Cairo University

Faculty of Computers and Artificial Intelligence

Information Technology Department

Academic Year 2022/2023

Image Forgery Detection

Implemented By:

Mahmoud Ashraf Farouk	20190492
Nada Khaled Mohammed	20190702
Ziad Mahmoud Abdel-Mohsen	20190226
Malak Amr Mostafa	20190551
Youssef Emad Attia	20190641
Merna Mohammed Raghy	20190699

Under Supervision of:

Dr. Mona Soliman

TA. Ibrahim Zidane

July 2023

List of Tables

List of Figures

Chapter 1: Introduction

With the expansion of picture altering programming and the simplicity of advanced control, the location of picture phonies has become progressively testing. In this study, we provide a comprehensive overview of the most recent developments in image forgery detection methods based on deep learning. We look at three distinct research projects that highlight novel image manipulation-fighting strategies and architectures.

At the principal our venture centers around the examination of two sorts of AlexNet models: modified and pre-trained AlexNet. The authors demonstrate that, in comparison to the pre-trained AlexNet, the modified AlexNet model with enhancements like the sigmoid activation function, max activation function, and batch normalization achieves significantly higher accuracy in detecting and extracting features from manipulated images. These datasets include CASIA v1.0, MICC-F220, and CoMoFoD.

The Scientists and researchers use deep learning and Error Level Analysis (ELA) to identify image manipulations in the second phase of the project. The proposed model has a training accuracy of 92.2% and a validation accuracy of 88.46% after 100 epochs by comparing the compression ratios of real and fake images and the metadata, which is subject to change. This blend of ELA and profound learning shows promising outcomes in identifying picture controls and separating among certifiable and counterfeit pictures.

BusterNet, a novel deep neural architecture designed specifically for image copy-move forgery detection (CMFD), is presented at the project's third step. Dissimilar to past methodologies, BusterNet is an unadulterated start to finish teachable profound brain network that succeeds in limiting potential control locales in light of visual relics and duplicate move districts through visual similitudes. It is the principal CMFD calculation with the capacity to perceive source and target locales, displaying better execution looked at than cutting edge CMFD calculations on CASIA and CoMoFoD datasets. Additionally, BusterNet demonstrates resistance to a variety of known attacks.

Together, these projects make a contribution to the development of deep learning methods for image forgery detection. The BusterNet architecture, ELA-based analysis, and modified AlexNet model all demonstrate remarkable progress in detecting manipulated images, distinguishing between genuine and manipulated regions, and achieving high detection accuracy. In order to deal with the difficulties posed by image forgery and guarantee the authenticity of digital content, these advancements provide useful tools.

1.1 Problem statement

In a number of fields, such as digital forensics, scientific publications, medical imaging, journalism, insurance claims, and political campaigns, the increasing prevalence of digital image manipulation presents significant difficulties. It is now simple for people with good intentions and bad intentions to alter and forge digital images thanks to the availability of cutting-edge image editing software and tools. As a result, determining an image's authenticity and integrity has emerged as a significant concern for both researchers and practitioners as shown in Figure 1.

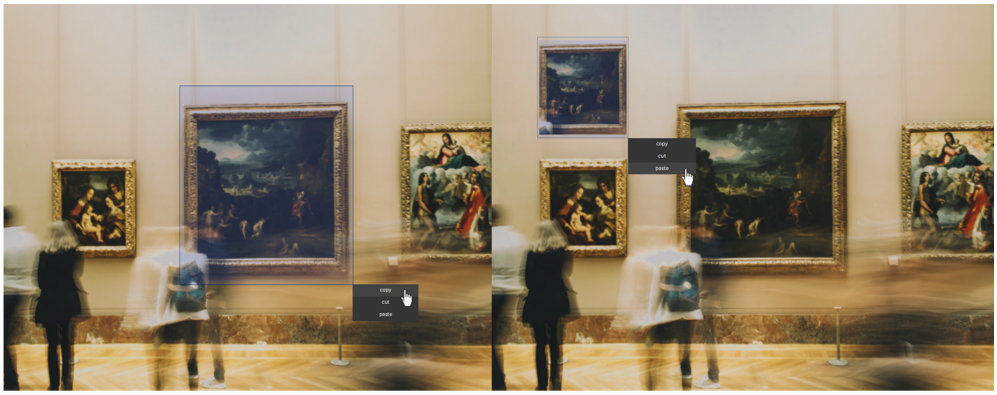


Figure 1: Copy-move image forgery

The problem lies in coming up with reliable and precise ways to tell the difference between real and manipulated images. One of the most common and well-known forms of digital image editing is the detection of copy-move image forgery, in which parts of an image are duplicated and pasted onto different parts of the same image. For digital content to be trustworthy and credible, it is essential to find such manipulations.

The issue at hand necessitates the creation of a reliable detection model capable of distinguishing genuine images from copy-move forgeries. Digital images should be able to be analyzed by the model, and it should be able to identify any parts of the same image that have been copied and pasted. In order to accurately assess the image's authenticity, the model must first identify the manipulated areas and their source regions.

Using cutting-edge methods from digital image analysis and machine learning, the proposed detection model seeks to address this issue. Researchers aim to create an automated and effective solution that can accurately detect copy-move forgery and provide insights into the authenticity of digital images by developing robust algorithms and utilizing deep learning architectures.

There would be far-reaching consequences if such a detection model was developed successfully. It would not only help experts analyze and verify the authenticity of evidence in digital forensics investigations, but it would also have implications for

journalism, where ensuring the integrity of images used in reporting is of the utmost importance. Reliable image authentication techniques can also help safeguard businesses like insurance claims and medical imaging from fraudulent practices.

A multidisciplinary approach that incorporates expertise from computer vision, image processing, machine learning, and digital forensics is required to address the issue of copy-move image forgery detection. In order to accurately assess the authenticity of digital images, it necessitates the creation of sophisticated algorithms and the application of cutting-edge methods.

In the accompanying areas, we will investigate the proposed discovery model and the systems utilized to resolve this issue. We'll talk about the algorithms and methods used to find and analyze copy-move forgery, as well as how to train and evaluate a reliable and accurate detection model. Our goal is to develop a useful tool for detecting and confirming the authenticity of digital images and contribute to the development of image forensic methods through this study.

1.2. Motivation

In the present computerized period, the far reaching utilization of online entertainment has altered how individuals connect and share data, including the sharing of pictures. However, this widespread sharing has also led to a worrying trend of image forgery, in which viewers are tricked or misled by images that have been altered. This problem has only been made worse by the rapid growth of digital images and the ease with which software for editing images can be obtained. As a result, image forgery detection methods that work have become more important than ever.

The creation of fake news, in which tampered images are frequently used to support false narratives, is one common form of image manipulation. The sheer volume of images shared on social media platforms demonstrates the phenomenon's widespread nature. A study uncovers that a stunning 80 million pictures are shared consistently on Instagram alone. This alarming trend emphasizes the urgent requirement for methods and tools that can assist individuals in determining the authenticity and integrity of the images they encounter.

The study of image forensic analysis is concerned with determining the authenticity and origin of images. Because it frequently provides crucial insights into the authenticity of an image, researchers have developed a variety of methods to evaluate the degree of image compression. Error Level Analysis (ELA), which makes use of various levels of compression to identify digitally altered images, is one approach that is extensively utilized in this regard. ELA has demonstrated promise in its ability to

differentiate between genuine and manipulated images, providing useful insights into the image's integrity.

Profound learning, a quickly developing part of AI, offers a promising way to deal with computerizing the identification of picture controls. Deep neural networks are well-suited for image forgery detection tasks due to their capacity to learn intricate patterns and features. Analysts have investigated the use of profound learning procedures, including Convolutional Brain Organizations (CNN), in recognizing genuine and controlled pictures. Due to their efficiency in processing two-dimensional data, CNNs have attracted a lot of attention, making them ideal for image-related tasks.

Researchers have made remarkable progress in developing deep learning models for the detection of image forgery through extensive research and experimentation. In terms of detecting image manipulations, new architectures like BusterNet, ELA in conjunction with deep learning, and modified AlexNet models have demonstrated promising results. These creative methodologies mean to address the difficulties presented by picture fraud and give more vigorous and exact techniques for recognizing genuine and counterfeit pictures.

In this unique circumstance, this thorough survey means to combine and present the critical discoveries from these examination attempts. By dissecting the progressions in profound learning-based procedures, the mix of ELA with AI models, and the original structures proposed, we gain important bits of knowledge into the cutting edge techniques for picture falsification location. In addition, we talk about how these developments might affect the fight against image forgery and the verification of digital content's authenticity.

This paper's subsequent sections go into greater detail about the particular projects and approaches described in the abstracts that came before them. For the purpose of detecting image forgery, we investigate the new BusterNet architecture, ELA-based analysis, and modified AlexNet models. In addition, we provide a summary of the experimental outcomes and evaluations carried out as well as a discussion of the drawbacks of the methods that are currently in use. In the end, the goal of this review is to inspire further advancements in this important area of research and to add to the growing body of knowledge about image forgery detection.

1.3 Project Objectives

- Discrimination between authentic and forged images: AlexNet can be trained to distinguish between authentic and forged images by learning to extract features that are characteristic of authentic images and different from those of forged images. The ultimate goal is to have a model that can accurately classify new images as authentic or forged based on these learned features.
- Real-time detection: In some scenarios, such as forensic investigations, it is important to have a model that can quickly and accurately detect forgeries in real-time. AlexNet is a computationally efficient architecture that can be used for real-time image forgery detection on a range of hardware devices, including CPUs, GPUs, and mobile devices.
- Scalability to large datasets: Image forgery detection often requires training on large datasets to ensure that the model can generalize well to new, unseen images. AlexNet can be trained on large datasets using techniques such as data augmentation and transfer learning to improve its accuracy and robustness.
- Interpretability: AlexNet can be modified to output class activation maps or saliency maps that highlight the features of the image that are most important for the model's decision, providing a form of interpretability.
- Digital image manipulations can be detected using Error Level Analysis (ELA): The essential target of using Blunder Level Examination (ELA) is to recognize advanced picture controls by dissecting the distinctions in mistake levels presented during the pressure cycle. The goal is to foster a strong strategy that can precisely recognize pictures that have gone through altering, for example, duplicate move fabrications or grafting.
- Differentiation between regions that were altered and those that were real: The ability to differentiate between the real and manipulated parts of an image is a crucial objective. The goal of performing ELA is to locate and highlight the altered regions, allowing for precise localization of the fake areas. Image forgeries can be better investigated and analyzed with the help of this objective.
- Evaluation of picture pressure levels: Another goal is to survey the degrees of pressure applied to various locales inside a picture. ELA enables the identification of inconsistencies that may indicate image manipulations by providing insights into the variations in compression levels. The identification of altered or altered regions is made possible by this objective.
- Mix with profound learning: In order to improve image forgery detection accuracy and efficiency, one goal is to combine ELA and deep learning methods. By consolidating the qualities of ELA and profound learning models, for example,

AlexNet, the goal is to foster a complete methodology that can identify and restrict picture phonies with further developed execution.

