**Image Forgery detection using MD5, OpenCV And SHA256**

***Abstract*— Gone are the days when photo editing had a simple scope; now, it is highly advanced and poses a drastic challenge in terms of verifying and authenticating visual content. Such conditions make it strange from having facilities for catching image fraud from everywhere, including social media, journalism, surveillance, as well as legal evidence. To illuminate the current scenario, this project presents a unique solution for image forgery detection by integrating MD5 combined with OpenCV and SHA256. The high-level goal for the system is to check if two images are exactly the same or slightly altered versions of the same image. First of all, process samples images by using the MD5 (Message Digest Algorithm 5) function, which gives unique hash values for each image sample. Subsequently, this hash serves as a digital fingerprint, so that fast and efficient comparison could be implemented. However, according to the vulnerabilities in MD5, like collision; SHA256 (Secure Hash Algorithm 256) is enabled into the system for more accurate dependability and cryptographic strength. SHA256 would give a stronger and collision-free hash, making a better and much more tamper-proof verification process. In parallel, OpenCV (Open Source Computer Vision Library) is used for pixel-by-pixel and structural comparison for identifying differences between two images. This mode of visual analysis will become useful in identifying the minute manipulations an image may have undergone, such as region cloning, resampling, or splicing, which might not always be detected by use of hash values. Thus, the proposed system combines the strength of visual inspection with those of cryptographic validations to attain greater accuracy in forgery detection. If both the MD5 and SHA256 match and the OpenCV image comparison validates visual similarity, the system declares the images as authentic and keeps them in the secure database for later retrieval or audit trails. In cases of differences in hash or visual mismatches, images are reported as tampered or forged and user alerted about such post-processing. This hybrid style indeed provides a very reliable and user-friendly way of detecting image forgery, which can adapt well in digital forensics, media verification, secure storage of data, and legal documentation.**

**Keywords: Image forgery detection, MD5, SHA256, OpenCV, image comparison, hashing algorithms, image authenticity, digital image processing, database storage, secure hashing.**

Introduction

Ensuring authenticity of digital images has become very significant, especially in our present digital world, where visual content plays a pivotal role in communication, documentation, journalism, and evidence sharing. With advanced program software for the manipulation of images being very much available, and easy to use, image forgery is becoming common and, consequently, very difficult to detect. Image manipulation can have serious implications in the media, law enforcement, healthcare, and cybersecurity: spreading fake news on social media platforms; altering evidence in court and forensic cases. Manual methods to identify these forgeries involve heavy amounts of time and error due to the subtle techniques of modern image manipulation. Therefore, an automated, intelligent, and reliable system to detect image tampering will help in maintaining the integrity of visual data needs to come into being. To resolve this building-up challenge, the project under the title "Image Forgery Detection using MD5, OpenCV and SHA256" will have a multi-directional approach involving both cryptographic as well as computer vision tools to verify image authenticity. The system uses two widely applicable hash functions, MD5 and SHA256, to generate a unique digital signature for every image. These hash values act as fingerprints: if the digital image is tampered with, say, with the change of one pixel, its hash will be such that it can be easily said to have changed. While MD5 offers speed, it has collisions; therefore, for more strength in verification, SHA256 is introduced. Apart from the cryptographic methods, the system adopts OpenCV, a library for computer vision which is freely available, to analyze the image in-depth. The OpenCV system detects pixel-level differences and structural changes, as well as manipulations like copy-move forgery, splicing, and resampling that don't change the hash but would affect the visual content of the image greatly. Basically, the workflow of the system is that two images are uploaded for a comparative analysis by the user. Hash comparison is first done using MD5 and SHA256 among these images by the system. If any difference is found, that image is flagged as forged. If the hashes match, the system proceeds to further investigate using OpenCV to check whether the images are visually identical. Once both hash checks and visual checks pass an image is stored securely in a database for further referencing or auditing. This layered validation increases the accuracy of forgery detection while ensuring the system's resiliency against both cryptographical weaknesses and actual visual manipulation approaches. The proposed solution is made to be fast, reliable, and flexible, enabling it to be adopted in practice for innumerable real-time applications requiring trust in digital imagery. By synergizing MD5, SHA256, and OpenCV, this project wishes to produce a secure and scalable framework for image forgery detection.

