**DEEP LEARNING**

**FINAL PROJECT**



Team Members:

Yegneshwar Rao Ginjupalli

Avinash Reddy Bhavanam

Ramya Krishna Gadiparthi

**Abstract**:

This paper introduces an innovative approach to enhance digital image forgery detection, addressing fake information dissemination through forged images on social media. Unlike existing techniques that often focus on specific forgery types, the proposed method employs deep learning, specifically transfer learning, to simultaneously uncover two common types of image forgery. The approach centers around identifying discrepancies in compressed quality within the forged area, distinguishing it from the rest of the image.

The methodology involves a deep learning-based model that calculates the disparity between the original image and its compressed version. The resulting featured image serves as input to a pre-trained model, with its classifier replaced and fine-tuned for binary classification. The study evaluates the performance of eight different pre-trained models adapted for this purpose. Experimental results demonstrate the superior efficacy of the proposed technique, outperforming state-of-the-art methods according to various evaluation metrics, charts, and graphs. Particularly noteworthy is the employment of the MobileNetV2 pre-trained model, yielding a remarkable 95% detection accuracy rate with expedited training time due to fewer parameters. This comprehensive approach showcases a significant advancement in addressing the multifaceted challenge of digital image forgery detection.

**Introduction:**

The tampering of a digital image is called digital image forgery, these forged images cannot be detected by the naked eye. Such images are the primary sources of spreading fake news and misleading information in society with the aid of diverse social media platforms like Facebook, Twitter, etc. The editing software tools that can make these forgeries are available for free with some advanced features that are used for image tampering such as GNU, GIMP, and Adobe Photoshop. Such forgeries can be detected using digital image forgery algorithms and techniques. These algorithms are used in image security especially when the original content is unavailable. Digital image forgery means adding unusual patterns to the original images that create a heterogeneous variation in image properties and an unusual distribution of image features.

This task can be broken down into smaller sub-tasks depending on the way that the image has been manipulated. Three of the most common manipulations in literature are: copy-move, splicing and removal. There have already been numerous approaches in the literature that address the task of image forgery detection.

**Technical Approach:**

One of the main tasks of our study is to create a pipeline that is able to recognize tampered from authentic images. For that reason, we took inspiration from the architecture proposed by an IEEE journal. They propose a CNN that is used as a feature extractor that takes an image patch as input and outputs a feature representation. This feature representation is then fed into an SVM classifier that predicts whether the features correspond to an original or tampered image. The network architecture and the procedure via which the CNN and the SVM are trained are detailed in the following sections.

**CNN Training:**

To train the CNN architecture in a way that it can focus on the local regions of the artifacts and learn to recognize them, image patches must be extracted from the dataset used. The size of the extracted patches is 128x128x3, meaning that there is a 128x128 patch for every color channel. The extraction was performed by applying a patched-size sliding window with a stride equal to eight for the whole image. Following that, the tampered patches are discriminated against from the non-tampered ones. As far as the tampered patches are concerned, we compare each patch with the equivalent patch (from the same region of the image) of the mask of this image and keep the ones that contain part of the tampered region, as demonstrated in Figure 5. Moreover, we only keep two random tampered patches per image, as training CNN with a huge number of extracted patches would be computationally expensive. When it comes to the non-tampered patches, we apply the same technique but now on the equivalent authentic image and randomly select two of these patches. Finally, to improve the generalization ability of CNN and avoid overfitting, we augment the patches extracted by rotating them four times by a step of 90 degrees.

**SVM Architecture:**

After the training of the CNN network, the next step is to train the SVM classifier. For that purpose, we extract every possible p × p patch from both the original and the tampered images using a sliding-window with strides to scan the whole image. This process results in n new patches per image which are passed through the CNN resulting in n feature representations Yi (400-D). That said, these representations need to be fused into a single Yˆ [k] representation for each image before being passed as an input to the SVM. Similarly, to [9], max or mean pooling is applied on each dimension of Yi over all the n patches extracted from each image: Yˆ [k] = Mean or M ax {Y1[k]...Yn[k]} where k ∈ [1, 400] dimensions. The resulting 400-D feature vector is then used by the SVM to classify images either as original or tampered.

**Input Design and Output Design:**

**INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use while retaining the privacy. Input Design considered the following things:

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

OBJECTIVES

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

**OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can be used easily and effectively. When analyzing design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create documents, reports, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**Implementation:**

**MODULES:**

* Dataset
* Importing the necessary libraries
* Retrieving the images
* Splitting the dataset
* ELA image analysis
* Building the model
* Apply the model and plot the graphs for accuracy and loss
* Accuracy on test set
* Saving the Trained Model

**MODULES DESCRIPTION:**

**Dataset:**

In the first module of Digital Image Forgery Detection, we developed the system to get the input dataset. Data collection process is the first real step towards the real development of a machine learning model, collecting data. This is a critical step that will cascade in how good the model will be, the more and better data that we get, the better our model will perform. There are several techniques to collect the data, like web scraping, manual interventions. Our dataset is placed in the project and it’s located in the model folder. The dataset is referred from the popular standard dataset repository kaggle where all the researchers refer it. The dataset consists of 12,615 Digital Image Forgery images.

The following is the URL for the dataset referred to, from kaggle.

<https://www.kaggle.com/datasets/jayaprakashpondy/casia2-dataset>

**Importing the necessary libraries:**

We will be using Python language for this. First, we will import the necessary libraries such as keras for building the main model, sklearn for splitting the training and test data, PIL for converting the images into array of numbers and other libraries such as pandas, numpy, matplotlib and tensorflow.

**ELA image analysis:**

Error level analysis is a 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. When the image is preserved, 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 preserved 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 describes how the output of ELA on the condition of the image.

**Splitting the dataset:**

In this module, the image dataset will be divided into training and testing sets. Split the dataset into Train and Test. 80% train data and 20% test data. This will be done to train the model on a subset of the data, validate the model's performance, and test the model on unseen data to evaluate its accuracy. Split the dataset into train and test. 80% train data and 20% test data.

**Results:**

The accuracy and cross-entropy loss per epoch during the CNN training for the two datasets is shown below:

A graph of a graph

Description automatically generated with medium confidence A graph of a number of data

Description automatically generated with medium confidence

**Accuracy:**

Training Accuracy:98% and

Validation Accuracy of 92%.

**Conclusion:**

In the ever-evolving landscape of digital media, ensuring the authenticity and integrity of images is a critical concern. The project, "Digital Image Forgery Detection Using CNN and Error Level Analysis (ELA)," presents a comprehensive and effective solution to address this challenge.

Through the fusion of Convolutional Neural Network (CNN) model architecture and Error Level Analysis (ELA), this project has demonstrated a robust system capable of accurately detecting digital image forgeries. By leveraging the strengths of both techniques, the system achieves high accuracy and adaptability, making it well-suited for a wide range of forgery detection scenarios.

The utilization of a diverse dataset containing authentic and tampered images ensures the system's ability to generalize and perform effectively in real-world applications. Its real-time implementation potential further enhances its utility, allowing it to be seamlessly integrated into various platforms and applications where immediate forgery detection is paramount.

With its ability to identify both simple and complex forgeries, the proposed system contributes to the preservation of image authenticity and the prevention of digital manipulation. It offers a practical solution for forensic analysts, content moderators, and individuals seeking to verify the credibility of digital visual content.

In conclusion, the "Digital Image Forgery Detection Using CNN and ELA" project represents a significant advancement in the field of digital image forensics. Its robustness, adaptability, and real-time capabilities position it as a valuable tool in the ongoing battle to maintain the trustworthiness of digital images in an era of digital manipulation and misinformation.