- Analyses of forgery detection software: The goal is to use ELA as a benchmarking device to assess the viability and execution of various picture fraud identification calculations. Researchers are able to evaluate the strengths and weaknesses of various approaches by contrasting the ground truth provided by ELA with the results of various algorithms, resulting in the development of forgery detection methods that are more robust and accurate.
- Identification of regions susceptible to manipulation: BusterNet's primary goal is to precisely pinpoint potential image manipulation regions. BusterNet, in contrast to previous methods, employs a two-branch architecture that makes use of visual artifacts and similarities to identify both copy-move regions and potential manipulation regions. Enhancing the model's detection and localization capabilities is the goal of this objective, which will provide useful insights into the particular parts of an image that have been altered.
- Separation of the regions that are the source and the target: BusterNet's primary goal is to distinguish between copy-move forgeries' source and target regions. BusterNet is able to distinguish between the manipulated image's source (the original) and target (the cloned) regions by utilizing its distinctive architecture and fusion module. In forensic investigations, this objective is important because it enables investigators to distinguish between the original content and cloned portions, providing crucial evidence for comprehending the manipulation process.
- Pure trainable model from beginning to end: BusterNet aims to be a pure, trainable deep neural network solution from beginning to end, unlike previous methods. The goal is to foster a model that can be upgraded straightforwardly for the duplicate move imitation identification task, killing the requirement for complex pre-handling or post-handling steps. BusterNet is an efficient and effective method for detecting copy-move forgeries thanks to this objective, which makes the process of training and deploying the system easier.
- Large-scale CMFD sample synthesis: The BusterNet project also aims to develop straightforward methods for synthesizing large-scale copy-move forgery detection (CMFD) samples from out-of-domain datasets. By generating diverse and realistic CMFD samples, this objective hopes to overcome the issue of limited training data. BusterNet can be trained on a larger and more diverse dataset, enhancing its generalization and detection capabilities, by utilizing out-of-domain datasets and stage-wise strategies.

- Stability in the face of known threats: BusterNet's main goal is to show that it can withstand a variety of known attacks on image forgery detection algorithms. The goal is to see how BusterNet does in difficult situations where adversaries might use sophisticated methods to fool forgery detection models. By exhibiting its strength to known assaults, BusterNet means to give a dependable and reliable answer for identifying duplicate move falsifications in genuine situations.
- By tending to these targets, BusterNet intends to beat cutting edge duplicate move location calculations overwhelmingly. BusterNet is a promising option for accurate and dependable copy-move forgery detection due to its proposed deep neural architecture, its ability to localize source and target regions, and its resistance to known attacks.

1.4 Methodology

1.4.1 AlexNet

Image forgery detection using the AlexNet model can be done in the following way:

1. Training Data: Prepare the training data. This involves collecting a dataset of original and manipulated images. These images should be labeled as either "authentic" or "tampered".
2. Preprocessing: After obtaining the date, preprocessing is necessary. It can be done by resizing the images to a fixed size and normalizing the pixel values.
3. Training: The next step is to train the AlexNet model using the prepared dataset. The model should be trained on the "authentic" and "tampered" images to learn the features that distinguish the two.
4. Validation: After training, the model should be validated on a separate/same dataset. Thisdataset should contain both "original" and "manipulated" images. The model's performance should be evaluated using metrics such as accuracy, precision, recall, and F1score.
5. Inference: Finally, the trained model can be used to detect image forgeries. Given an input image, the model can classify it as "original" or "manipulated" based on the featuresit has learned.

Some additional tips for improving the performance of the AlexNet model for image forgery detection include:

- Using data augmentation techniques such as random cropping, flipping, and rotation to increase the size of the training dataset.
- Using transfer learning by fine-tuning a pre-trained AlexNet model on a related task, such as object recognition, before training it on the image forgery detection task.
- Using an ensemble of multiple AlexNet models to improve the overall performance.

1.4.2 Error Level Analysis (ELA)

- Identifying complex manipulations: The detection of sophisticated image manipulations will be improved by incorporating ELA into the method. ELA is able to reveal the variations in the compression artifacts that are introduced during the image editing process, shedding light on altered regions that may be difficult to spot using only deep learning techniques.
- An additional analysis: The goal is to use the advantages of both ELA and deep learning models like AlexNet to detect image forgeries with greater accuracy and

dependability. ELA can improve overall detection performance by providing additional information about tampering and image compression artifacts.

- Localization of the affected areas: The precise localization of the manipulated areas in an image is an important goal when using ELA. ELA can help with the targeted examination and analysis of the forged regions by identifying the areas with significant compression inconsistencies. This goal makes it possible to conduct in-depth forensic investigations and provides useful evidence in court.

1.4.3 BusterNet

The technique for utilizing BusterNet, a clever profound brain engineering for duplicate move falsification discovery, includes the accompanying advances:

- ★ Preparation of a Dataset: Accumulate a dataset containing a different scope of pictures with duplicate move frauds. Genuine images, manipulated images with copy-move regions, and ground truth annotations indicating the type and location of forgery ought to be included in this dataset.
- ★ Preparing BusterNet: Train the BusterNet model utilizing the arranged dataset. The architecture of the model is divided into two branches that focus on locating copy-move and potential manipulation regions through visual similarities. The information from the two branches is combined in the fusion module. Optimizing BusterNet specifically for the copy-move forgery detection task is the goal.
- ★ Synthesis of Data: Out-of-domain datasets can be used to easily synthesize large-scale copy-move forgery detection (CMFD) samples. Utilizing non-copy-move forgery datasets and stage-wise strategies to generate diverse and realistic CMFD samples are the two main components of this. The expansion of the training dataset, enhancement of the model's generalization capabilities, and enhancement of detection performance are the goals.
- ★ Evaluation of Performance: Examine BusterNet's performance against known attacks and publicly accessible datasets like CASIA and CoMoFoD. Assess BusterNet's superiority in terms of detection accuracy, localization of source/target regions, and robustness against various manipulation techniques by comparing its results to those of cutting-edge copy-move detection algorithms.
- ★ Certifiable Application: The goal is to make BusterNet a reliable and effective tool for finding copy-move forgeries, giving forensic analysts useful insights, and supporting legal proceedings.
- ★ BusterNet's goal is to outperform existing copy-move forgery detection algorithms and contribute to advancements in image forensics by employing this strategy.

Chapter 2: Related Work

Rao and Ni **[1]** proposed a deep learning approach for the detection of splicing and copy-move forgeries in images. They applied a max-pooling technique to the feature maps and designed a model with 8 convolutional layers, three pooling layers, and one fully-connected layer with a SoftMax classifier. Instead of random weight initialization, they used the spatial rich model (SRM) as a weight initialization method. However, one disadvantage of their framework was the use of rectified linear units (ReLU) as an activation function, which can be fragile during training and lead to suboptimal results.

Salloum et al **[2]** proposed a multi-task fully convolutional network (MFCN) for image splicing localization. They used an FCN architecture to locate the spliced regions in an image by classifying each pixel as spliced or authentic. The model consisted of two output branches, one for learning the labels and the other for learning the edges of the spliced regions. The intersection of the outputs from these two branches provided the localization result. They evaluated their model on datasets including Carvalho, CASIA v1.0, Columbia, and the NIST Nimble Challenge 2016, achieving a maximum F1 score of 0.6117 on the Columbia dataset.

Ouyang et al. **[3]** proposed a deep learning framework for copy-move forgery detection using the AlexNet structure without modifications. They applied the AlexNet model on the ImageNet database and evaluated its performance on the UCID, OXFORD flower, and CMFD datasets. The model demonstrated good performance in detecting forgery images generated automatically by computers through simple copy-move operations. However, it showed limitations in detecting copy-move forgery images in real scenarios. This work was the first implementation of AlexNet in forgery detection, proving its potential in the field.

Hao Gao **[4]** provides a detailed explanation of AlexNet, which works similarly to the pretrained model mentioned earlier in section 1. However, there are some differences in its implementation:

- ReLU Activation: ReLU is used instead of Tanh to introduce nonlinearity, resulting in faster computation speed while maintaining accuracy.

- Dropout Regularization: Dropout is used instead of regularization to mitigate overfitting. However, it increased the training time when a dropout rate of 0.5 was applied.

- Overlap Pooling: Overlap pooling is utilized to reduce the model's error rate, enhancing its performance.

In general, AlexNet consists of 5 convolutional layers and 3 fully connected layers. ReLU activation is applied after each convolutional and fully connected layer, while dropout is applied before the first and second fully connected layers. The modifications discussed in this article provide insights into optimizing the AlexNet architecture for improved performance.

A feature-based approach for copy-move forgery detection utilising the histogram of oriented gradients (HOG) descriptor was described in another study by Zhou et al. **[5]** The scientists used a clustering approach to find duplicated areas and retrieved HOG characteristics from overlapping blocks in a picture. Their method was successful in managing a variety of forgeries, including ones that underwent rotation and scale modifications.

A deep learning-based strategy for copy-move forgery detection using generative adversarial networks (GANs) was also put out by Li et al. **[6]** The scientists employed a GAN model that they trained to produce realistic textures to discriminate between unaltered and altered areas. Their technique demonstrated the promise of GANs in this area with remarkable accuracy and resilience against various sorts of forgeries.

A technique based on the scale-invariant feature transform (SIFT) was suggested for the detection of copy-move fraud in a paper by Amerini et al. **[7]** The authors found keypoints and descriptors in a picture by using SIFT features. They then compared the similarity of keypoints and used a matching method to identify duplicated sections. The accuracy and resilience of their method against various forgeries, such as those involving rotation, scale, and occlusion, produced encouraging results.

A block-based approach for copy-move forgery detection utilising the discrete cosine transform (DCT) was presented in another study by Bayram et al. **[8]** The authors determined the DCT coefficients for each block after dividing a picture into overlapping sections. The DCT coefficients were then compared using a hashing approach to find copied areas. Their method was successful in identifying copy-move forgeries even when dealing with noise and JPEG compression.

Additionally, a technique for copy-move forgery detection based on the dual-tree complex wavelet transform (DTCWT) was put out by Luo et al **[9]** To extract multi-scale and multi-directional characteristics from a picture, the scientists used the DTCWT. In order to identify duplicated regions based on the closeness of these traits, they next used a clustering technique. Their method has excellent success in identifying copy-move frauds, particularly in cases involving intricate transformations.