## Objective Of The Study

## This growth in digital content gave rise to a corresponding growth in images manipulation and brought about the need to develop reliable techniques for the detection of image forgery, for depending on which sector the image forgery can be serious-journalism and law enforcement or the markets where the authenticity of visuals counts. With the increasing sophistication of image editing tools, it is almost impossible for humans to detect the alterations and hence gives the motivation to have an automated solution. Given the acute requirement of having a highly efficient, trustworthy, and scalable system which involves state-of-the-art algorithms such as MD5, SHA256, and OpenCV to detect forgery in images; this project is becoming very critical. Combining secure cryptographic hashing with OpenCV's visual analysis will help to assess changes and ensure integrity of digital content while increasing protection against the threats of forgery and misinformation.

## Scope Of The Study

## The scope of this project is to develop an automated image forgery detection system for ensuring electronic authentication of images using MD5, SHA256, and OpenCV algorithms. The system will allow users to upload two different images and subsequently uses MD5 and SHA256 hashing techniques to generate unique hash values for both images by which slight alterations in it can be tested very easily. OpenCV analysis will then be employed to the evaluation of visual comparisons and pixel-wise differences between the two input images. The system notifies to the user if the images compare or have been modified with an accurate and reliable result. It has an additional feature of storing the validated images in the database so that the image can be verified at some point in time. It is ultimately applicable to media, law enforcement, and digital content creation where integrity of image is of paramount importance. The project would emphasize speed, accuracy, and ease of use, so it practically can be utilized by several systems at multiple places.

## Problem statement

Currently, images are relevant in the digital world as they are used in fields such as media, law enforcement, and social media. Yet, the fact that images can be manipulated fairly easily raises serious concerns regarding their verifiability. Today, the common image forgery detection tools either lack sensitivity to detect subtle alterations or they are computationally expensive. The proposed solution solves these problems by providing a multi-tiered approach utilizing MD5, SHA256, and OpenCV for image forgery detection. Hashes like MD5 and SHA256 are secure ways of determining any changes to the outer structure of the image, while OpenCV is a means to delve deep into a visual analysis that can discover differences obscured to hashing techniques. The aim of this project would be to propose an image forgery detection technique that is reliable, efficient, and accurate for ascertaining the authenticity of digital content.

RELATED WORK

**The rapid advancements in image editing software and the ever-increasing dependence on visual data for digital communication have turned image forgery detection into a very important area of research. Much work has been done to improve forgery detection accuracy and reliability by using different techniques, which include cryptographic hashing functions and computer-vision algorithms. The present literature survey makes an attempt to indicate some major contributions concerning the integration of MD5, OpenCV, and SHA256 in connection with image forgery detection.**

**Bhatia and Ghosal (2018) proposed a hybrid approach using MD5 and SHA algorithms to generate unique hash values for image authentication. Their study demonstrated that combining multiple hashing techniques enhances detection accuracy and provides robustness against various image manipulations, emphasizing the value of hybrid cryptographic approaches in forgery detection systems.**

**Chen and Zhang (2020) conducted a comprehensive survey on image forgery detection techniques based on visual features, highlighting the effectiveness of using OpenCV in conjunction with machine learning algorithms. The study concluded that visual inconsistencies at the pixel level can be accurately detected using computer vision tools, significantly improving the system's ability to identify subtle tampering.**

**Zhang and Wang (2019) explored hashing functions like MD5 and SHA256 integrated with OpenCV for detecting forgeries in digital images. Their research confirmed that cryptographic hash functions offer high-speed detection of image alterations, while OpenCV adds a layer of visual verification to ensure comprehensive analysis.**

**Sharma and Kumar (2017) introduced a method that combines MD5 hashing and pixel-level comparison for detecting tampered regions in images. Their findings showed that MD5 efficiently identifies changes in image data, and the use of OpenCV for structural comparison further validates the authenticity of the image.**

