seghal (4) also utilize a CNN architecture with multiple convolution layers and an SVM classifier in their work. Their paper aims to detect if an image has been modified through a copy-move, splice, or object removal. Their algorithm gave them a 96% training accuracy, and SVM classification had a 96.8% effective accuracy. This shows that using a CNN has shown to be effective for image forgery detection.

AI-generated image Detection Bird and Lotfi (2) use a CNN to determine if an image was generated by AI or not. This model found that focusing on small imperfections in the background of the images gave the most accurate results. The dataset created for this study, called the CIFAKE dataset, was made publicly available for research, which is used in this paper.

Cross-dataset Generalization One of the main issues in image forgery detection is that the performance of models vary significantly depending on the kinds of forgery characteristics in the datasets. Patterns found in the CASIA dataset, like duplicated regions or compression inconsistencies, would not be considered patterns in the CIFAKE dataset. Thus, there is a gap for an AI-generated image forgery detection model across multiple datasets. Cao (5) utilizes multi-dataset image-label matching in open-world object detection, noting significant improvements compared to the baseline. This may suggest that training using multiple datasets will improve the overall performance of the models, which we can then apply into AI-generated image detection.

3 Technical Description/Algorithm

To investigate our research question on CNN generalization, we designed a series of experiments using two distinct forgery datasets and a single, standardized CNN architecture. This section details the datasets, the data preprocessing steps, the model architecture, and the training and evaluation procedures used in our study.

3.1 Datasets

Our experiments rely on two publicly available datasets, each representing a different class of image forgery.

- **CIFAKE:** This dataset, introduced by Bird et al., is designed for the problem of detecting AI-generated images. It consists of 120,000 *32 x 32* pixel images. The "real" images (60,000) are drawn from the original CIFAR-10 dataset. The "fake" images (60,000) are synthetically generated using a Latent Diffusion model, with prompts designed to mirror the CIFAR-10 classes. Our training set contains 100,000 images (50,000 real, 50,000 fake), and the test set contains 20,000 images (10,000 real, 10,000 fake).
- **CASIA 2.0:** This, via Patel et al., is a widely used dataset for traditional image forgery detection. It contains images that have been manipulated using splicing and copy-move techniques. Unlike CIFAKE, these images vary in resolution and complexity. For our experiments, the dataset was split into 9,426 training images (55,547 real, 3,879 fake) and 3,188 test images (1,944 real, 1,244 fake).
- **DEEPGUARD:** This dataset (6) contains 13,000 images, with 6,500 AI-generated and real images. In it, there are 4 different AI models, DALL-E, GLIDE, IMAGEN, and SD. The synthetic images were generated by the prompts used to find the real images. This dataset was used mainly as a testing set against our main model.

3.2 Data Preprocessing

A critical step in our methodology was standardizing the input for our CNN. As implemented in our PyTorch pipeline, all images from both datasets were subjected to the same two transformations:

1. **Resize:** All images were resized to *32 x 32* pixels. This was necessary to match the architecture's input dimensions, which were designed for the CIFAKE dataset.
2. **To Tensor:** Images were converted to PyTorch tensors, and pixel values were normalized from (0, 255) to a (0.0, 1.0) range.

Notably, this approach differs from the methods like those in Patel et al. (1), which use a specialized preprocessing step like Error Level Analysis (ELA) to highlight compression artifacts. Our implementation feeds the raw, resized pixel data directly into the network, tasking the CNN with learning the forgery artifacts without this specialized guidance. This decision has significant implications for detecting traditional forgeries, as the resizing process may obscure the high-frequency details that ELA is designed to capture.

For the combined dataset and subsequent generalization tests (Experiments 3-5), we introduced additional data augmentation techniques to improve model robustness, including random horizontal flip and affine transformation (rotation and shear).

3.3 Model Architecture

The algorithm we implemented is a standard Convolutional Neural Network (CNN), identical to the one proposed by Bird et al. (2) for the CIFAKE dataset. The architecture, shown in Figure 1, is intentionally simple to serve as a clear baseline to see how well the model can generalize to different tasks.

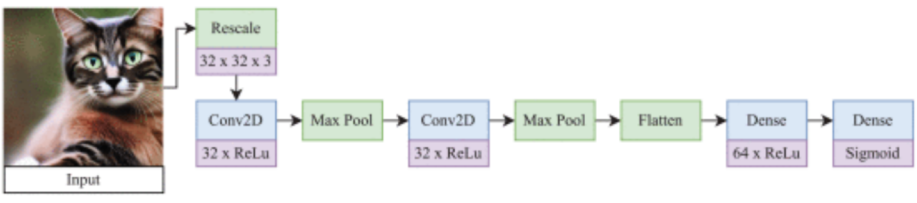

Figure 1: The CNN architecture used for all experiments, based on the model from Bird et al.