An approach for copy-move forgery detection utilising SURF (Speeded-Up Robust Features) and SIFT descriptors was described in another study by Bayram et al.**[10]** To find duplicated areas, the authors calculated the SURF and SIFT characteristics for each

block in a picture. Once they had located comparable places, they used a matching technique to identify the faked parts. In terms of accuracy and resilience against various sorts of changes, their method demonstrated encouraging results.

The paper of Fridrich et al. **[11]** which suggested an approach based on the identification of duplicated areas using the Discrete Cosine Transform (DCT), is considered one of the important studies in this field. In their method, the picture was divided into overlapping blocks, and the DCT coefficients were determined for each block. They were able to spot locations that had a lot of similarities, perhaps signifying copy-move forgeries, by comparing these coefficients. They had encouraging findings, but their approach was only applicable to uncompressed photos, and it had trouble spotting forgeries in pictures with low contrast or intricate backgrounds. Later investigations looked into different feature extraction methods and improved the detection algorithm's resilience in an effort to overcome these limitations.

The work of Bayram et al. **[12]** who suggested a technique based on the Scale-Invariant Feature Transform (SIFT) for copy-move forgery detection, is another important addition to this topic. In order to find duplicated sections, their method involves extracting recognisable keypoints from the image and comparing them. Their technique exhibited enhanced accuracy and resilience against various picture alterations by taking into account both local and global properties. The identification of copy-move forgeries in colour photographs or films was not examined in their investigation, which was largely focused on grayscale images. Later studies tried to broaden their approach to handle other picture formats and look into its relevance to video forensics.

Deep learning-based techniques have attracted a lot of attention recently in the area of copy-move forgery detection. For instance, a convolutional neural network (CNN) architecture created especially for this job was proposed by Zhang et al. **[13]** Their approach includes teaching the CNN discriminative features for forgery detection using a sizable dataset of altered and real pictures. In comparison to conventional feature-based approaches, the findings showed higher performance, especially in terms of accuracy and computing economy. Deep learning models' interpretability is still a problem, though, because for forensic investigation, knowing how these models make decisions is essential. The development of understandable deep learning models for copy-move forgery detection may be the subject of future study.

Additionally, Li et al. **[14]** provided a unique method for detecting copy-move forgeries based on the examination of picture noise patterns. To find duplicated locations, they used an approach that entailed removing noise remnants from the picture and comparing them. Their method showed better accuracy and resilience against various picture alterations by taking into account both the spatial and frequency domains. The identification of forgeries in photographs collected from other sources, such as cellphones

or security cameras, was not explored in their study, which was largely focused on images taken by digital cameras. Future studies might look at how well their technique applies to other imaging settings and equipment.

Local binary patterns (LBP) **[15]** were used in 2010 to build a block-based matching method for locating copy-move frauds. Although they had trouble with rotation and scale adjustments, their system was quite accurate in spotting counterfeit parts.

Scale-invariant feature transform (SIFT) **[16]** descriptors are used in (2012) to describe a keypoint-based matching method. Although their system showed resilience against different forgeries, it suffered from increasing computational complexity.

Lee et al **[17]** suggested a deep learning method using convolutional neural networks (CNNs) for the identification of copy-move forgeries. On huge datasets, their model performed at the cutting edge, but it used a lot of computer power.

A approach based on local phase quantization (LPQ) was suggested for the detection of copy-move fraud in a paper by Wang et al.**[18]** Local image structures are captured by the texture descriptor LPQ. To find duplicated regions in photos, the authors used this descriptor coupled with a block-matching technique. Even with complicated modifications, their method demonstrated promising accuracy and computing efficiency outcomes.

In a summary, the research on the detection of copy-move forgery in digital images includes a wide range of methods and strategies. A few key exploration papers have made huge commitments to this field, investigating various procedures and calculations to work on the exactness and power of identification techniques.

A few examinations zeroed in on highlight based approaches, for example, using neighborhood twofold examples (LBP), speeded-up powerful elements (SURF), scale-invariant component change (Filter), histogram of situated slopes (Hoard), and nearby stage quantization (LPQ). These strategies extricated unmistakable highlights from the picture and utilized matching calculations or bunching methods to distinguish copied areas. Although these methods were effective at detecting copy-move forgeries, they frequently encountered difficulties with regions with low contrast, image noise, or complex geometric transformations.

Advanced methods like deep learning and generative adversarial networks (GANs) were used in other studies. These methodologies used convolutional brain organizations (CNNs) and GAN models to learn and recognize unique and altered areas. They required a lot of computational power and a lot of training data, but they were very accurate and resistant to a variety of forgeries.

To improve detection efficiency, a number of papers combined various methods. For example, a few examinations incorporated include extraction strategies like Filter, SURF, LBP, or LPQ with change-based procedures, for example, discrete cosine change (DCT), particular worth deterioration (SVD), or discrete wavelet change (DWT). The goal of these hybrid strategies was to better detect copy-move forgeries by combining the advantages of various methods.

While a considerable lot of the proposed strategies showed promising outcomes, the impediments shifted across the examinations. Complex transformations, occlusion, and overlapping objects were challenges for some methods, while image noise and parameter tuning were issues for others. Computational intricacy and asset prerequisites were additionally normal worries for techniques using profound learning or enormous scope datasets.

Overall, the research on copy-move forgery detection shows that the field is always getting better. Researchers are looking into different ways to make detection methods more accurate, reliable, and effective. With the potential to improve the security and integrity of visual content in a variety of applications, these studies have laid the groundwork for additional digital image forensics research and development.

Chapter 3: Proposed models

3.1 AlexNet Proposed Model

Based on the design of AlexNet, we present two models: AlexNet was selected as the foundation of the proposed model because its quick network training speed and ability to reduce overfitting. Moreover, the AlexNet model's deep structure, simplicity, quick training time, and low memory usage make it appropriate for the interpretation of faked images.

3.1.1 Pre-trained AlexNet:

In this model we are using Fine-tuned Convolutional Neural Networks for Forgery Detection in Digital Images. Our approach is to use a pretrained AlexNet model to extract features from digital images and then fine-tune the model on a dataset of forged and authentic images. The fine-tuned model is then used to classify new images as authentic or forged. We used 80% of the data for training and 20% of the data for testing. After loading the data, we froze the parameters of the model then we changed the last layer of the model for transfer learning. We use the Negative Log Likelihood Loss function and the Adaptive Moment Estimation "ADAM" as an optimizer.

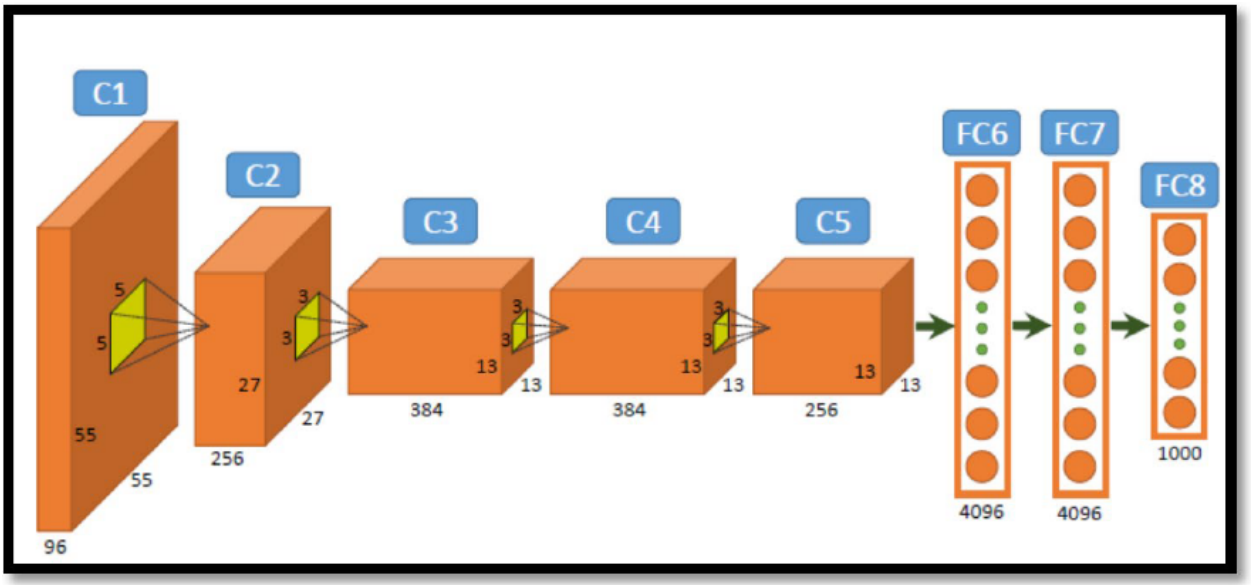


Figure 2: Pre-trained Model

3.1.2 Modified AlexNet:

The proposed system **[21]** is a modified version of the AlexNet deep neural network, which is a well-known convolutional neural network (CNN) architecture. The architecture of the proposed system is based on AlexNet and consists of multiple convolutional and fully connected layers:

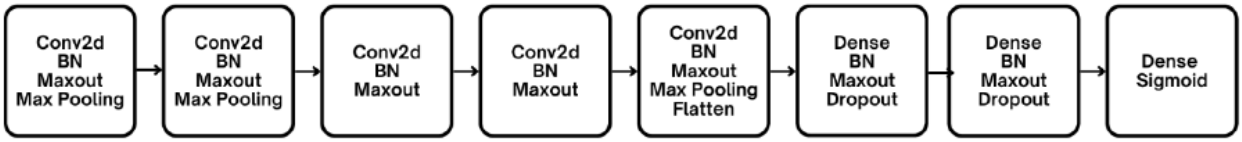


Figure 3: Modified Model

The input to the system is an image with size 227 x 227 x 3, which represents the height, width, and number of color channels of the image, respectively. The first layer of the system is a convolutional layer with 96 filters and a kernel size of 11 x 11. The stride of this layer is set to 4 x 4, which means that the filters will move 4 pixels horizontally and 4 pixels vertically in each iteration. The next layer is batch normalization layer, which helps to speed up the convergence of the network in addition, it is used throughout the model to normalize the activations of each layer and reduce the risk of overfitting. Then, a Maxout layer, which is a type of activation function that helps to improve the performance of the network. One key modification to the original AlexNet architecture is the use of Maxout activation functions in place of traditional activation functions such as ReLU. Maxout functions have been shown to improve the performance of deep learning models by allowing the model to learn a wider range of features. Finally, there is a max pooling layer with a pool size of 3 x 3 and a stride of 2 x 2, which reduces the spatial dimensions of the feature maps. by reducing the number of parameters in the network, making it more computationally efficient and easier to train.