**Singh and Arora (2016) evaluated deep learning models for image forgery detection in comparison to traditional hashing methods. While deep learning demonstrated promising results in classification, the authors concluded that integrating simpler approaches like MD5 and OpenCV could offer a more interpretable and performance-balanced solution for real-world applications.**

**Zhao and Liu (2021) developed an image authentication system using SHA256, citing its superior cryptographic strength compared to MD5. Their work emphasized the enhanced security and reliability of SHA256 in forgery detection, especially in sectors requiring high data integrity such as legal and governmental domains.**

# **Patel and Mehta (2018) investigated OpenCV's capability in real-time forgery detection, focusing on visual discrepancies such as splicing, cropping, and resizing. Their research demonstrated OpenCV's effectiveness in capturing minor yet critical image alterations that may be missed by hashing techniques alone.**

# **Khan and Ahmed (2017) proposed a combined approach using image hashing and visual analysis techniques. The integration of cryptographic and computer vision methods improved detection accuracy and speed, confirming the benefits of a dual-layered strategy for robust image verification.**

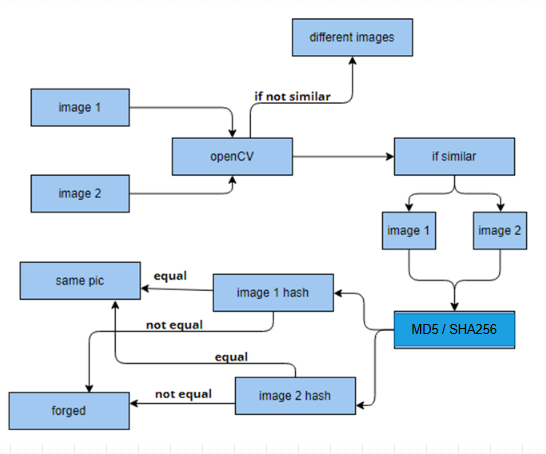
# **Li and Yu (2019) enhanced forgery detection by combining SHA256 with visual feature matching using OpenCV. Their approach proved effective in detecting both localized and global tampering, offering a reliable framework suitable for high-stakes environments such as forensic investigations.**

# **Lastly, Singh and Gupta (2020) presented a system that utilizes MD5 and SHA256 hashing in conjunction with pixel analysis via OpenCV. Their study reported that the proposed system outperforms conventional methods by detecting even minor manipulations, thereby validating the effectiveness of a multi-layered detection mechanism.**

