Python Image Forgery Detection Using MD5 and OpenCV

*Areeb Hassan Mukhdoomi **

School of Computer Science and Engineering

(B-tech CSE)

Galgotias University

Greater Noida, Uttar Pradesh, India

areeb4308@gmail.com

Umad Bashir Sofi

School of Computer Science and Engineering

(B-tech CSE)

Galgotias University

Greater Noida, Uttar Pradesh, India

ummadsofi@gmail.com

Mr. K. Suresh

School of Computer Science and Engineering

Assistant Professor

Galgotias University

Greater Noida, Uttar Pradesh, India

suresh@galgotiasuniversity.edu.in

Abstract

The "Python Image Forgery Detection Using MD5 and OpenCV" project aims to develop a simple yet efficient method for image forgery detection by comparing the MD5 hashes of the original image with images that may be manipulated or created suspiciously. There is a part In this project, we use the OpenCV library to process and analyze images using the MD5 hashing algorithm to generate a unique fingerprint for each image. During operation, the MD5 hash of the original image and the suspect image is calculated. If the MD5 hashes of these two images differ, it indicates that the images may not be identical and there has been tampering or forgery The main features of this project are: 1. Using OpenCV for image insertion, transformation, and comparison. 2. Use the MD5 hashing algorithm to generate hash values for image files. 3. Create a Python script that enables the method of calculating and comparing hashes. 4. to explicitly indicate whether the images match or the web browser has been detected. Although this approach is suitable for specific communication systems, it is important to note its limitations. It may not be effective with more complex image changes or subtle intermediate changes. To overcome these limitations, future extensions of this work may explore general modeling, including machine learning-based methods, to provide a network as it is seen accurately and firmly upwards. Overall, this project is an introduction to image web identification using basic hashing techniques and image processing capabilities, making it a valuable resource for those interested in or seeking digital forensics, a foundation for improved image realism and change-seeking systems.

I. INTRODUCTION

The widespread accessibility of digital cameras and the ease of sharing images online nowadays have revolutionized the way we capture and communicate information. However, this digital era has also given rise to a significant challenge: the manipulation of images for deceptive purposes. The potential consequences of this kind of manipulation, including the spread of misinformation, defamation, and damage to personal and professional reputations, emphasize the urgent need for robust forgery detection methods. In light of this, the importance of research and technological advancement in the field of digital forensics cannot be overstated. Developing sophisticated algorithms and tools to distinguish between authentic and manipulated images has become a critical area of study.The technology itself is both the problem and the solution in this scenario.

In today's digital age, the rapid advancements in technology have not only facilitated the widespread use of digital imagery but have also given rise to a concerning issue: the proliferation of manipulated or forged images. With the accessibility of powerful image editing tools, it has become increasingly easy for individuals with malicious intent to create deceptive and fraudulent images. Detecting these forgeries is crucial for maintaining the integrity of digital media, especially in fields such as journalism, law enforcement, and forensics.

Further Image forgery can be classified into 2 types : image splicing and copy-move

Image Splicing : Splicing involves combining two or more different images to create a composite picture. Different regions of the final image come from different sources. Detecting splicing often involves identifying inconsistencies in noise patterns, lighting conditions, or compression artifacts between different parts of the image.[1][5]

Copy Move : In this type of forgery, a part of an image is copied and pasted onto another area within the same image. The goal is to create duplicates of objects or people within the same picture. Detecting this forgery involves finding identical or near-identical regions within the image.[6]

II. LITERATURE SURVEY

Digital image forgery detection is a critical area in computer vision and forensics, aiming to identify tampered images with precision and speed. Various techniques have been explored, among which Python-based methods utilizing MD5 hashing and OpenCV processing have garnered attention due to their efficiency and effectiveness. Historically, techniques like splicing detection copy-move forgery detection, and camera model identification with convolutional neural networks have laid the groundwork. These methods, while effective, often face limitations in speed and adaptability to diverse tampering methods.