The model consists of two convolutional blocks followed by a classifier head:

1. **Input Layer:** A *32 x 32 x 3* image tensor.
2. **Convolutional Block 1:**
   - A 2D Convolutional layer (*Conv2D*) with 32 filters and a *3 x 3* kernel, using the ReLU activation function.
   - A *MaxPool2D* layer with a *2 x 2* window and a stride of 2 to downsample the feature map.
3. **Convolutional Block 2**
   - A second *Conv2D* layer with 32 filters and *3 x 3* kernel, again using ReLU activation.
   - A second *MaxPool2D* layer *2 x 2* to further downsample the features.
4. **Classifier Head:**
   - A *Flatten* layer to convert the 2D feature maps into a 1D vector.
   - A *Dense* (fully connected) layer with 64 units and ReLU activation.
   - An **Output Layer** (*Dense*) with a single unit and a **Sigmoid** activation function.

The final sigmoid activation compresses the output to a value between 0.0 and 1.0, which is interpreted as the model's confidence. A value close to 0.0 indicates a prediction of "Fake," and a value close to 1.0 indicates "Real."

3.4 Training and Evaluations

The model was implemented in Python using the PyTorch library and was trained on a GPU using CUDA for acceleration.

- **Loss Function:** We used **Binary Cross-Entropy Loss** (nn.BCELoss). This loss function is the standard choice for binary classification problems, as it effectively measures the penalty for a strong prediction.
- **Optimizer:** The model's weights were updated during training using the **Adam optimizer** (torch.optim.Adam), a widely used an effective algorithm for gradient descent.
- **Evaluation Metrics:** To provide a comprehensive picture of the model's performance, we evaluate it using four standard metrics. Given True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN):
  - **Accuracy:** The percentage of all correct predictions.
  $$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$
  - **Precision:** Of all the images the model predicted as "Real," what percentage actually were "Real"?
  $$\text{Precision} = \frac{TP}{TP + FP}$$
  - **Recall:** Of all the "Real" images in the dataset, what percentage did the model correctly identify?
  $$\text{Recall} = \frac{TP}{TP + FN}$$
  - **F1-Score:** The harmonic mean of Precision and Recall. This is the most important metric as it balances the trade-off between false positives and false negatives, which is crucial for an unbalanced dataset like CASIA.
  $$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

4 Experiments

Colab Notebook for the experiments can be found here: `https://colab.research.google.com/drive/1I_GnrKPcAP16K-S7gXuD0zPbM5EsOWGq?usp=sharing`

To evaluate our model's performance and test its generalization capabilities, we conducted a series of five core experiments. We first established two baseline performance benchmarks: one for detecting AI-generated forgeries (CIFAKE) and one for traditional forgeries (CASIA). We then performed cross-dataset generalization tests and, finally, evaluated the performance of a model trained on a combined dataset.

The results are presented using the four metrics defined in our methodology: Accuracy, Precision, Recall, and F1-Score. The F1-Score is our primary metric for comparison as it provides a balanced measure of a model's performance, which is especially important for the unbalanced CASIA dataset.

4.1 Experiment 1: Baseline on AI-Generated Images (CIFAKE)

First, we established a baseline by training and testing the model exclusively on the CIFAKE dataset. The model was trained on 100,000 CIFAKE images and evaluated on the 20,000-image test set.

Result: The model performed exceptionally well, confirming its effectiveness for this specific task. After tuning, the best-performing model (50 epochs, 0.0001 learning rate, 32 batch size) achieved an **F1-Score of 0.939**, with an **Accuracy of 0.938**, **Precision of 0.917**, and **Recall of 0.963**. The training loss curve for this experiment is shown in Figure 2. The graph shows a rapid decrease in loss within the first 10 epochs, followed by a steady, gradual convergence over the remaining 40 epochs. The loss begins at approximately 0.35 and ends at a very low value of approximately 0.10. This smooth, consistent decline is characteristic of a successful training process, indicating the model was effectively learning the features required to distinguish real images from AI-generated fakes.

Figure 2: Training loss for the baseline CIFAKE model.

4.2 Experiment 2: Baseline on Traditional Forgeries (CASIA)

Next, we established a baseline for traditional forgeries by training and testing exclusively on CASIA 2.0 dataset. The model was trained on 9,426 CASIA images and evaluated on 3,188 test images.

Result: The performance was significantly lower. The model struggled to learn the features associated with splicing and copy-move forgeries. The best-performing model (30 epochs, 0.01 learning rate, 64 batch size) achieved a peak **F1-Score of only 0.720**, with an **Accuracy of 0.637**, **Precision of 0.680**, and **Recall of 0.766**. Figure 3 shows the training loss for this model. In sharp contrast to the CIFAKE model, the loss here decreases far more slowly and erratically and also diverges from the validation loss indicating bad generalization to the validation set. After 30 epochs, the loss plateaus at a high value of approximately 0.50. This high final loss, more than 5x that of the CIFAKE model, visually confirms that the CNN struggled to find consistent, learnable features for detecting traditional forgeries.