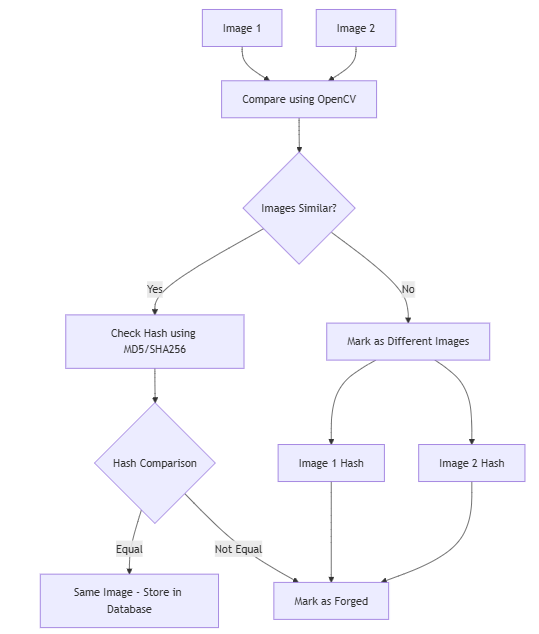
# Proposed System Workflow

The proposed system is capable of a significant improvement trade possible in the image forgery detection by employing three strong algorithms: MD5, SHA256, and OpenCV. This multi-layered approach is much more reliable in the accuracy of detecting image manipulation as opposed to the consideration of cryptographic and visual discrepancies with regard to digital images. The base of the system's hash-based detection method will be MD5 and SHA256. MD5 will be used to hash the images so that detection can be performed quickly for very simple changes, such as re-encoding or modification of an individual pixel in an image. Speed in computing makes MD5 suitable for simple changes, thus a fast efficiency in verifying the authenticity of the image. On the other hand, the SHA256 will add another layer of security, guaranteeing a more solid and secure hash value. When an image is changed, it becomes very difficult to generate the same hash as the original. Therefore, this dual-hashing mechanism will protect tampering more thoroughly, where the MD5 algorithm is known for its speed, while SHA256 is renowned for ensuring cryptographic security levels. Besides these M5 and SHA256 cryptographic verifications, OpenCV will also be there for detailed pixel-layer analyses. It is this vital component that will detect very minor changes in the images that are likely missed by hashing algorithms. OpenCV will inspect the visual structure of the images, trying to bring out differences that will indicate object insertion, image splicing, and other forms of pixel manipulation. It is by this extensive analysis that the more advanced forgery of images can be detected and flagged by the system that normal hashing methods probably would not catch. In such a system, the user uploads two images and the system compares the two images. If the images are exactly the same, the system will save the original image into a secure database for further reference. If it shows any difference, the system will alert the user that the images have been modified. This only with hashing algorithm and pixel-level analysis improves the accuracy of detection from the proposed system. Thus, this hybrid system is one effective solution for image forgery detection. The entirety of this will handle digital image tampering using cryptographic hardness and image processing techniques.

**Fig-1: Project flow**



**ARCHITECTURE**



**IMPLEMENTATION AND RESULTS**

**Modules**

 **Image Upload Module**

* **Function**: Allows users to upload two images for comparison. This module handles the file input from users and stores the uploaded images temporarily for further processing.
* **Features**:
  + User-friendly interface for image selection and upload.
  + Supports multiple image formats (JPEG, PNG, etc.).
  + Handles errors related to invalid file types.

 **MD5 Hash Generation Module**

* **Function**: This module generates the MD5 hash value for the uploaded images. The MD5 hash is used to check the integrity of the image and detect any changes.
* **Features**:
  + Generates a unique MD5 hash for each image.
  + Compares MD5 hashes of the two uploaded images to detect any discrepancies.
  + Provides the hash value for further processing.

 **SHA256 Hash Generation Module**

* **Function**: Similar to the MD5 module, this module generates the SHA256 hash value, which is more secure and robust than MD5, to strengthen the forgery detection process.
* **Features**:
  + Generates a SHA256 hash for each image.
  + Compares SHA256 hashes of the two uploaded images.
  + Offers enhanced security and accuracy in forgery detection.

 **Image Comparison Using OpenCV Module**

* **Function**: This module uses OpenCV to perform pixel-based image comparison, detecting visual differences between the two uploaded images. It helps in detecting subtle alterations that may not affect the hash values.
* **Features**:
  + Analyzes images pixel-by-pixel.
  + Detects differences in image size, cropping, splicing, or pixel-level modifications.
  + Provides a visual output of image differences to the user (highlighting differences, etc.).

 **Forged Image Detection Module**

* **Function**: This module combines the outputs from the MD5, SHA256, and OpenCV comparison modules to determine whether the images are identical or altered. It issues a notification to the user based on the comparison results.
* **Features**:
  + Aggregates results from MD5 and SHA256 hashes and OpenCV image analysis.
  + If the images match, stores them in a database.
  + If the images are different, notifies the user that the images are altered.