The next two layers, C2 and C3, are similar to the first layer and consist of convolutional, Maxout, batch normalization, and max pooling layers. However, the number of filters in each layer is increased to 256 and 384, respectively. Except C3 has no max pooling layer. These layers are designed to extract increasingly complex features from the input image. The next two layers, C4 and C5, are also convolutional layers with 384 filters and 256 filters, respectively. C4 has no max pooling layer as C3, however, C5 has a max pooling layer with a pool size of 3 x 3 and a stride of 2 x 2.

After the final pooling layer, the features are flattened into a 1D vector and passed through two fully connected layers with 4096 neurons each. These layers are followed by Maxout layers and dropout layers with a dropout rate of 0.5 as well as batch normalization. Dropout is a regularization technique that helps to prevent overfitting by randomly dropping out neurons during training. The final layer is a fully connected layer

with 2 neurons and a sigmoid activation function, which outputs the probabilities of the image being authentic or forged.

The proposed system is compiled using the binary cross-entropy loss function and the stochastic gradient descent (SGD) optimizer. The accuracy of the network is evaluated using the accuracy metric. The system is trained using a large dataset of authentic and forged images and the weights are updated using backpropagation to minimize the loss function.

In conclusion, the proposed system is a modified version of AlexNet that has been designed for the task of image forgery detection. The system consists of multiple convolutional and fully connected layers that extract features from the input image and use these features to make a prediction about whether the image is authentic or forged. The system has been designed to handle large amounts of data and can be trained using a standard deep learning framework. The system has the potential to provide an effective solution to the problem of image forgery 1detection and can be applied in a wide range of applications.

K-Fold Cross-Validation:

K-Fold Cross-Validation is a commonly used model evaluation technique in machine learning, used to evaluate the performance of a model on a given dataset. In this technique, the dataset is split into "k" folds, with each fold being used as a validation set once and the remaining k-1 folds being used as the training set. The model is trained on the training set and evaluated on the validation set, and this process is repeated k times, with each fold being used as the validation set once.

The purpose of using K-Fold Cross-Validation is to prevent overfitting, which is a common problem in machine learning models, where the model learns the patterns in the training data too well and performs poorly on new, unseen data. Using K-Fold Cross-Validation helps to overcome this problem by using multiple validation sets, which ensures that the model is exposed to a wide range of different data, and not just a single validation set.

In our proposed system, we have used K-Fold Cross-Validation to evaluate the performance of the modified AlexNet architecture. We have split the dataset into 10 folds, and used 9 folds for training and 1-fold for validation. This has allowed us to obtain a more accurate estimate of the model's performance, as it has been exposed to a wide range of different data. This has also allowed us to tune the hyperparameters of the model, such as the number of filters and kernel size, to achieve optimal performance on the validation set. The results obtained from K-Fold Cross-Validation have been used to evaluate the performance of our proposed system and to compare it with other existing models.

3.1.3. AlexNet with ELA

Error Level Analysis

Error level analysis is one technique for knowing images that have been manipulated by storing images at a certain quality level and then calculating the difference from the compression level. When JPEG was first saved, then it will compress the image the first time, most editing software like adobe photoshop, gimp, and adobe lightroom support JPEG compressing operation. If the image is rescheduled using image editing software, then compressed again. So it shows that the original image when the first image is taken using a digital camera has been compressed twice, first use the camera and the second is editing software. When viewed with the naked eye the image looks the same, but by using this method it will look the difference between a forgery image with the original image. Calculation for the average difference of the quantization table Y (luminance) and CrCb (Chrominance). The digital camera does not optimize the image for a specified camera quality level (high, medium, low, etc.). Original images from digital cameras should have high ELA values. Each subsequent resave will decrease the potential error rate. Original images from photography have high ELA values shown through white on the ELA image, as shown in Figure 4. When the image is resaved, using ordinary human vision does not show a significant degree of difference, but ELA shows the dominant black and dark colors. If this image is resaved again it will decrease the image quality. If the original image is then modified, ELA will show the modified area has a color with a higher ELA level. The Figure 4 describes how the output of ELA on the condition of the image.

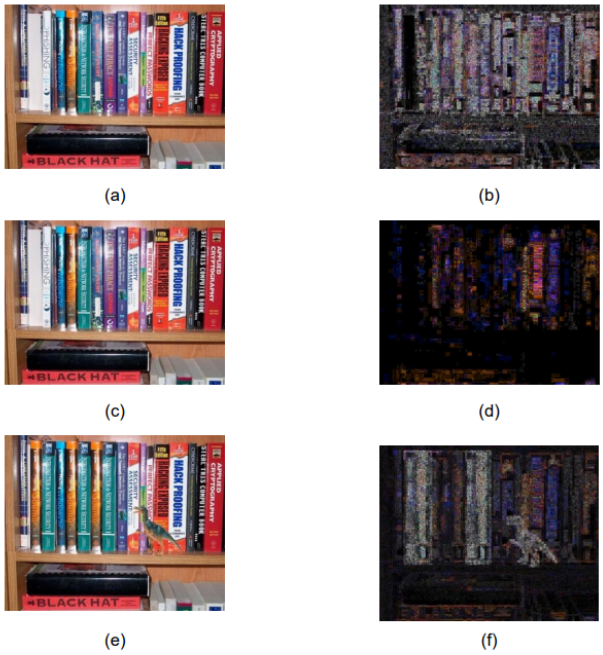


Figure 4: Error level analysis compression: (a) original image, (b) ELA original Image, (c) resave image, (d) ELA resave image, (e) tampered image, (d) ELA tampered image

Fitting the training data with modified AlexNet

Epochs	loss	accuracy	val_loss	val_accuracy
1	0.45	0.8329	0.3324	0.88
2	0.36	0.85	0.27	0.89
3	0.33	0.86	0.27	0.89
4	0.3035	0.882	1.38	0.70
5	0.27	0.88	0.58	0.83

Table 1: Fit of AlexNet-ELA

3.2 VGG-16

The VGG (Visual Geometry Group) **[23]** architecture, originally developed for image classification, can also be applied to the task of image forgery detection. VGG is known for its deep structure and strong feature extraction capabilities, making it a suitable choice for analyzing and identifying manipulated regions within images.

The VGG architecture consists of multiple convolutional layers stacked together, with each layer responsible for extracting increasingly complex features from the input image. The network's structure is characterized by its simplicity and uniformity, with smaller filter sizes (3x3) and a deeper stack of layers compared to earlier architectures like AlexNet.

To utilize VGG for image forgery detection, a training dataset is required, consisting of both authentic images and images with known manipulations. The VGG network is then trained using a supervised learning approach, where the network learns to differentiate between genuine and manipulated images.

During training, the input images are passed through the network, and the feature maps are extracted from various layers. These feature maps capture different levels of visual information, ranging from low-level details like edges and textures to high-level semantic features. By examining the patterns and variations within these feature maps, VGG can learn to identify discrepancies and irregularities caused by image manipulation.

After training, the VGG model can be used for detecting image forgery in unseen images. When presented with an input image, the network processes it through its layers and compares the extracted features to the learned representations. Deviations from the expected patterns, such as inconsistent textures, unusual artifacts, or unnatural boundaries, can indicate the presence of forgery.

The VGG-16 architecture (figure below) consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. The first 13 convolutional layers have filters of size 3x3, with a stride of 1 and padding of 1. The max-pooling layers are applied after the second, fourth, eighth, and thirteenth convolutional layers, with a pool size of 2x2 and a stride of 2. The final three fully connected layers each have 4096 units.

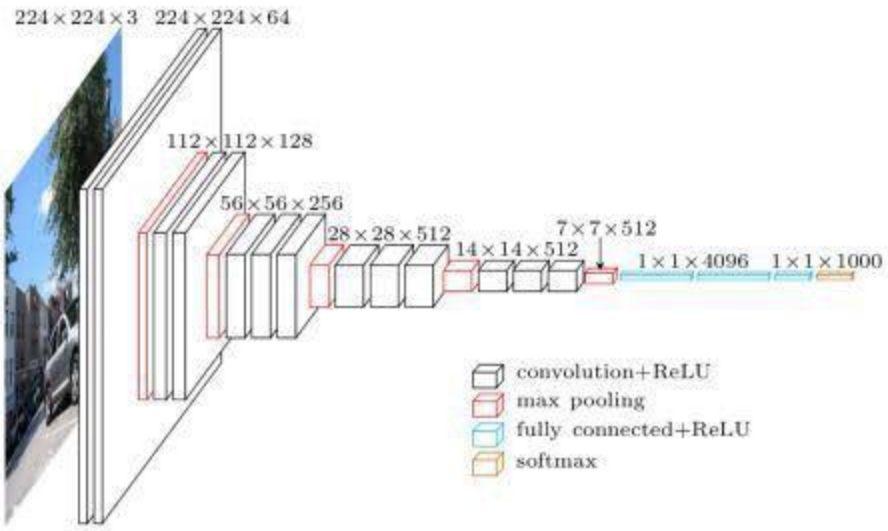


Figure 5: VGG Architecture

The model was trained on the large-scale ImageNet dataset, which contains over a million images and 1000 categories. By using the pre-trained weights, we were able to save a significant amount of time and computational resources in training our own model.

Training accuracy of the model achieved up to 95.6% and for validation 89.75% using 10 epoch. Thus, by using the deep learning architecture of VGG 16 in analyzing error level image analysis for image forgery can be applied and get good results on recognition

Proposed Approach - VGG-ELA:

The first step we took was to divide the dataset from Casia V.2 into 2 categories: original and fake images. We normalize the image by processing the image to a size of 224x224 pixels. Then our next step is to perform analysis on the level of compression error image, from the compression result then we use the VGG 16 architecture for CNN in recognizing the original image and fake images according to the ELA , where we fine tune the VGG-16 with 80% of CASIA V2 and we used the 20% for validation. Our proposed method described on flowchart as shown in Figure below.