4.3 Experiment 3: Cross-Dataset Generalization Tests

The most critical experiments were the cross-dataset tests to measure generalization. We trained a model on one forgery type and tested it on the other.

Result (Train: CIFAKE, Test: CASIA): We took a high-performing model from Experiment 1 (trained on AI fakes) and evaluated it on the traditional fakes in the CASIA test set. The model achieved an **F1-Score of 0.751** and an **Accuracy of 0.609**. While this is marginally better than the

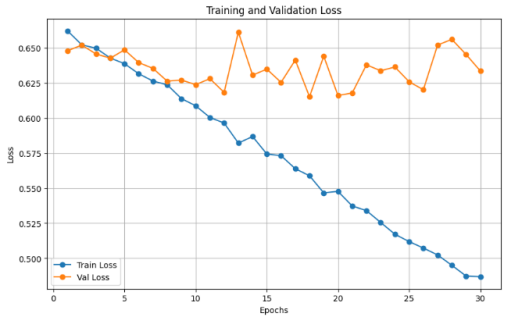

Figure 3: Training loss for the baseline CASIA model.

CASIA-only model, a deeper look reveals an extremely high **Recall of 0.967** but very low **Precision of 0.614**. This indicates the model was heavily biased and incorrectly classified a large number of real images as "fake," making it unreliable. The training loss for this model, shown in Figure 4, demonstrates that the model did train successfully on the CIFAKE data. The loss curve shows a clear, fast convergence to a low value (approx. 0.20) in just 15 epochs. This confirms that the model became an effective CIFAKE detector; its failure was not in learning, but in generalizing what it learned to the new task.

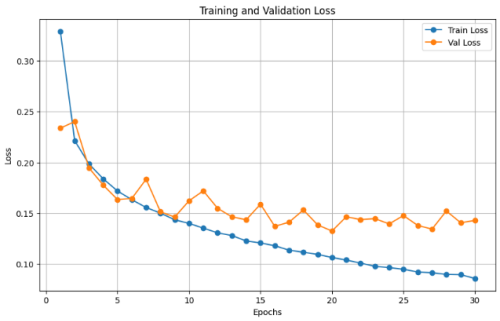

Figure 4: Training loss for the CIFAKE-trained generalization model.

We did not conduct the reverse test (Train: CASIA, Test: CIFAKE) as the model from Experiment 2 had already failed to learn a robust set of features (0.720 F1-Score), making a generalization test inconclusive.

4.4 Experiment 4: Combined Dataset Training

Finally, we explored whether training the model on a combined dataset of both forgery types would create a more robust, "universal" detector.

Result (Train: Both, Test: CIFAKE): When tested on the AI-generated images from CIFAKE, the combined modeled achieved an **F1-Score of 0.939**, with an **Accuracy of 0.939**, **Precision of 0.930**, and **Recall of 0.949**. This is identical to the baseline F1-Score from Experiment 1, showing that adding the CASIA data did not hurt its ability to detect AI fakes. The training loss for this combined model again shows a smooth convergence to a low loss value (approx. 0.16). This indicates that even with the CASIA data present, the model's training was dominated by the features from the much larger and more consistent CIFAKE dataset.

Result (Train: Both, Test: CASIA): When tested on the traditional forgeries from CASIA, the combined model achieved an **F1-Score of 0.745**, with an **Accuracy of 0.620**, **Precision of 0.630**, and **Recall of 0.910**. This is a negligible improvement over the other models and still demonstrates a fundamental inability to reliably detect traditional forgeries. The training loss for this specific run converges to a value (approx. 0.27) that is higher than the CIFAKE models but lower than the

CASIA-only model. This suggests that while the model primarily learned AI-related features, the presence of the CASIA data had a slight, but not significant, impact on the training process.

4.5 Experiment 5: Finalized Baseline Models and Additional Preprocessing Steps

We then tested our model on varying combinations of data with additional preprocessing steps. Namely, randomized affine transformations to the input images and randomized flips. In addition, the following plots were trained for 50 epochs and represent the finalized baseline models that we will use to discuss the results of.