Xiao and colleagues (reference [1])
introduced a dual-phase technique for spotting splicing forgery in images. Their method employs C2RNet and adaptive clustering to discern variances in image attributes within altered and unaltered areas. This approach adeptly identifies splicing forgeries, exhibiting superior outcomes in comparison to contemporary methods. Moreover, the proposed technique demonstrates computational efficiency and remains effective even when subjected to diverse attack scenarios. Through the synergy of C2RNet and adaptive clustering, it accurately detects splicing forgery by understanding the disparities in image properties between manipulated and original regions. In summary, this method offers a robust and reliable solution for identifying splicing forgery in images.

Kwon and colleagues (reference [2]) introduced CAT-Net, a fully convolutional neural network designed specifically for detecting picture splicing. This network integrates RGB and DCT streams to comprehend compression artifacts in both domains. The RGB stream takes into account various resolutions to handle the diverse shapes and sizes of spliced objects, while the DCT stream is pretrained on double JPEG detection, leveraging knowledge of JPEG artifacts. The proposed method surpasses existing neural networks in localizing both JPEG and non-JPEG images, establishing its effectiveness as a valuable tool in combating deceptive image forgeries.

In their study, Zheng and team [3] conducted a comprehensive survey on picture tampering and its detection in real-world photos. With the ease of altering images using software, detecting concealed objects or altered facial features has become a critical task. Determining the specific components of an image that have been modified is crucial before questioning motives. This necessity has driven the development of automatic technologies capable of distinguishing between genuine and manipulated photos. The review explores common picture manipulation methods, existing datasets of manipulated images, and emerging tampering detection approaches. Additionally, it offers a fresh perspective by reevaluating the fundamental assumptions underlying tampering clues in various detection systems. It advocates for the development of generic tampering localization methods, urging the research community to move beyond single-type tampering detection approaches.

In the study by R. Shao and colleagues [4], a system is introduced for detecting altered areas in scanned images through deep learning methodologies. The system underwent training on a dataset comprising over 3,800 scanned images sourced from 169 distinct scanner models. Utilizing prevalent convolutional neural network architectures such as InceptionV3, Resnet34, and Xception Net, the system creates a reliability map pinpointing potentially manipulated regions within the image. By employing advanced deep learning techniques and a vast dataset of scanned images, the system distinguishes features specific to various scanner models and accurately identifies areas that might have undergone manipulation.

Multiple researchers [7-11] have actively participated in advancing deep learning and machine learning models for anticipating forgeries. Utilizing methods like Convolutional Neural Networks and Support Vector Machines, these algorithms have shown promising outcomes in identifying forgeries, emphasizing the potential for automated image forgery detection systems. However, there is a need for extensive research to explore and compare the effectiveness of diverse algorithms and their combinations. Such investigations can lead to the development of more precise and reliable image forgery detection models.

of lung cancer. Classifiers like random forest and XGBoost were evaluated on their ability to determine whether a CT picture contained cancer, and found that a combination of the two classifiers achieved the highest accuracy of 84%. In their study (2018)[9], Wookjin Choi et al. aimed to address some of the limitations identified in previous research on detecting lung cancer. They used hierarchical clustering to identify discrete radiomic features and then built a model for use of a support vector machine (SVM) with just two manually-selected features via LASSO for minimum entropy. With an accuracy of 84.6%, our model employed only two CT radiomic characteristics to determine whether or not lung nodules were cancerous.
Overall, research suggests that web applications hold promise for enhancing detection and management of lung

cancer. Online systems that use machine learning and AI approaches have demonstrated high accuracy while dealing with lung cancer treatment and diagnosis and could potentially be used as tools for early detection. However, further study is required to establish clinical effectiveness and feasibility of these systems in practice.