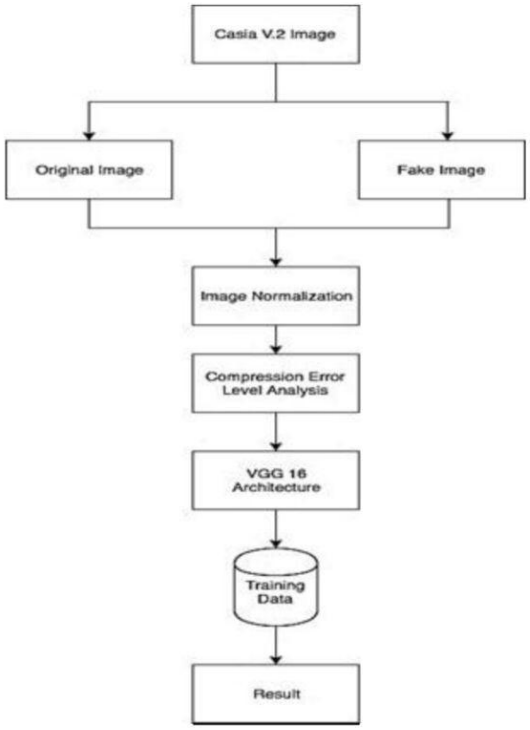


Figure 6: VGG-ELA

3.3. Buster Net Proposed Model

Buster Net model **[22]** is a novel deep neural architecture for image copy-move forgery detection (CMFD). Unlike any other model, BusterNet is a pure end-to-end trainable deep neural network solution. It features a two-branch architecture followed by a fusion module. The two branches localize potential manipulation regions via visual artifacts and copy-move regions via visual similarities, respectively. This model will be responsible for localizing the source/target regions of the tampered image. The model was trained on a hundred thousand quality synthetic samples for copy-move detection. So, our main concern was evaluating tampered/authentic test samples on two approaches and finding the best dataset that contains various attacks to get the best evaluation.

3.3.1. Buster Net Architecture

To achieve the goals of localizing the source/target destination, a valid DNN solution should attain two feature properties simultaneously: source/target features that are dissimilar enough to be distinguished from one another, and how similar than those features can be in pristine regions. The best idea to achieve these properties is to explicitly consider them by a two-branch DNN architecture as shown in the below figure:

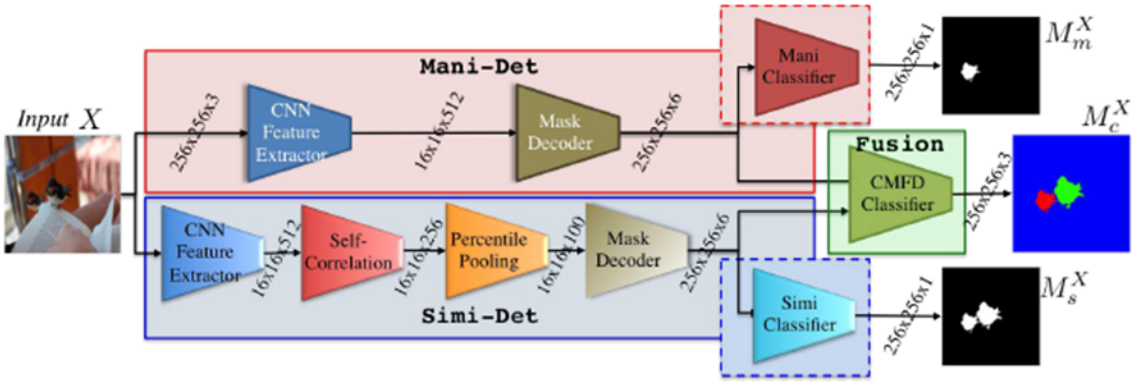


Figure 7: BusterNet for Copy-Move Forgery Detection

The Mani-Det(manipulation) branch is used to detect manipulated regions such that its feature is good for the first property, while the Simi-Det (similarity) branch is used to detect cloned regions such that its features attain the second property. Finally, both features will be merged in Fusion to predict pixel-level copy-move masks differentiating pristine, source copy, and target copy classes. The assumed input image will be of size 256×256×3, however, BusterNet is capable of handling images of other sizes.

3.3.1.1. Manipulation Detection Branch

The manipulation detection branch (i.e., Mani-Det as shown by red shaded regions in figure 1) can be thought of as a special segmentation network whose aim is to segment manipulated regions. More precisely it functions by the following steps:

- It takes input image
- Extracts features using CNN Feature Extractor
- Upsamples the feature maps to the original image size using Mask Decoder
- Applies Binary Classifier to produce the manipulation mask.

Any convolutional neural network (CNN) can serve as CNN Feature Extractor.

Here, the first four blocks of the VGG16 architecture were used due to its simplicity. The resulting CNN feature is of size 16×16×512, whose resolution is much lower than that is required by the manipulation mask. Due to the resolution problem, we need to decode this feature, and apply deconvolution to restore the original resolution via the Mask Decoder shown figure 5. This decoder applies BN-Inception and BilinearUpPool2D in an alternating to produce a tensor of shape 256×256×6. For further clarification, 16 times of the spatial dimension increase is due to the 4 times of BilinearUpPool2D (i.e. 2^4=16), and the output filter dimension 6 is because of the last BN-Inception(2@[5,7,11]), which concatenates 3 Conv2D responses, each with 2 output filters while using kernel sizes of: [(5,5), (7,7), (11,11)] respectively (i.e. 3×2=6). Finally, we predict pixel-level manipulation mask with a binary classifier, which is as simple as a single Conv2D layer with 1 filters of kernel size (3,3) followed by the sigmoid activation.

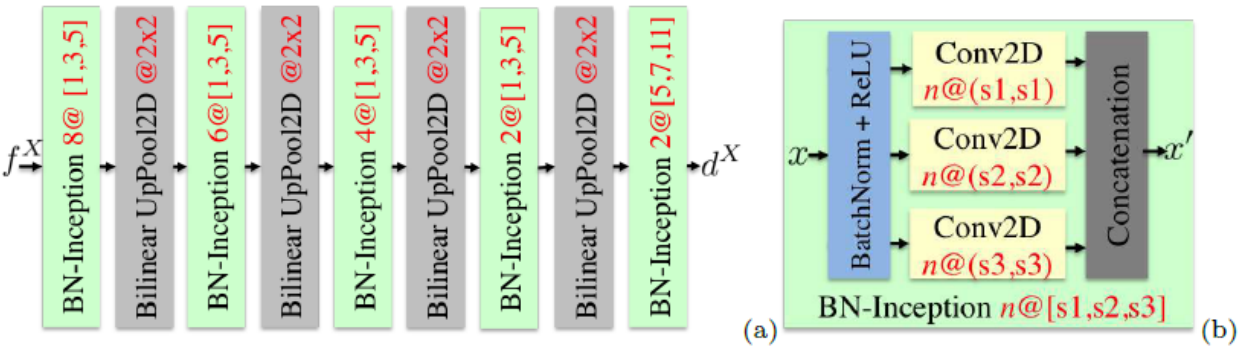


Figure 8: (a) Mask deconvolution network

Figure 8: (b) parametric BN-inception module, where s1, s2 and s3 indicates the kernel sizes

3.3.1.2. Similarity Detection branch

The similarity detection branch (i.e., Simi-Det as shown by blue shaded regions in figure 4) where it functions by the following steps:

- Takes an input image.
- Extracts features using CNN Feature Extractor
- Computes feature similarity via Self-Correlation module.
- Collects useful statistics via Percentile Pooling
- Upsamples feature maps to the original image size using Mask Decoder
- Applies Binary Classifier to produce a copy-move mask at the same resolution of original input image.

It can be clearly shown that the two branches share a common CNN Feature Extractor but only its architecture not weights.

Like the Mani-Det branch, Simi-Det branch starts with feature representation via CNN Feature Extractor. It again produces a feature tensor of size 16×16×512, which can be also viewed as 16×16 patch-like features each with 512 dimensions. Since our goal is to recover the potential copy-move regions, we must mine useful information to decide what are matched patch-like features.

To do so, first compute all to all feature similarity score using Self-Correlation and collect meaningful statistics to identify matched patches via Percentile Pooling. Specifically, given two patch-like features. We will use the Pearson correlation coefficient to quantify the feature similarity as shown in equation 1.

$$\rho(i,j) = (\tilde{f}_m^X[i])^T \tilde{f}_m^X[j]/512$$

Equation 1 Pearson Correlation Coefficient

where $(\cdot)^T$ is the transpose operator, and $\tilde{f}_m{}^X[i]$ is the normalized version of $f_m{}^X[i]$ by subtracting its mean $u_m{}^X[i]$ and dividing by its standard deviation $\sigma_m{}^X[i]$ as shown in equation 2.

$$\tilde{f}_m^X[i] = (f_m^X[i] - \mu_m^X[i])/\sigma_m^X[i]$$

Equation 2

For a given $f_m{}^X[i]$, we repeat the process over all possible $f_m{}^X[j]$, and form a score vector $S^X[i]$ namely,

$$S^X[i] \models [\rho(i,0), \cdots, \rho(i,j), \cdots, \rho(i,255)]$$

Equation 3

As a result, Self-Correlation outputs a tensor S^X of shape 16×16×256

When ${f_m}^X$ and Pearson correlation coefficient are both meaningful, if ${f_m}^X[i]$ is matched, some score $S^X[i][j]$ with $j \neq i$ should be significantly greater than the rest of the scores $S^X[i][k]$ with $k \in \{i, j\}$. Since the corresponding ${f_m}^X[j]$ is unknown it is difficult to check this pattern in the context of DNN. Alternatively, it is easier to check this pattern in a sorted score vector. Specifically, Percentile Pooling first sorts a score vector $S^X[i]$ to $S'^X[i]$ in the descending order as shown equation 4.

$$S'^X[i] = \text{sort}(S^X[i])$$

Equation 4

One can directly feed S'^X to future modules to decide matched features. However, one drawback of doing so is that the resulting network loses the capability of accepting an input of an arbitrary size, because the length of score vector is dependent on the input size. To remove this dependency, Percentile Pooling also standardizes the sorted score vector by only picking those scores at percentile ranks of interests. In other words, regardless the length L of raw sorted score vector, we always pick K scores to form a pooled percentile score vector $P^X[i]$ as shown in equation 5.

$$P^X[i][k] = S'^X[i][k']$$

Equation 5

where $k \in [0 \ldots, K-1]$ and k' is the index of raw sorted score vector after mapping a predefined percentile $p_k \in [0.1]$ according to L as shown in equation 6

$$k' = \text{round}(p_k \cdot (L-1))$$

Equation 6

Another advantage of the above standardization is dimension reduction, because only a small portion of all scores is kept. Once Percentile Pooling is done, we use Mask Decoder to gradually upsample feature to the original image size. Then, Binary Classifier is used to produce a copy-move mask to fulfill the auxiliary task. Again, both Mask Decoder and Binary Classifier only have the same architecture as those in Mani-Det, but with distinctive weights.

3.3.1.3. BusterNet Fusion

As illustrated in figure 7, fusion module takes inputs of the Mask Decoder features from both branches, and jointly considers these two branches and make the final CMFD prediction. More precisely by the following steps:

- Concatenate features coming out from mani-det branch and simi-det branch
- Fuse these features using the BN-Inception with parameter set 3@[1,3,5] (fig 5b)
- Predict the three-class CMFD mask using a Conv2D with one filter of kernel size 3×3 followed by the softmax activation.