 **Database Storage Module**

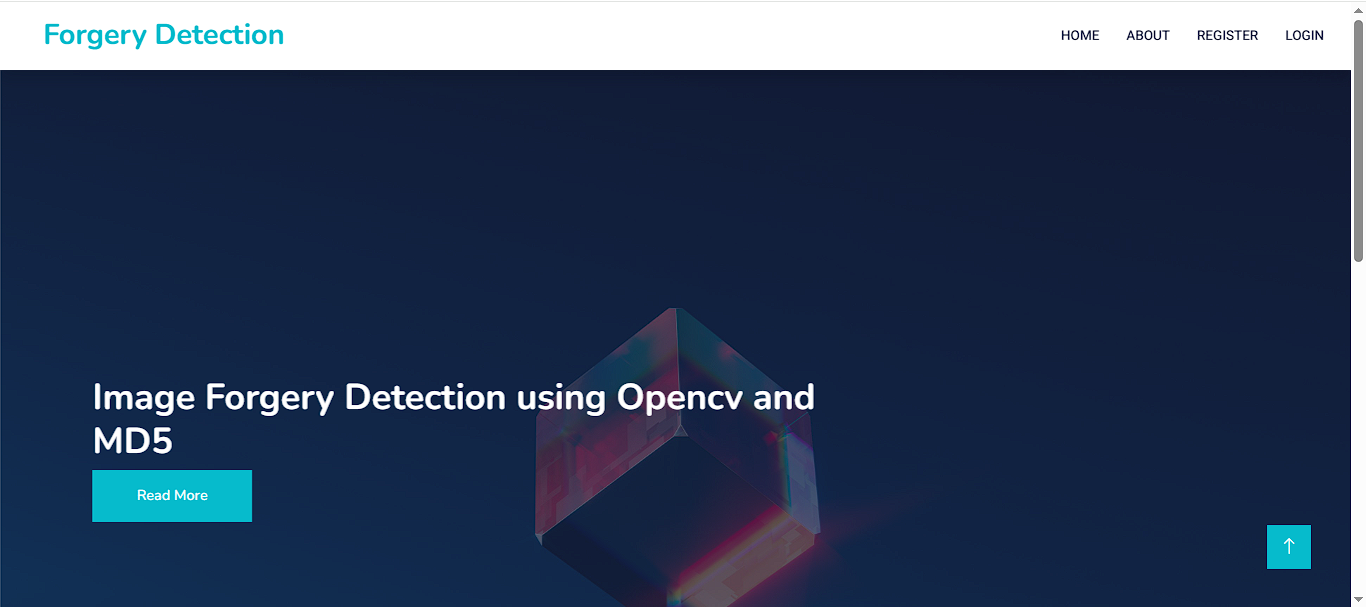
* **Function**: Stores images that are verified as identical and authentic in a database for future reference and record-keeping.
* **Features**:
  + Secure storage of authentic images.
  + Enables future verification and retrieval of stored images.
  + Provides database management tools for easy access and querying.

 **User Interface (UI) Module**

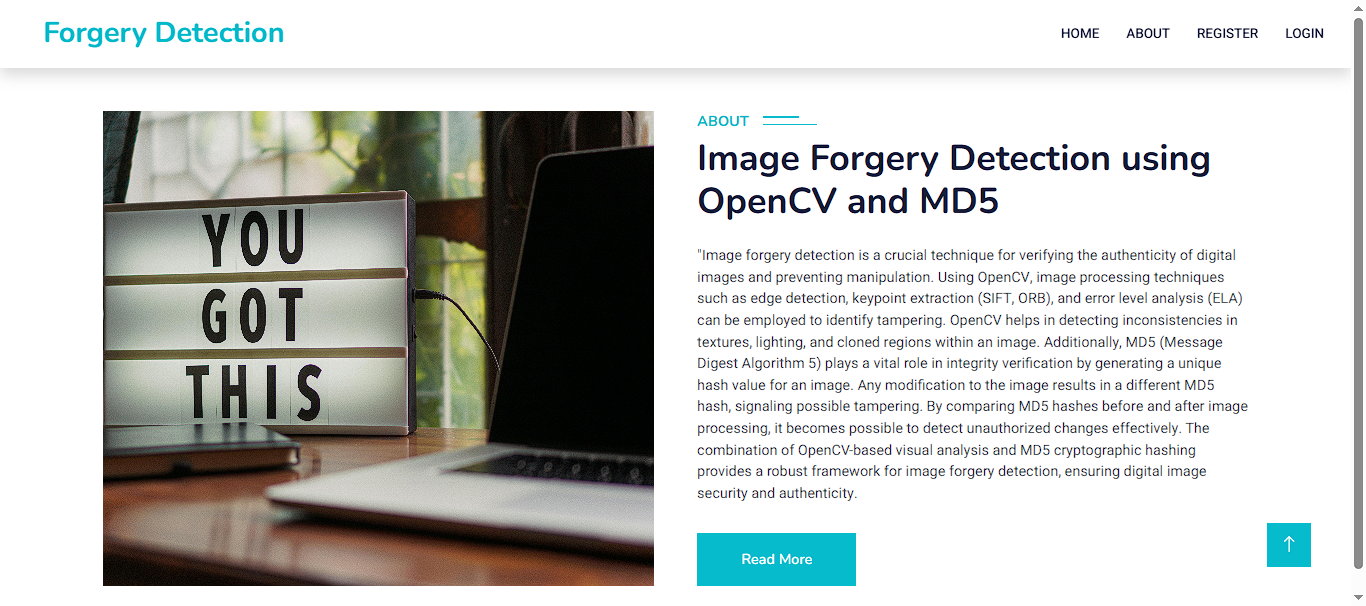
* **Function**: Provides an interface for users to interact with the system, upload images, view comparison results, and receive notifications.
* **Features**:
  + Simple, intuitive interface for image upload and result display.
  + Allows users to view the hash values of the images and the pixel differences detected by OpenCV.
  + Displays feedback on whether the images are identical or not, along with relevant details.