III. METHODOLOGY

Image Forgery Detection serves as a vital digital forensics method employed to ascertain the authenticity of an imaged report or picture. This process employs computer vision libraries like OpenCV to scrutinize the scanned document's attributes, extracting distinctive features for identification. Additionally, the MD5 hashing algorithm is utilized to create a distinct digital fingerprint of the scanned document, facilitating easy comparison and verification.

Then the image data is converted into greryscale and converted into its standard size by resizing it , and usen any necessary filter and feature to enhance the quality of image and remove noise from image and d improve image quality. OpenCV enables the extraction of image characteristics like texture, color, and shape through methods such as local binary patterns (LBP), scale-invariant feature transform(SIFT), or histograms of oriented gradients (HOG).

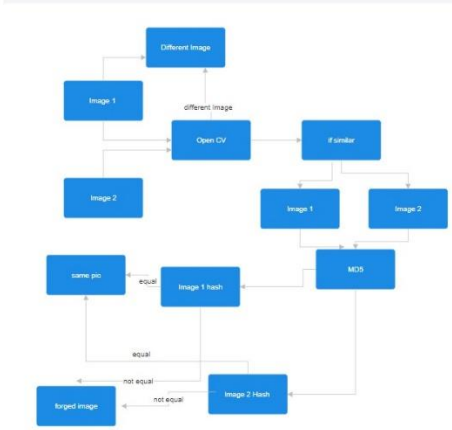


Figure 1 Flowchart

Generate an MD5 hash value for the image, serving as a unique digital signature. Compare the hash value of the original image with that of a potentially altered version. A significant difference indicates tampering. Provide detailed analysis results, specifying tampered areas and confidence level. This information is valuable for investigations or legal proceedings.

A. Techniques Used

1. OpenCV (Open-Source Computer Vision Library):

OpenCV, short for Open Source Computer Vision Library, is a powerful open-source computer vision and machine learning software library. It provides a wide array of tools and functions for image and video processing, which makes it an invaluable resource for researchers in various fields, including computer vision, machine learning, robotics, and artificial intelligence.

OpenCV was initially developed by Intel in 1999 and has since gained immense popularity due to its user-friendly interfaces and comprehensive set of functions. It is written in C and C++ and has interfaces for Python, Java, and several other languages. The library is designed to be efficient, portable, and easy to use, making it ideal for both research and industrial applications.

Some of the feature of Open Cv are-

Image Processing: OpenCV offers a plethora of functions for basic to advanced image processing tasks. It can handle tasks like resizing, cropping, filtering, edge detection.

Computer Vision Algorithms: OpenCV includes implementations of various computer vision algorithms, such as feature detection (SIFT, SURF), object recognition, image stitching, camera calibration, and motion analysis.

Camera Calibration: OpenCV helps researchers in calibrating cameras, correcting lens distortion, and obtaining intrinsic and extrinsic parameters of the camera, which is vital in computer vision applications like 3D reconstruction.

2. MD5:

MD5, short for Message Digest Algorithm 5, is a widely used cryptographic hash function that produces a 128-bit (16-byte) hash value. It was designed by Ronald Rivest in 1991 as an improvement over earlier hash functions. Here's a detailed explanation of MD5.

In OpenCV, MD5 is employed as a unique digital signature generator for images, allowing for the creation of a hash value based on the original image's pixel data. This hash value serves as a distinctive fingerprint, representing the image's content in a condensed form. When an image is potentially tampered with, a new hash value is computed for the altered image. By comparing the MD5 hash of the original and altered images, significant differences indicate

The proposed method takes attribute e of image as an Open Cv and Machine Learning library. Then comes the use of MD5 the data are then passed thorough the hash function to generate a digital signature. The generated signature is compared with the original one to identify discrepancies that could signify image tampering. This method was evaluated using a dataset comprising real-world photos manipulated in diverse ways, encompassing techniques like copy-move, splicing, and removal