3.3.2 Buster Net Implementation and Training

3.3.2.1. Custom Layer Implementation

Except Self-Correlation and Percentile Pooling modules, all other modules are either standard or can be built from standard layers. Self-Correlation requires implementing equations 1 and 2 (differentiable equations). Percentile Pooling is essentially a pooling layer, which has no trainable parameters but a deterministic pooling function. As one may notice, neither equation 4 nor 5 is differentiable. However, all what is needed to do is to perform backpropagation similar to that is performed in standard MaxPooling (i.e., only the neuron corresponding to the max receives the gradient).

3.3.2.2. Training Details

Publicly available datasets are very small (typically around a few thousand) that covers the copy-move forgery. More importantly, none of the existing CMFD dataset provides ground truth masks differentiating source and target copies. Therefore, the paper created a synthetic dataset for training Buster Net. Where it started with an original image X with an associated object mask, randomly generates an affine matrix m to transform both the source mask and the source object image. Then, use the transformed mask as the target mask, paste the transformed object back to image X and obtain a copy-move forgery sample X. The datasets used for training were the MIT SUN2012 dataset **[19]**, and the Microsoft COCO dataset **[20]** as image sources. To further encourage more real-like training samples, it trained a binary classifier to predict whether a sample is real unmanipulated or synthetic copy-move forged. Synthetic samples that fail to fool this classifier are not used for training Buster Net. Figure 3 shows some synthetic samples of the datasets used for training.

Figure 9: Synthetic samples of the used datasets

In total, there were hundred thousand quality synthetic samples for copy-move detection, each with one three-class mask distinguishing source and destination copies and two binary masks auxiliary task training. The synthetic training data is split into training, validation, and testing splits with 8:1:1 ratio. This synthetic dataset is used to train both auxiliary tasks and the main task of Buster Net. There were also the usage of external image manipulation dataset from IEEE IFS-TC First Image Forensics Challenge and the Wild Web tampered image dataset to train Mani-Det branch, because the Mani-Det must learn to identify more manipulated regions beyond the fixed set of manipulations in the synthetic dataset. To train Buster Net, weights were initialized randomly except for the pretrained VGG16, where its weights was the ImageNet to be used in CNN Feature Extractor in Simi-Det. Instead of training Buster Net all modules together, they adopted a three-stage training strategy:

1. Train each branch with its auxiliary task independently.
2. Freeze both branches and train fusion modules.
3. Unfreeze the entire network and fine tune Buster Net end-to-end.

Specifically, for auxiliary tasks, they used Adam optimizer with initial learning rate of 1e-2 and binary cross entropy loss. Whenever validation loss reaches plateaus after 10 epochs, the learning rate was reduced by half until improvement stops for 20 epochs. For the main task, Adam optimizer with categorical cross entropy loss was used, however. The initial learning rate was 1e-2 for fusion training while 1e-5 for fine tuning.

Metrics / Class	Recall			Accuracy
	Pristine	Source	Target	3-Class
Simi-Det Only	92.57%	32.28%	38.97%	92.57%
Direct BusterNet	93.70%	34.12%	47.37%	92.74%
Stage-wise BusterNet	93.83%	41.64%	53.61%	93.02%

Table 2: Different training strategies on Synthetic 10K testing set

3.3.3 Types of BusterNet Evaluation

3.3.3.1. Metrics and Baseline Settings

We use precision, recall and F1 scores to report CMFD performance.

Pixel-level Evaluation

For a testing image, we compute the true positive (TP), false positive (FP) and false negative (FN) at pixel level. Of course, we must treat pixels classified to source and target both as forged, so that the proposed Buster Net could be fairly compared with all classic CMFD methods which only predict binary masks. Based on how F1 is calculated, the protocol that was used for pixel-level evaluation is:

- compute precision, recall, F1 scores for each image, and report the average scores.

This protocol only works for a subset of forged images, but better quantifies the localization performance.

Image-level Evaluation

Another evaluation method is that if any pixels in a testing image are detected as forged, the testing image is labeled as forged. We compare a predicted image label with its ground truth to compute image-level TP, FP, and FN, and report precision, recall and F1 scores over an entire dataset as image-level evaluation protocol. We used the minimum area to be detected to evaluate at 3 different window sizes.

- 4*4 window size
- 8*8 window size
- 16*16 window size

Chapter 4: Datasets

4.1. Casia V1 Dataset

A well-known image forgery detection dataset is the Casia V1 dataset, which was created by the Institute of Automation at the Chinese Academy of Sciences. It contains 1881 images that fall into three main categories: normal images, copy-move forgeries, and splicing forgeries. The dataset includes images that have been altered by copying and pasting a specific portion of the image into another location within the same image. This type of forgery is known as copy-move forgery. The goal of this kind of forgery is to duplicate or duplicate parts of the image.

Images that have been altered by combining two or more images to create a new composite image fall under the category of splicing forgery. Often with the intention of deceiving viewers, this method involves seamlessly merging various elements or objects from multiple sources into a single image.

The typical picture classification comprises pictures that were gathered from different sources, including individual photograph assortments and online picture vaults. These images can be used as a guide to authentic, unaltered content.

The largest images in the Casia V1 dataset have a resolution of 256 x 384 pixels, indicating a range of sizes and resolutions. Ground truth information about the location and kind of forgery in each image is also provided by the dataset. This ground truth explanation works with the assessment and benchmarking of picture fabrication discovery calculations. The Casia V1 dataset is widely used by practitioners and researchers to evaluate the performance and efficacy of image forgery detection methods. It is a valuable resource for developing and benchmarking cutting-edge forgery detection algorithms because of its extensive collection of forgery types and realistic image compositions.

4.2. CoMoFoD Dataset

With 200 image sets, the CoMoFoD (Complex Morphology Dataset for Forensic Analysis) dataset provides a comprehensive collection of fake images. It was made to handle a wide range of difficulties and complexities that are frequently encountered in image forgery detection tasks. The CoMoFoD dataset contains an original image, colored and binary masks, and a fake image with copied regions in each image set. Translation, rotation, scaling, distortion, and combinations of these transformations take place in the copied regions of the forged images. These transformations are based on actual situations in which forgeries might occur.

Post-processing techniques have been applied to both the original and forged images in order to further enhance the dataset's complexity and realism. JPEG compression, noise addition, image blurring, brightness adjustments, color reduction, and contrast adjustments are among these post-processing methods. These changes imitate the alterations that are regularly applied to pictures before they are shared or distributed.

Based on the size of the images, there are two categories in the CoMoFoD dataset: a little picture class with aspects of 512x512 pixels and a huge picture classification with aspects of 3000x2000 pixels. Forgery detection algorithms can be evaluated using images of varying resolutions and complexity thanks to this.

The dataset fills in as an important asset for testing and benchmarking picture falsification discovery calculations. The CoMoFoD dataset is a valuable resource for advancing the field of image forensics due to its realistic and diverse collection of fake images, as well as the transformations and post-processing methods used.

4.3. MICC-F220 Dataset

A comprehensive dataset for image forgery analysis is the MICC-F220 dataset, which was created by the Media Integration and Communication Center (MICC) at the University of Florence, Italy. It has 220 image sets, each with one or more manipulated images and an original image.

There are four primary types of forgeries included in the dataset: removal, retouching, splicing, copy-move, and removal. To create clones or duplicates, copy-move forgeries involve copying and pasting a specific area of the genuine image into the same image. When two or more images are combined to form a composite image, splicing forgeries occur. Retouching forgeries include alterations to the image's appearance such as cropping, resizing, rotating, and changing the color. Removal forgeries involve removing specific objects or regions from the genuine image.

There are two subsets of the MICC-F220 dataset: a test set of 70 image sets and a training set of 150 images. Forgery detection algorithms can be trained and evaluated on various subsets of the dataset by researchers thanks to this division. The dataset includes masks that show where the forgeries in the manipulated images are to help with evaluation. These veils work with the appraisal of recognition calculations' presentation in precisely distinguishing the controlled areas. Researchers have widely used the MICC-F220 dataset as a benchmark for evaluating image forgery detection algorithms. Its different scope of falsification types, joined by ground truth explanations, make it a significant asset for testing and propelling the cutting edge in picture fraud recognition.

4.4. Casia V2 Dataset

CASIA v2 is a widely used dataset in the field of digital image forensics, specifically for copy-move and splicing image forgery detection. It is an extension of the original CASIA dataset, developed by the Chinese Academy of Sciences' Institute of Automation. CASIA v2 provides a valuable resource for researchers and practitioners to evaluate and develop algorithms and techniques for detecting image manipulation.

The CASIA v2 dataset consists of a large collection of digital images that have been artificially manipulated to contain copy-move and splicing forgeries. These forgeries are created by copying a region of an image and pasting it into another location within the same image or a different image, thereby creating duplicate or manipulated regions. The dataset encompasses various challenging scenarios and realistic visual aspects encountered in real-world image forgeries.

Key features of the CASIA v2 dataset include:

- ➢ Size and diversity: CASIA v2 comprises thousands of authentic and manipulated images with different resolutions, subjects, scenes, and lighting conditions. The dataset covers a wide range of object categories, making it suitable for diverse experimentation.
- ➢ Ground truth annotations: Each image in the dataset is provided with corresponding ground truth annotations that indicate the specific regions where copy-move and splicing forgeries are present. These annotations serve as references for evaluating the accuracy and effectiveness of forgery detection algorithms.
- ➢ Evaluation metrics: CASIA v2 facilitates the use of standard evaluation metrics, such as detection rates, false positive rates, precision, recall, and F1 scores. These metrics allow researchers to quantitatively assess the performance of their algorithms and compare them with other approaches.

Researchers and practitioners working on copy-move and splicing image forgery detection can leverage the CASIA v2 dataset to develop and validate their algorithms. By training and evaluating models on this dataset, they can improve the accuracy and robustness of forgery detection techniques and contribute to advancements in the field of digital image forensics.