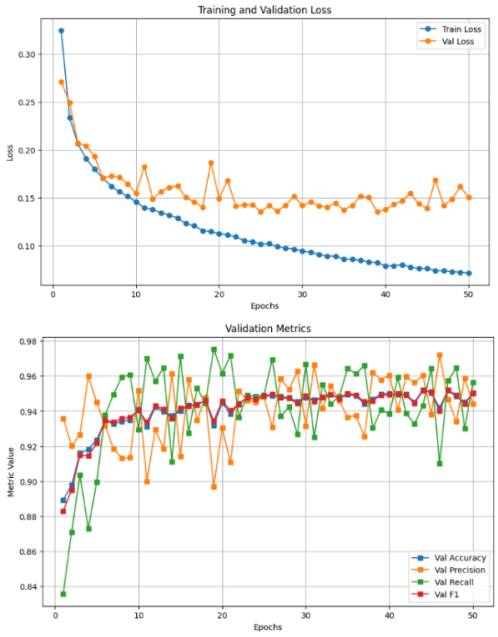

Figure 5: Plots for the model trained on CIFAKE data and tested on CIFAKE data.

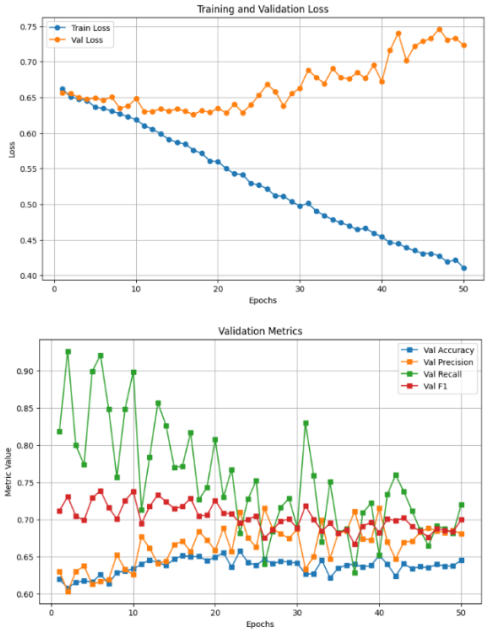

Figure 6: Plots for the model trained on CASIA data and tested on CASIA data.

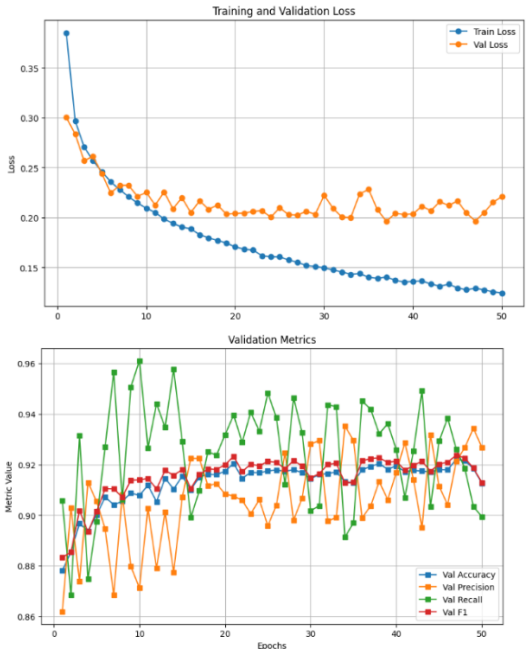

Figure 7: Plots for the model trained on both sets (CASIA AND CIFAKE) data and tested on both datasets.

4.6 Experiment 6: Cross-AI Model Generalization

The model trained on CIFAKE (Experiment 5, Figure 5) was a highly effective detector for its domain. We conducted a critical test of cross-generator generalization by evaluating this same model on images generated by other contemporary AI models: **DALL-E**, **GLIDE**, and **IMAGEN**, as well as a separate test set from a **Stable Diffusion (SD)** model. These images (real and fake), came from the

DEEPGUARD dataset. The goal was to determine if the artifacts learned from one AI model (the CIFAKE Latent Diffusion model) were transferable to another.

Result: The model demonstrated poor generalization across most novel AI models, confirming that success is highly dependent on matching the detector to the generating model's unique artifacts.

- **DALL-E, GLIDE, and IMAGEN:** Testing on DALL-E resulted in an **F1-score of 0.640**, testing on GLIDE yielded an **F1-score of 0.617**, and on IMAGEN it yielded an **F1-score of 0.630**. These results demonstrate a significant failure in detecting forgeries outside of the training distribution. The model failed to recognize the unique noise signatures of these distinct generators.
- **Stable Diffusion (SD):** An exception was observed with the Stable Diffusion (SD) dataset, where the model achieved a moderately higher **F1-score of 0.703**. Given that the CIFAKE dataset was also generated using a Latent Diffusion model, this improved performance likely stems from the two datasets sharing more fundamental, latent artifacts than the other, non-diffusion-based AI models.

This stark drop in performance reinforces the conclusion that forgery detection is a collection of highly specific problems.