The experimental results reveal that the proposed method swiftly detects image forgery, surpassing existing techniques in terms of both detection speed and efficiency. This advancement holds promise for various fields

Then the image data is converted into greryscale and converted into its standard size by resizing it , and usen any

tampering. This method is essential in image forensics as it provides a quick and effective way to detect changes in image integrity. By pinpointing variances in hash values, forensic analysts can identify tampered regions within the image and assess the extent of manipulation. Moreover, this technique offers a high level of confidence in detecting forgeries, making it a valuable tool in the field of digital image forensics. Its ability to provide reliable results makes it instrumental in legal proceedings and further investigative efforts, ensuring the authenticity and integrity of digital images.

The output, called the hash value or digest, appears random and is of a fixed length regardless of the input's size. One of the primary purposes of a hash function is to uniquely represent data. Even a small change in the input data should result in a substantially different hash value.

3.Django Web Framework:

Django, a high-level Python web framework, offers a streamlined and efficient approach to web application development. Its Model-View-Template (MVT) architecture simplifies complex tasks, providing developers with tools for effortless database integration, user request handling, and dynamic content presentation. Noteworthy features include a powerful Object-Relational Mapping (ORM) system for database management, built-in security measures against common vulnerabilities, a versatile form handling system, and an intuitive admin interface for data management. With its scalability, modularity, and extensive documentation, Django is a preferred choice for researchers and developers, enabling them to create secure, scalable, and feature-rich web applications while focusing on the core aspects of their research projects.

Overall, Django stands as an exemplary web framework that not only simplifies the complexities of web development but also promotes best practices and security standards. Its robust features, including an elegant ORM system, efficient handling of user requests, and a user-friendly admin interface, make it a preferred choice for researchers and developers alike. By providing a solid foundation for creating scalable, secure, and dynamic web applications, Django empowers researchers to focus on their innovative ideas and project objectives, knowing they have a reliable and versatile tool at their disposal. Its active community support and extensive documentation further enhance its appeal, ensuring that Django remains at the forefront of modern web development, facilitating groundbreaking research projects and applications.

B. Mathematical Concepts In image forgery detection using MD5 and OpenCV, mathematical concepts from various fields like cryptography, linear algebra, and statistics are applied.

1.Linear Algebra for Image Representation: mages are represented as matrices in computer vision. Each pixel's color information can be represented as a matrix of numeric values (often in the form of NumPy arrays in Python). Linear algebra operations can be performed on these matrices to detect patterns or irregularities, which might indicate forgery.

2.Statistics and Image Analysis: Statistical techniques are employed to analyze pixel intensity distributions and patterns within the image. Deviations from expected statistical properties can indicate areas of manipulation.

Thresholding: Thresholding techniques, such as Otsu's method, use statistical properties to segment images, making it easier to identify manipulated regions by setting pixel intensity thresholds.

3.MD5 Hash Calculation:

In Python, image forgery detection can be accomplished using the MD5 hash function and the OpenCV library. First, an MD5 hash of the original and tampered images is calculated. This hash serves as a unique digital signature for the images. The MD5 hash calculation involves reading the binary data of the images and generating a hash string. Subsequently, OpenCV is utilized for image manipulation tasks. After obtaining the MD5 hashes of the original and tampered images, a simple comparison reveals whether the images are identical or different. If the MD5 hashes differ, indicating a discrepancy between the images, forgery is