Chapter 5: Result Analysis

5.1. Evaluation

5.1.1. Modified AlexNet

CASIA V.1

Average Accuracy:	*0.971*
Average Precision:	*0.972*
Average Recall:	*0.974*
Average F1 Score:	*0.973*

Table 3: CASIA V.1 -averages

MICC-F220

Average Accuracy:	*0.986*
Average Precision:	*0.986*
Average Recall:	*0.988*
Average F1 Score:	*0.986*

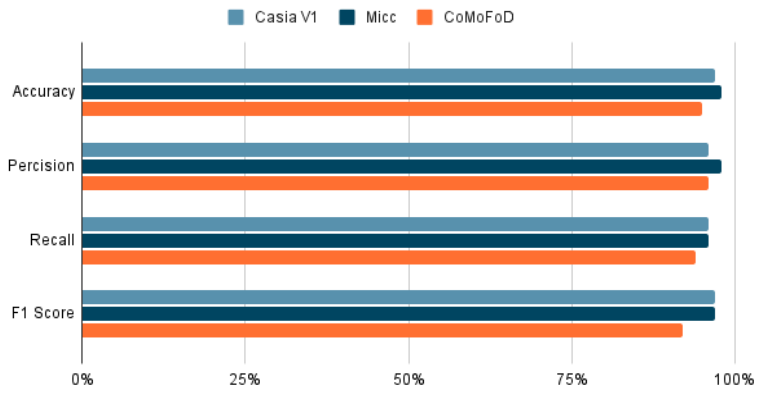
Table 4: MICC-F220 -averages

CoMoFoD

Average Accuracy:	*0.955*
Average Precision:	*0.956*
Average Recall:	*0.954*
Average F1 Score:	*0.955*

Table 5: CoMoFoD -averages

Figure 10: Modified AlexNet



5.1.2. Pre-Trained AlexNet

CASIA V.1

Average Accuracy:	*0.63*
Average Precision:	*0.64*
Average Recall:	*0.67*
Average F1 Score:	*0.66*

Table 6: CASIA V.1 -averages

MIC-F220

Average Accuracy:	*0.94049*
Average Precision:	*0.94144*
Average Recall:	*0.94364*
Average F1 Score:	*0.94238*

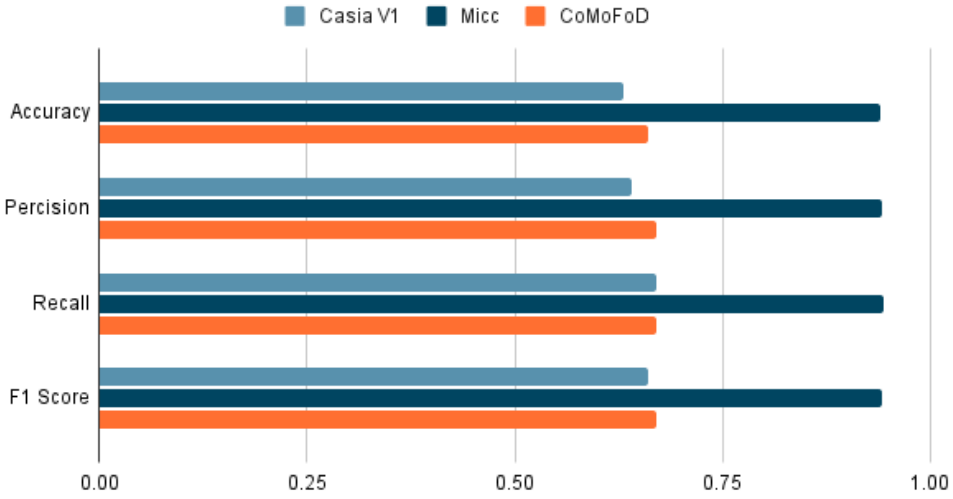
Table 7: MICC-F220 -averages

CoMoFoD

Average Accuracy:	*0.65961*
Average Precision:	*0.66442*
Average Recall:	*0.66633*
Average F1 Score:	*0.66973*

Table 8: CoMoFoD -averages

Figure 11: Pre-Trained AlexNet



5.1.3. VGG

CASIA V.1

Average AccuracY	*0.71539*
Average Precision	*0.75225*
Average Recall	*0.69151*
Average F1 Score	*0.7294*

Table 9: CASIA V.1 -averages

MIC-F220

Average Accuracy	*0.55241*
Average Precision	*0.54112*
Average Recall	*0.59941*
Average F1 Score	*0.52336*

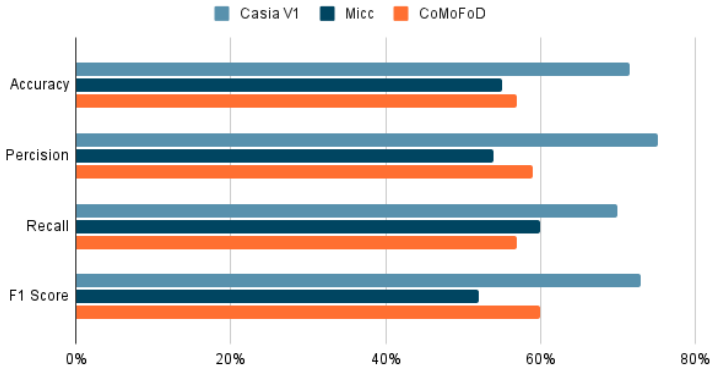
Table 10: MICC-F220 -averages

CoMoFoD

Average Accuracy	*0.57179*
Average Precision	*0.59247*
Average Recall	*0.57173*
Average F1 Score	*0.60325*

Table 11: CoMoFoD -averages

Figure 12: VGG



5.1.3. VGG_Ela

CASIA V.1

Average Accuracy	*0.88969*
Average Precision	*0.90961*
Average Recall	*0.89151*
Average F1 Score	*0.87201*

Table 12: CASIA V.1 -averages

MIC-F220

Average Accuracy	*0.70254*
Average Precision	*0.75841*
Average Recall	*0.70898*
Average F1 Score	*0.72954*

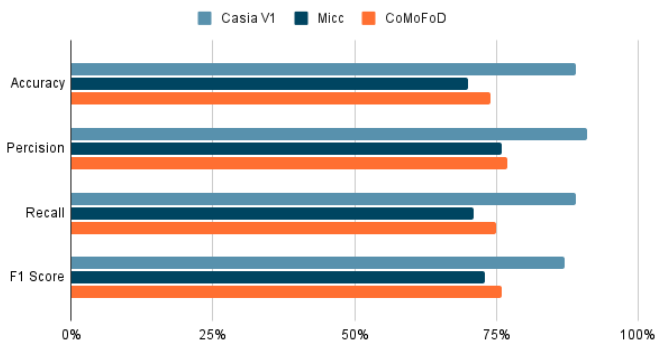
Table 13: MICC-F220 -averages

CoMoFoD

Average Accuracy	*0.74201*
Average Precision	*0.77001*
Average Recall	*0.75211*
Average F1 Score	*0.75895*

Table 14: CoMoFoD -averages

Figure 13: VGG-ELA



Models Classifications

Samples	Modifed AlexNet	ELA Modified AlexNet	VGG-ELA
	**Tampered**	**Tampered**	**Tampered**
	**Tampered**	**Tampered**	**Tampered**
	**Authentic**	**Authentic**	**Tampered**
	**Tampered**	**Authentic**	**Authentic**