# **Results**

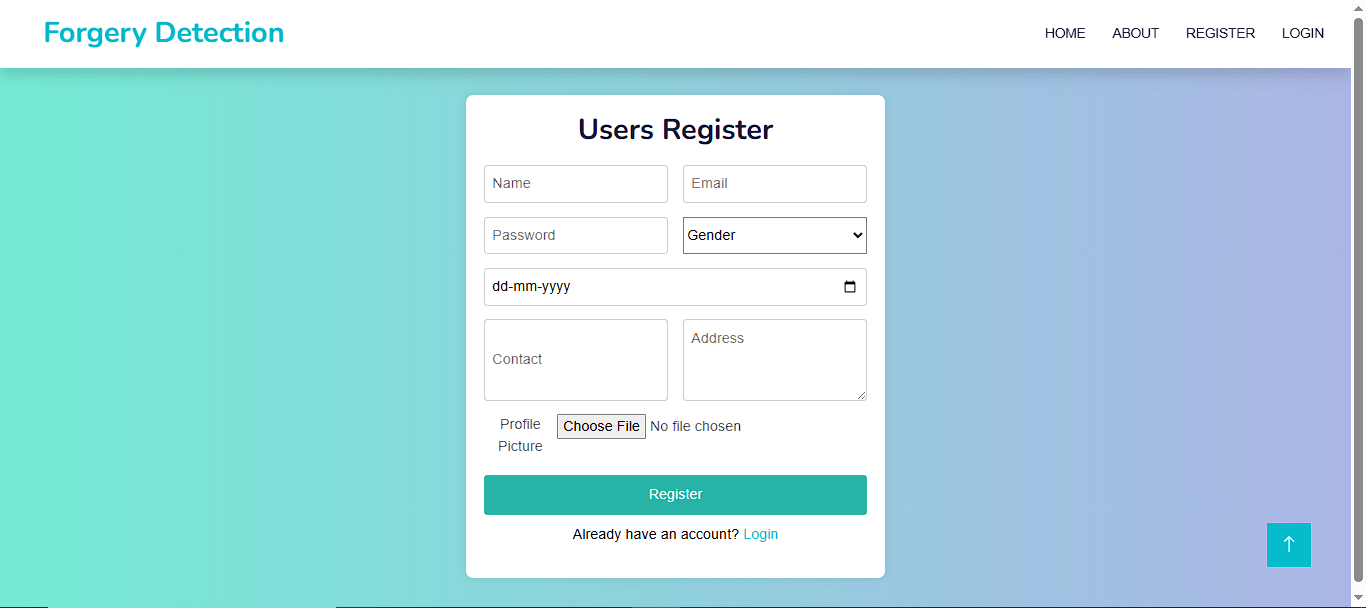
Home: It is a home page.