4.7 Summary of Results

Our complete experimental findings are summarized in Table 1. The data clearly shows high proficiency in detecting AI-generated images (through the stable diffusion model used by CIFAKE) and a persistent failure to detect traditional forgeries with the same architecture and preprocessing even with extensive new training data. The two main takeaways are the stark divergence in baseline performance (CIFAKE F1-Score of 0.949 vs. CASIA F1-Score of 0.667) and the failure of cross-dataset generalization when training on CIFAKE and testing on CASIA (F1-Score of 0.751). The results of Experiment 6 further reinforced this generalization gap, showing poor performance when testing against novel AI generators like DALL-E (F1-Score of 0.640) and GLIDE (F1-Score of 0.617) with a model trained exclusively on CIFAKE. This indicates that the features our model detects to predict fake images do no generalize to other AI generators; We will continue a discussion of this upon visualizing our CNN's output layer gradients.

Table 1: Summary of Best Performance Across All Experiments

Training Dataset	Testing Dataset	F1-Score	Accuracy	Precision	Recall
CIFAKE (Baseline)	**CIFAKE**	**0.949**	0.949	0.947	0.951
CASIA (Baseline)	**CASIA**	**0.667**	0.594	0.667	0.667
CIFAKE (Generalization)	**CASIA**	**0.751**	0.609	0.614	0.967
Combined (Both)	**Combined (Both)**	**0.889**	0.887	0.904	0.874
CIFAKE (Baseline)	**DALL-E**	**0.640**	0.516	0.510	0.860
CIFAKE (Baseline)	**GLIDE**	**0.617**	0.480	0.488	0.838
CIFAKE (Baseline)	**IMAGEN**	**0.630**	0.507	0.504	0.840
CIFAKE (Baseline)	**SD**	**0.703**	0.638	0.595	0.860

5 Discussion

Our central research question was whether a single CNN architecture could generalize across traditional (splicing, copy-move) and modern (AI-generated) forgeries. The results, summarized in Table 1, show that it cannot.

5.1 Interpreting Experimental Results

The most significant finding is the stark performance divergence between the two forgery types. Our model achieved a high-performing baseline on the AI-generated CIFAKE dataset (0.949 F1-Score) but failed to achieve a reliable baseline on the traditional CASIA 2.0 dataset (0.667 F1-Score).

This divergence strongly supports the hypothesis that the model is learning to detect *specific artifacts* that are not shared between forgery methods.

- **Success on CIFAKE (Strengths):** The model's high performance (0.949 F1-Score) replicates the findings of Bird et al. (2), whose architecture we implemented. This confirms that our CNN is highly effective at identifying the specific visual artifacts produced by the Latent Diffusion model used to create the CIFAKE dataset. As noted in the Bird et al. (2) paper, the model is likely learning to find "small visual imperfections" and "visual glitches" that are consistent across the 50,000 fake images.
- **Failure on CASIA (Weaknesses):** The model's poor performance (0.667 F1-Score) on traditional forgeries is just as significant. The indications for a spliced or copy-move image, such as unnatural edges, mismatched lighting, or compression differences between regions, are fundamentally different from the indications of an AI-generated image. Our model, trained to find one, was ineffective at identifying the other.

Analysis of Combined Training: By analyzing the model trained on the combined dataset (Experiment 4), we gain further insight into the generalization challenge. The training and validation loss curves (Figure 8) converge relatively closely, as both the training and validation losses decrease steadily and stabilize at around 8-10 epochs. This convergence suggest minimal overfitting and good generalization. However, Figure 9 shows a distinct variation between validation metrics, exhibiting no clear convergence pattern. Accuracy, precision, recall, and F1 scores fluctuate quite a bit, ranging between approximately 0.60 and 1.00. This suggests that the model struggles to learn stable features that can be generalized across datasets of traditional manipulations and AI-generated images. This instability contrasts with the low loss seen in Figure 8, indicating that low loss might not necessarily mean consistent classification. This fluctuation likely stems from patterns learned from one dataset being inappropriately applied to the other in certain cases, preventing the model from consistently applying the right patterns to the right images.

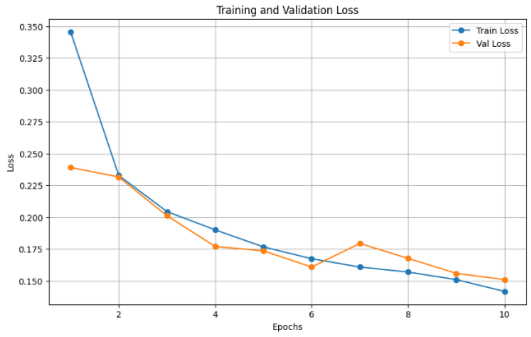

Figure 8: Training and validation loss over epochs for the combined dataset model

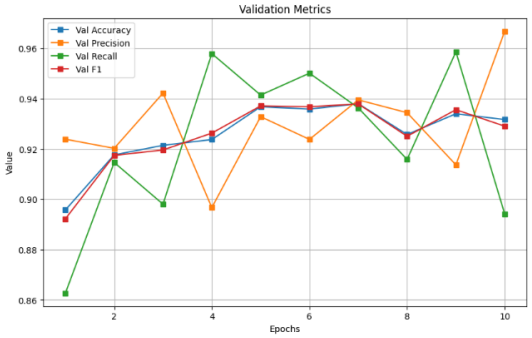

Figure 9: Validation accuracy, precision, recall, and F1 score over epochs for the combined dataset model.