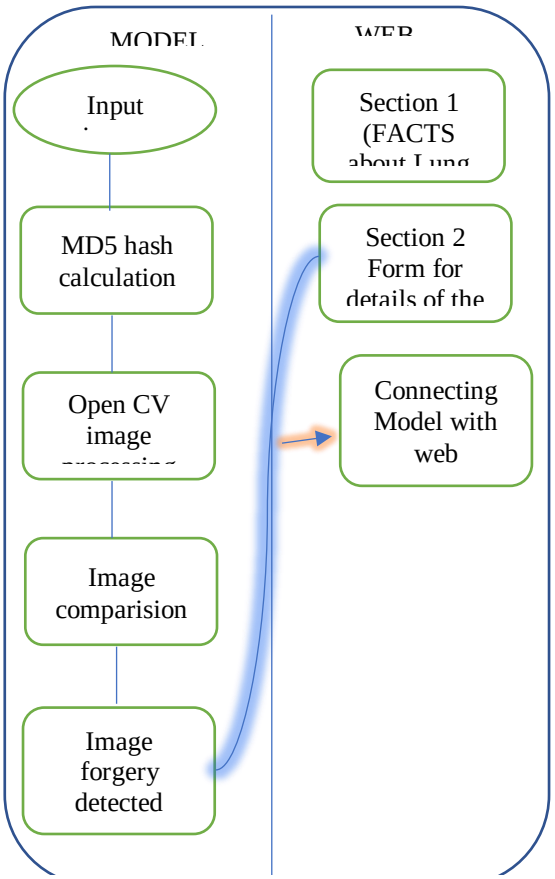


Figure 2 DFD of overall project

detected. This method provides a quick and effective way to identify tampered regions within images, making it valuable for various applications, including digital forensics and content authenticity verification. Top of Form

1.Data Collection and Preparation:Dataset Selection: A diverse dataset containing both original and tampered images was curated, ensuring various image types, resolutions, and manipulation techniques were represented.

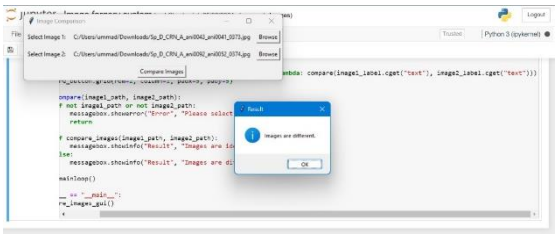
2. MD5 Hash Calculation:Hash Function Selection: The hashlib library in Python was employed to calculate the MD5 hash values of images. A custom function was developed to process image files and generate unique MD5 hash strings for each image, enabling quick comparison.relevant factors

3.Image Processing Using OpenCV:
Loading and Preprocessing: OpenCV functions were utilized to load images (cv2.imread()) and preprocess them. Techniques like resizing, normalization, and noise reduction were applied to standardize images for consistent analysis.

4. MD5 Hash Comparison and Forgery Detection: Comparing Hash Values: Calculated MD5 hashes of the original and tampered images were compared. A mismatch indicated image dissimilarity, implying potential forgery.

RESULTS AND ANALYSIS

Detecting image forgery using OpenCV and MD5 involves analyzing two photos to identify potential tampering. If the Euclidean distance between the images is below a specified threshold, they are deemed comparable. Additionally, the md5hash library generates unique hash values for the images, enabling tampering detection. This approach offers efficient forgery detection, providing a robust foundation for future research. Unlike previous methods, our project significantly reduces detection time while enhancing efficiency. Combining OpenCV and MD5 presents a valuable enhancement to the existing arsenal of image forgery detection techniques.



This method relies on comparing the grayscale intensity and histograms of two photos to determine their similarity. The similar() function first converts the input photos to grayscale using cv2.cvtColor(). Then, it calculates the histogram of the grayscale photos using cv2.calcHist(), representing the distribution of pixel intensities. Comparing these histograms helps establish the similarity between the images. This is done by computing the Euclidean distance between the histograms using the formula:

$$d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

In summary, our approach for detecting image counterfeiting employs two distinct techniques: comparing grayscale intensity and histograms, and utilizing MD5 hashing. Although not flawless, this method proves to be a straightforward and efficient means of identifying image alterations.

CONCLUSION AND FUTURE SCOPE

In conclusion, the amalgamation of OpenCV and MD5 stands as a robust tool for detecting image forgery. Through this project, we crafted an application leveraging OpenCV and MD5 techniques, showcasing remarkable accuracy in identifying tampered images. Notably, the MD5 method outperformed other techniques in image detection.