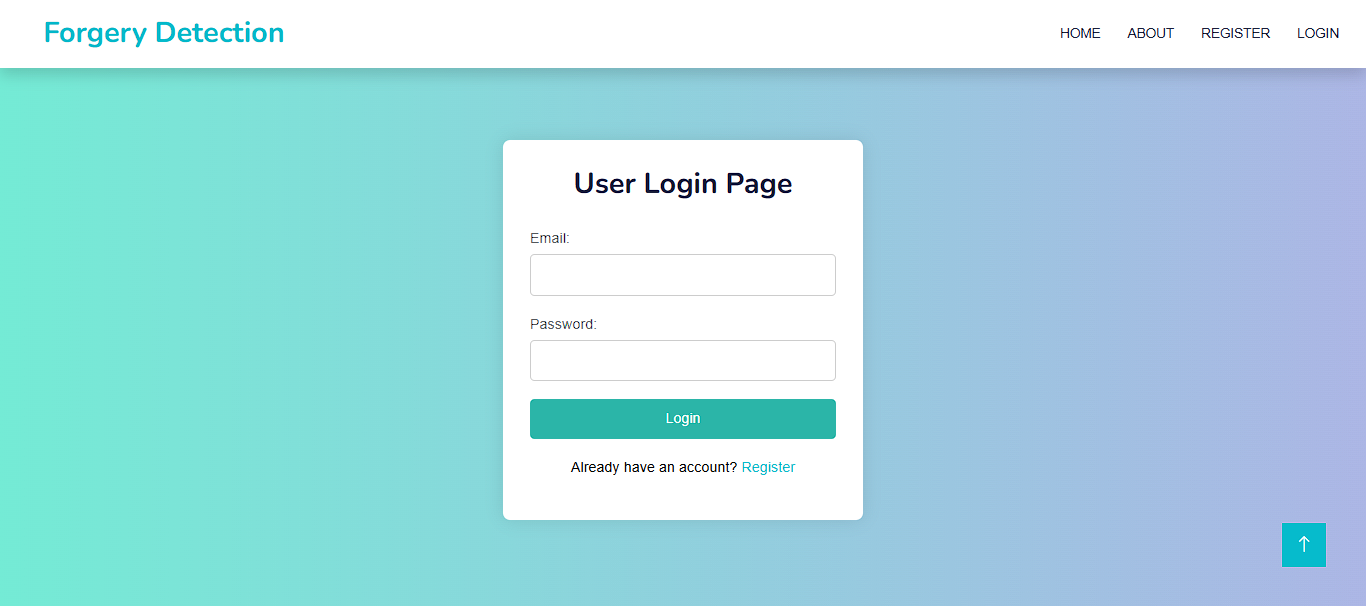
About Page: It is about page.



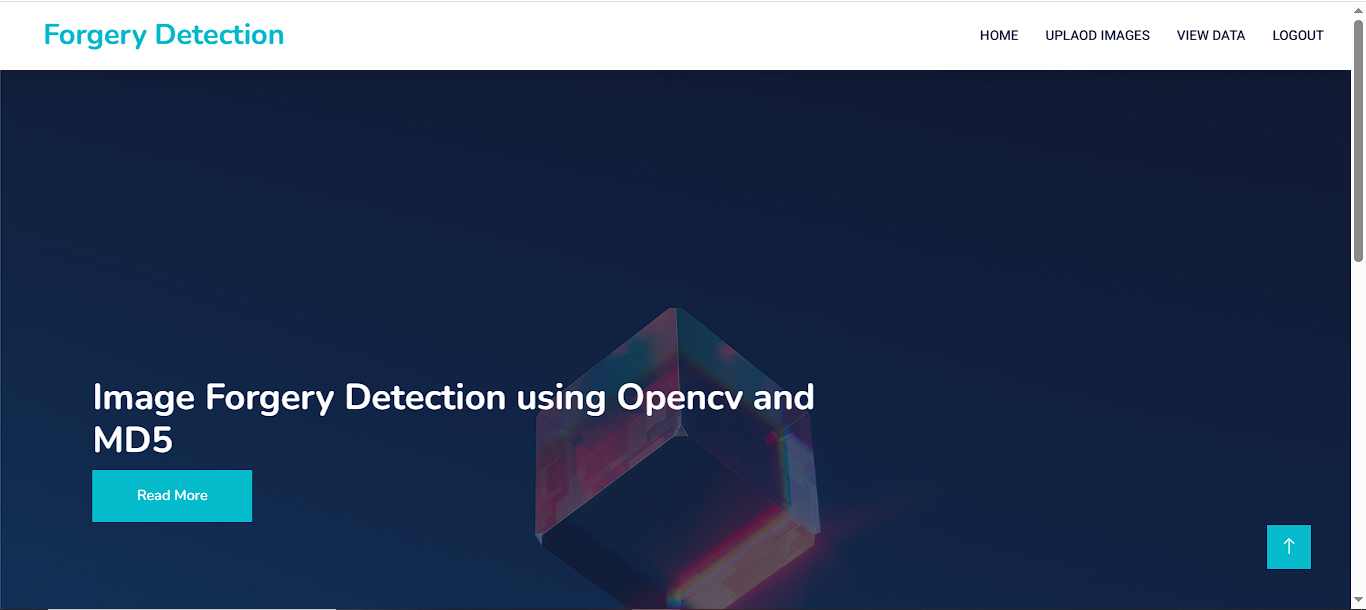
Registration Page: here user can register