5.2 Generalization Failure and Study Limitations

Our results highlight a critical discrepancy when compared to our initial base paper, Patel et al. (1), which reported 93% accuracy on traditional forgeries. Our CASIA baseline (0.720 F1-Score) did not come close. We have identified two primary reasons for this, which also serve as the main limitations of our study.

1. **Missing ELA Preprocessing:** The Patel et al. paper integrated Error Level Analysis (ELA) with their CNN. ELA is a technique specifically designed to highlight compression mismatches, which are a key artifact in spliced images. Our model was fed only raw pixel data, so it never "saw" the artifacts that the Patel et al. model was trained to find.
2. **Destructive Resizing:** The CASIA dataset contains high-resolution images. Our architecture, adopted from the CIFAKE paper, required *32 x 32* input. This aggressive downsampling likely destroyed teh very high-frequency artifacts (like subtle edge mismatches) that are needed to detect traditional forgeries.

This context makes our generalization test (Experiment 3) even more insightful. The model trained on CIFAKE artifacts achieved an F1-Score of 0.751 on CASIA. A deeper look at the metrics show a very high Recall (0.967) but extremely low Precision (0.614). This means that the model was unreliable, generating a high number of false positives, confirming that the features it learned for AI detection were not helpful for traditional forgery detection.

5.3 Visualizing Feature Learning (XAI)

To visualize what our model learned, we used the same Explainable AI (XAI) techniques used in CIFAKE, specifically Grad-CAM, to show the regions of an image that influence the model's predictions.

CIFAKE Baseline Feature Learning: Figure 10 shows several test images from the CIFAKE dataset with a corresponding heatmap, where warmer colors, like red and orange, indicate the regions that had a stronger influence on the model's decision to label it fake. Notice that the model does not place heavy emphasis on the 'entity' depicted in each image (frog, deer, robot, horse). Rather, very strong gradients are detected arbitrarily throughout the image. This aligns with prior work from Patel et al. and is indicative of the models decisions to detect subtle artifacts (noise) within the image rather than large visual imperfections.

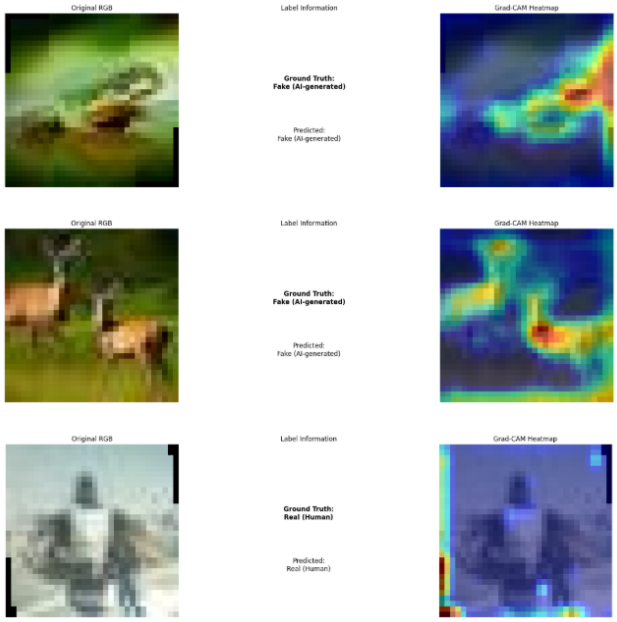

Figure 10: XAI heatmap highlighting regions influencing the model's prediction for a test image.

CASIA Generalization Failure: Contrasting the results shown in Figure 11, the heat maps for predictions on the CASIA data (with a model trained on CIFAKE) show the models inability to detect similar features on these new data points whether they are manipulated or not. This is very representative of the inherent inability of the model to learn on one type of noise artifact and generalize to all other image manipulation artifacts.