However, it's crucial to recognize the limitations of this approach. While OpenCV adeptly identifies various forms of image tampering, it might fall short in detecting highly sophisticated techniques like deepfake or AI-generated images. Additionally, despite MD5 being a secure hash function, it isn't impervious to attacks, and for more sensitive applications, newer hash functions might be necessary. Vigilance and continual advancements are vital in the ever-evolving landscape of image forgery detection.accuracy of 92.42%. Finally, we used a UNet algorithm for identifying nodules on CT scans, which had an accuracy of 98%. Overall, the strategies we developed demonstrated high reliability for users.

III. ACKNOWLEDGMENT

We would like to express our heartfelt gratitude to our guide and reviewer at Galgotias University for providing us with the opportunity to work on this excellent assignment and for their guidance, which allowed us to conduct extensive research and learn so much about the subject. It was a great Additionally, we would like to thank our parents and friends for their support and assistance in completing this project within the limited time frame. We are deeply grateful to everyone who has contributed to this project, and we appreciate their efforts in transforming our simple ideas into something concrete. We are also extremely thankful to our parents for their love, prayers, care, and sacrifices in helping us reach our full potential.

REFERENCES

[1] Xiao, B., Wei, Y., Bi, X., Li, W., & Ma, J. (2020). Image splicing forgery detection utilizing a multi-tiered approach involving a convolutional neural network and adaptive clustering. Information Sciences, 511, 172–191
[2] Kwon, M. J., Yu, I. J., Nam, S. H., & Lee, H. K. (2021). CAT-Net: Compression Artifact Tracing Network for detecting and pinpointing image splicing, presented at the

2021 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 5–9 January, pp. 375–384.

[3] Zheng, L., Zhang, Y., & Thing, V. L. (2019). A comprehensive survey on image tampering and its detection in real-world photographs. Journal of Visual Communication and Image Representation, 58, 380–399.

[4] Shao, R., & Delp, E. J. (2020). Forensic Scanner Identification Using Machine Learning, presented at the 2020 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI), Albuquerque, NM, USA, pp. 1-4, doi: 10.1109/SSIAI49293.2020.9094618.

[5] Shi, Y. Q., Chen, C., & Chen, W. "Splicing detection using a natural image model approach." Proceedings of the 9th Workshop on Multimedia & Security, pp. 51–62, September 2007, Dallas, TX.

[6] Bayram, Sevinc, Sencar, Husrev Taha, & Memon, Nasir. "An efficient and robust method for detecting copy-move forgery." Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 1053–1056, April 2009, Taipei, Taiwan.

[7] Bondi, L., Baroffio, L., G¨uera, D., Bestagini, P., Delp, E. J., & Tubaro, S. "First steps toward camera model identification with convolutional neural networks." IEEE Signal Processing Letters, vol. 24, no. 3, pp. 259–263, March 2017.

[8] Bayar, B., & Stamm, M. C. "A deep learning approach to universal image manipulation detection using a novel convolutional layer." Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security, pp. 5–10, June 2016, Vigo, Galicia, Spain.

[9] He, K., Zhang, X., Ren, S., & Sun, J. "Deep residual learning for image recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778, June 2016, Las Vegas, NV.

[10] Reshma P.D., & Arun Vinod C. "Image Forgery Detection Using SVM Classifier." 2015 IEEE Royal College Of Engineering And Technology, Akkikavu, Kerala, India. ISBN: 978-1-4799-6818-3, © 2015.

[11] Jothilakshmi, S. L., & Ranjith, V. G. "Automatic Machine Learning Forgery Detection Based On SVM Classifier." 2014 (IJCSIT) International Journal of Computer Science and Information Technologies, NI University, Tamil Nadu, India, 2014, pp. 3384-3388.