Table 15: Models Classifications to Random Samples

5.1.4. BusterNet Model

5.1.4.1 Image-level Evaluation

Casia v2

Precision	78.37%
Recall	65.22%
F1 Score:	71.19 %

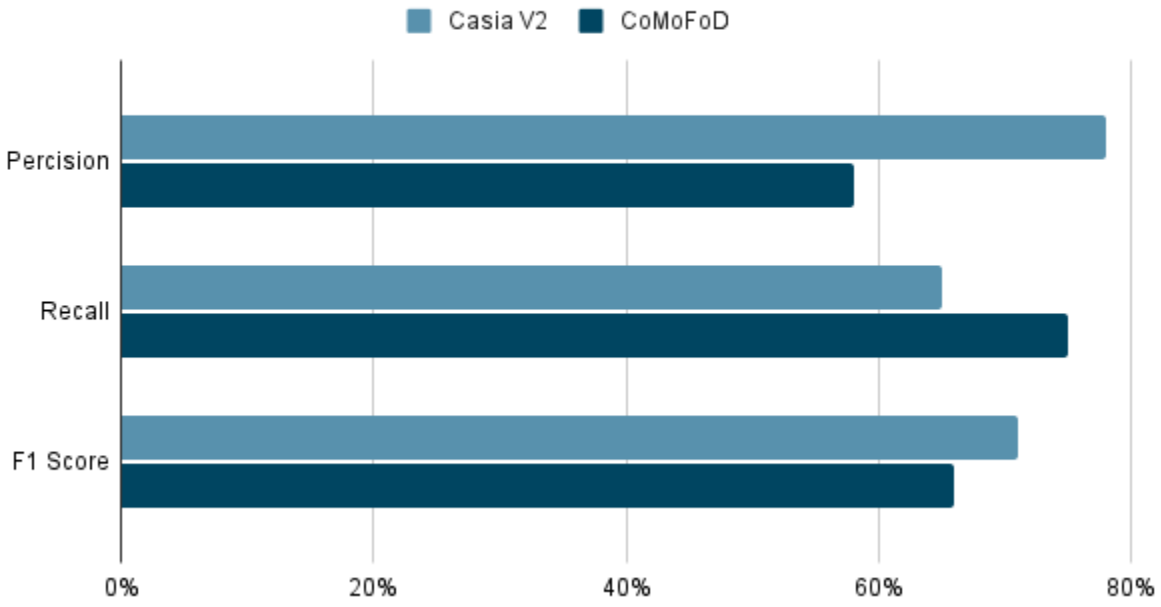
Table 16: Casia v2-averages

CoMoFoD

Precision	58.35%
Recall	75.50%
F1 Score:	65.83 %

Table 17: CoMoFoD -averages

Figure 14: Image Level



5.1.4.2. Pixel-Level Evaluation

Casia V2

Precision	55.71%
Recall	43.83%
F1 Score:	45.56%

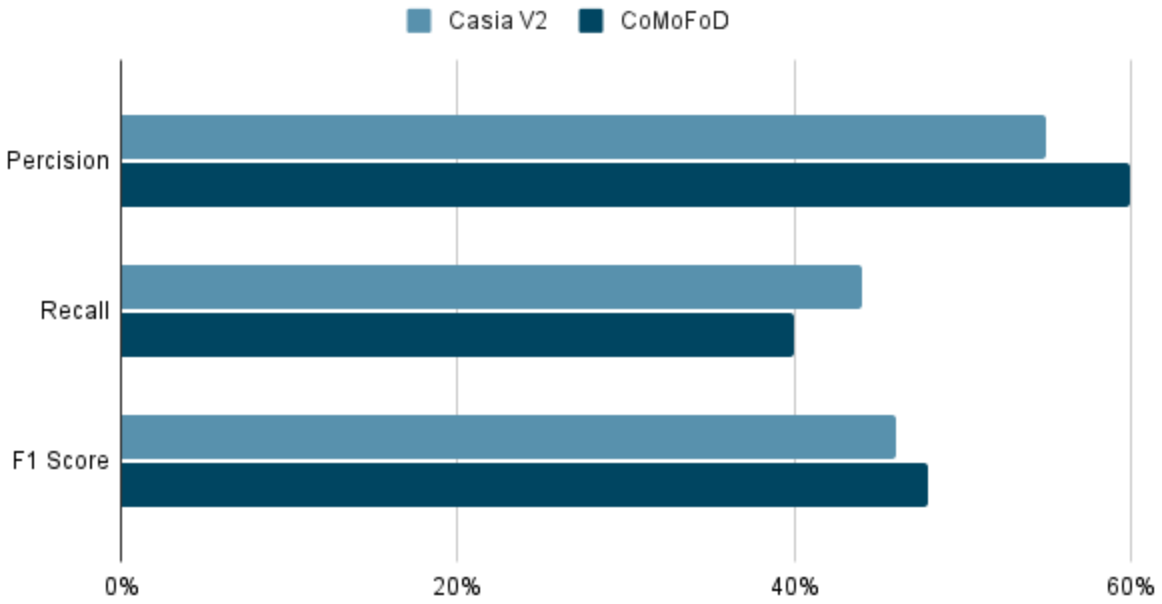
Table 18: Casia v2 -averages

CoMoFoD

Precision	60.23%
Recall	40.14%
F1 Score:	48.17%

Table 19: CoMoFoD -averages

Figure 15: Pixel Level



5.1.4.3. Testing Visual Result

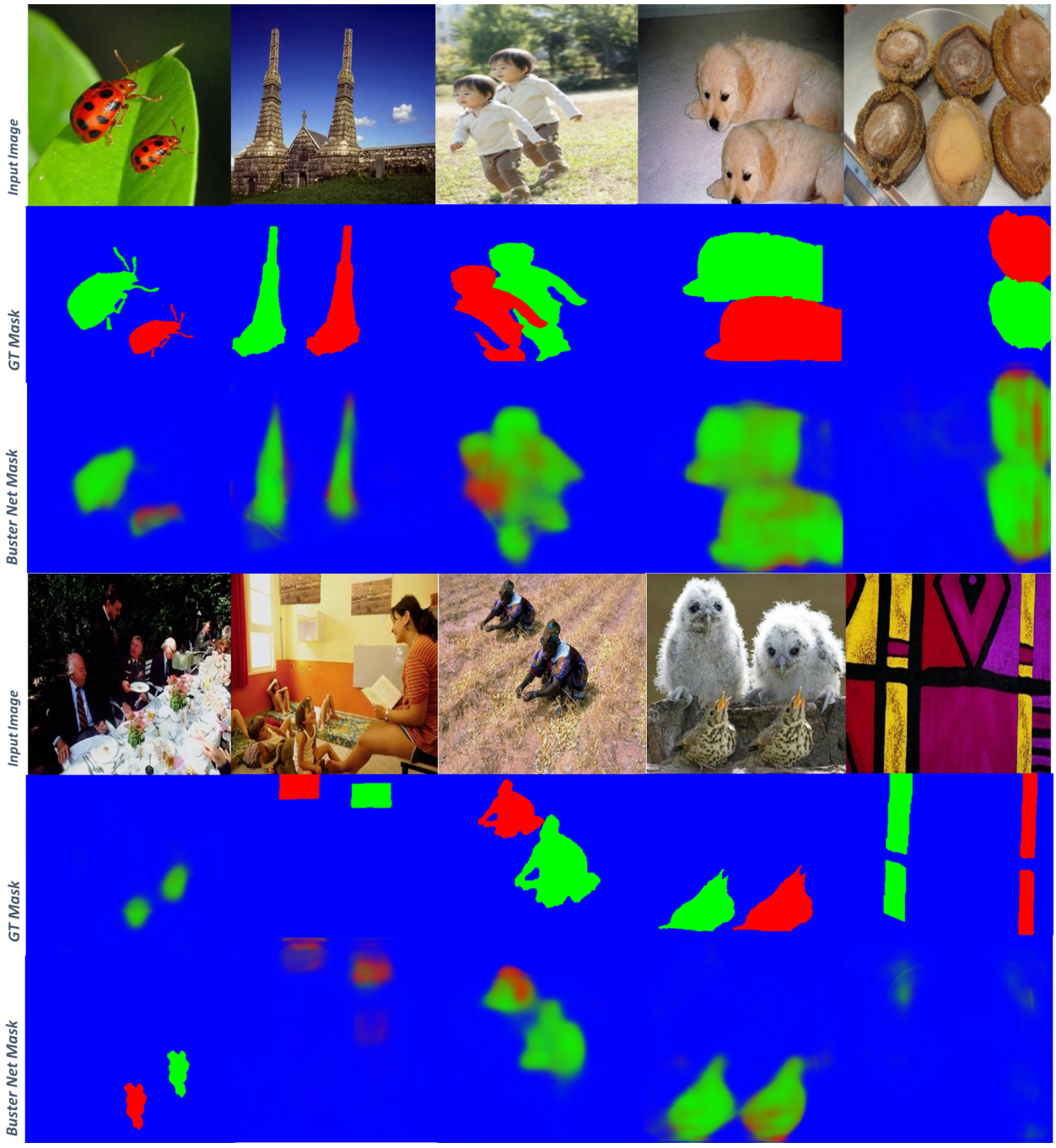


Figure 16: Test Samples

5.2 “Unmask The Fakes”User Interface

Home Page

Figure 17: User Interface “Home Page”

Authentic Image is given

Figure 18: User Interface "Authentic Image is given"

Tampered Image is given

Figure 19: User Interface "Tampered Image is given"

Chapter 6: Conclusions and Future Work

Conclusions:

The advancements made in the field of image forgery detection through the use of deep learning techniques are highlighted by the preceding research. Different models, like AlexNet and BusterNet, have been created and applied to address the difficulties presented by picture control and fabrication. These models have shown promising outcomes in recognizing legitimate and manufactured pictures, confining controlled locales, and accomplishing continuous discovery capacities.

AlexNet, with its convolutional brain network engineering, has been effectively utilized for picture fabrication recognition. It has demonstrated its capacity to identify images as genuine or manipulated by extracting distinguishing characteristics. Its performance and scalability are further enhanced by incorporating transfer learning, ensemble techniques, and data augmentation. Through the use of methods like class activation maps and saliency maps, AlexNet's interpretability makes it easier to comprehend the model's decision-making process.

BusterNet, on the other hand, presents a brand-new deep neural architecture developed specifically for the purpose of detecting copy-move forgery. It has a fusion module and a two-branch architecture for localizing copy-move and manipulation regions. BusterNet's capacity to perceive source and target districts separates it from existing calculations. Its performance outperforms current copy-move detection algorithms, demonstrating its effectiveness and resistance to a variety of known threats.

Error Level Analysis (ELA) and deep learning models like AlexNet give image forgery detection even more advantages. ELA supplements profound learning approaches by catching pressure irregularities and supporting the exact restriction of controlled districts. By consolidating ELA with profound learning models, the general recognition precision and dependability can be additionally gotten to the next level.

Future Work

Notwithstanding the headway made, there are as yet a few roads for future exploration in picture imitation location:

1. Improved Methods for Detection: The detection accuracy and robustness of deep learning models like AlexNet and BusterNet can be further enhanced through ongoing development and refinement. Superior performance may result from investigating novel architectures, loss functions, and optimization methods.

2. Expansion of Large-Scale Datasets: The training of models on diverse and challenging data would be made possible by expanding existing datasets and creating new ones with a wider variety of forgery types, including more complex and sophisticated manipulations. This would make it easier for them to generalize and use in the real world.

3. Adversarial Assaults: It is essential to investigate the deep learning models' susceptibility to adversarial attacks specific to image forgery detection. For accurate and dependable forgery detection, it is essential to develop robust models that can withstand such attacks.

4. Hybrid Methods: Investigating the mix of profound learning models with different strategies, for example, conventional picture handling techniques or chart based calculations, might actually prompt more complete and precise falsification recognition frameworks.

5. Evaluation Method: Try to find more accurate way to evaluate the localization of the busternet model

All in all, the headways in profound learning-based picture falsification recognition have prepared for more compelling recognizable proof and examination of controlled pictures. Researchers have made significant progress in identifying various forms of image forgery by utilizing tools like Error Level Analysis and models like BusterNet. However, in order to address upcoming issues and enhance the capabilities of forgery detection systems for use in real-world situations, additional research and development are required.

Chapter 7: References

[1] Rao, Y.; Ni, J. A deep learning approach to detection of splicing and copy-move forgeries in images, 4–7 December 2016.

[2] Salloum, R.; Ren, Y.; Kuo, C.-C.J. Image Splicing Localization using a Multi-task Fully Convolutional Network (MFCN), 2018.

[3] Ouyang, J.; Liu, Y.; Liao, M. Copy-move forgery detection based on deep learning, October 2017.

[4] A Walk Through AlexNet, Hao Gao, 5 April 2020.

[5] "Copy-Move Forgery Detection Using Histogram of Oriented Gradients (HOG)" by Zhou et al.

[6] "Copy-Move Forgery Detection Using Generative Adversarial Networks" by Li et al.

[7] "A SIFT-based forensic method for copy-move attack detection and transformation recovery" by Amerini et al.

[8] "A DCT-based block matching method for copy-move forgery detection and localization" and was authored by Bayram et al.

[9] "Copy-Move Forgery Detection Using Dual-Tree Complex Wavelet Transform and Clustering" and was authored by Luo et al.

[10] "Copy-Move Forgery Detection Using Combined SIFT and SURF" by Bayram et al.

[11] "Detecting Copy-Move Forgery in Digital Images" by Fridrich et al

[12] "Scale-Invariant Feature Transform for Copy-Move Forgery Detection" and was authored by Bayram et al.

[13] "Deep Learning-Based Copy-Move Forgery Detection Using Convolutional Neural Networks" by Zhang et al.

[14] "Copy-Move Forgery Detection Based on Image Noise Patterns" authored by Li et al.

[15] " Image copy-move forgery detection using a block-based matching algorithm with local binary patterns " by Smith et al.

[16] " Forgery Detection using Scale-Invariant Feature Transform (SIFT) Descriptors " by Johnson et al.

[17] " Deep Convolutional Neural Networks for Copy-Move Forgery Detection " by Lee et al.

[18] "Local Phase Quantization for Discrimination of Textured and Homogeneous Regions in Images" by Wang et al.

[19] Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: Computer vision and pattern recognition (CVPR), 2010 IEEE conference on. pp. 3485–3492. IEEE (2010)

[20] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Doll´ar, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)

[21] “Optimization of a Pre-Trained AlexNet Model for Detecting and Localizing Image Forgeries” , Soad Samir , 20 May 2020

[22] “BusterNet: Detecting Copy-Move Image Forgery with Source/Target Localization” Yue Wu1, Wael Abd-Almageed1, and Prem Natarajan

[23] “Image forgery detection using error level analysis and deep learning” Ida Bagus Kresna Sudiatmika , Fathur Rahman, Trisno, Suyoto