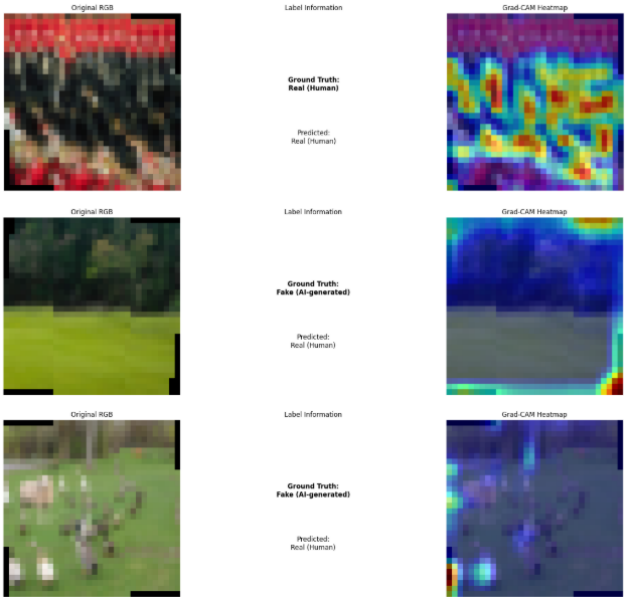

Figure 11: XAI heatmap highlighting regions influencing the model's prediction for a test image.

Cross-AI Generalization Failure: To gain insight into the model's failure to generalize across different AI generators (Experiment 5), we analyzed heatmaps for images from DALL-E, GLIDE, and Stable Diffusion (SD) when evaluated by the CIFAKE-trained model.

- In contrast to the CIFAKE baseline, the heatmaps for DALL-E (Figure 12) and GLIDE (Figure 13) show a highly dispersed and arbitrary activation pattern. The model does not consistently focus on artifact regions but often activates along generic semantic regions or seemingly random spots. This visual evidence confirms that the low F1-scores for DALL-E (0.640) and GLIDE (0.617) are due to a fundamental failure in feature transfer, as the unique noise signatures learned from the CIFAKE model were not transferable to the artifacts introduced by these distinct generation pipelines.
- The SD test achieved a moderately better F1-score of 0.703. As shown in Figure 14, while its heatmap is less dispersed than DALL-E and GLIDE, it still lacks the consistent, high-confidence detection signal seen in the CIFAKE baseline. This pattern suggests a reliance on features less robust than the training data, leading to the low reliability metrics.

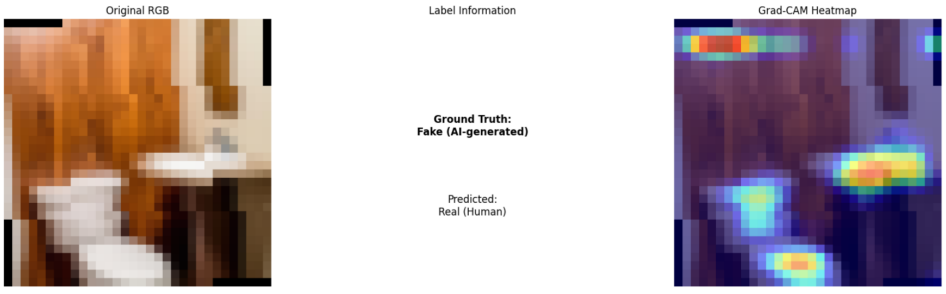

Figure 12: XAI heatmap for a DALL-E-generated image when tested on the CIFake-trained model.

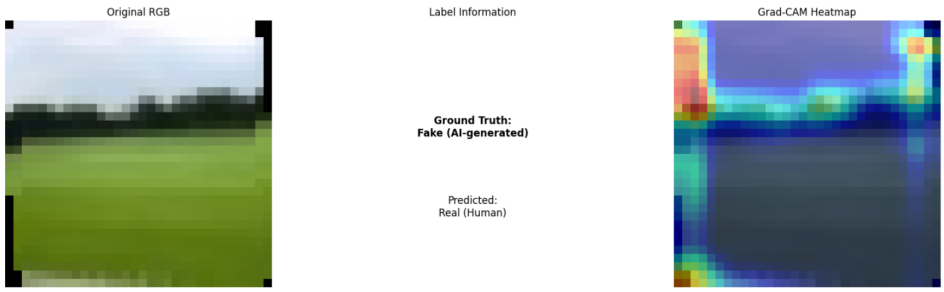

Figure 13: XAI heatmap for a GLIDE-generated image when tested on the CIFake-trained model.

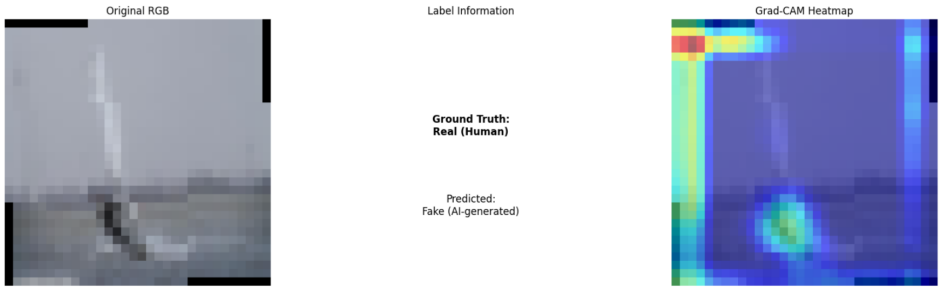

Figure 14: XAI heatmap for a Stable Diffusion(SD)-generated image when tested on the CIFake-trained model.

This collective lack of clear, consistent artifact localization across novel generators visually underscores our quantitative conclusion that a single CNN architecture is insufficient as a universal forgery detector.

5.4 Connection to Prior Work

Our approach directly builds on the work of Patel et al. (1) and Bird et al. (2), both of which used a CNN to detect real or fake images in the CASIA and CIFAKE datasets. However, these two papers focused on forgery detection on traditional image manipulations or AI-generated images, while our paper aimed to train a model that can detect both. Previous works have analyzed the generalization challenges within each domain, such as GAN-generated images vs. diffusion-generated images (3), but the cross-generalization across multiple categories in image forensics has yet to been explored.

Unlike multi-dataset training approaches that combine multiple datasets with the same forgery type (5), our work focuses on bridging the gap between different manipulations. While existing work optimizes for accuracy (4) within singular domains, we demonstrate that such optimization may come at the cost of cross-domain robustness. This finding has significant implications for real-world deployment, where detection systems must handle diverse and evolving manipulation techniques.

5.5 Implications for the Field

A key implication for this work is that creating a single detector for images generated by different models is a significant challenge.

Our results show that forgery detection is not a single problem, but a collection of highly specific ones. The CIFAKE dataset is generated by a single model. A detector that achieves 94% on it will likely fail on forgeries from Midjourney, DALL-E 3, or future models, as each will have its own unique "noise" or artifact signature.

The results from Experiment 5 confirm this lack of generalization extends to modern AI models as well. A detector is highly successful on CIFAKE failed to reliably detect forgeries from DALL-E and GLIDE, confirming that each generative model produces its own unique and "noise" that is difficult to generalize.

Furthermore, our hypothesis that a model cannot be effectively trained on all forgery types simultaneously was also confirmed. When we trained our model on a combined dataset (Experiment 4), its performance on CASIA (0.751 F1-Score) did not significantly improve. The model was clearly dominated by the larger, more consistent CIFAKE dataset, and failed to learn the features for the CASIA data. This shows that simply adding more varied data into one simple CNN is not a path to a "universal detector."

This project's main strength is in clearly demonstrating the lack of generalization. It proves that to create effective, real-world forgery detectors, researchers must focus on methods that are either (a) specifically tuned to the artifacts of individual generative models or (b) learn to find more fundamental, high-level inconsistencies rather than just low-level pixel noise.

6 Future Work

One direction for future work would be through creating multiple lightweight models, each trained on a dataset from a different generative AI model. Then, a given input image would be processed by all of the models in parallel to determine if there is any positive detection. However, there are certain limitations with this, as the amount of data required for this is very high.

We could also expand our research into other types of forgery detection, such as deep-faked videos or face-swap manipulations, and see if the generalization issues still persist within these fields. There are also many emerging generative AI models that we should consider training on. As these models get better, detecting if an image was created by one of these models will only get harder.

References

[1] M. Patel, K. Rane, N. Jain, P. Mhatre, and S. Jaswal, "Image forgery detection using CNN," in *Proc. 3rd Int. Conf. Intelligent Technologies (CONIT)*, Hubli, India, Jun. 2023, pp. 1–4, doi: 10.1109/CONIT59222.2023.10205377.

[2] J. J. Bird and A. Lotfi, "CIFAKE: Image classification and explainable identification of AI-generated synthetic images," *IEEE Access*, vol. 12, pp. 15642–15650, 2024, doi: 10.1109/ACCESS.2024.3356122.

[3] I. Castillo Camacho and K. Wang, "A comprehensive review of deep-learning-based methods for image forensics," *J. Imaging*, vol. 7, no. 4, p. 69, Apr. 2021, doi: 10.3390/jimaging7040069.

[4] S. Singh and V. K. Sehgal, "Image forgery detection model using CNN architecture with SVM classifier," in *Proc. 7th Int. Conf. Parallel, Distributed and Grid Computing (PDGC)*, Solan, India, Nov. 2022, pp. 263–268, doi: 10.1109/PDGC56933.2022.10053298.

[5] P. Cao, "Open-world object detection with multi-dataset image-label matching," SSRN, 2025. [Online]. Available: https://ssrn.com/abstract=5141716

[6] Ikram Reghioua, Mouna Yasmine Namani, Gueltoum Bendiab, Mohamed Aymen Labiod, Stavros Shiaeles, "DeepGuardDB: Real and Text-to-Image Synthetic Images Dataset,", doi: 10.21227/10ap-pk52.