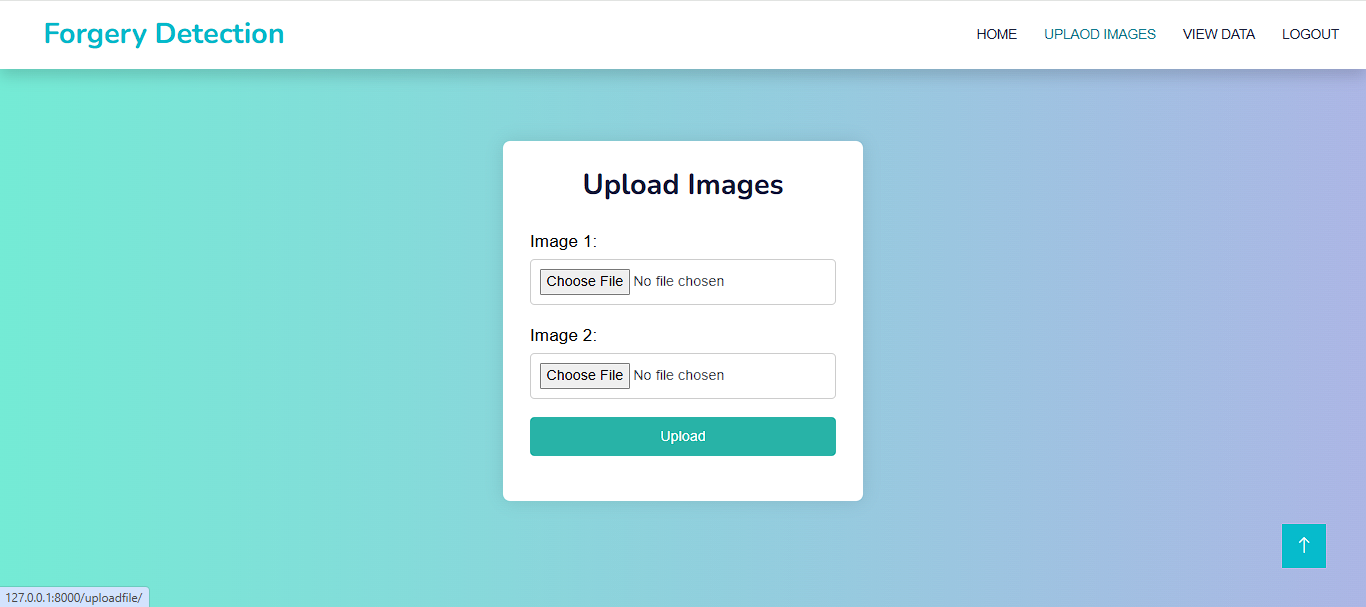
Login Page : here user can login



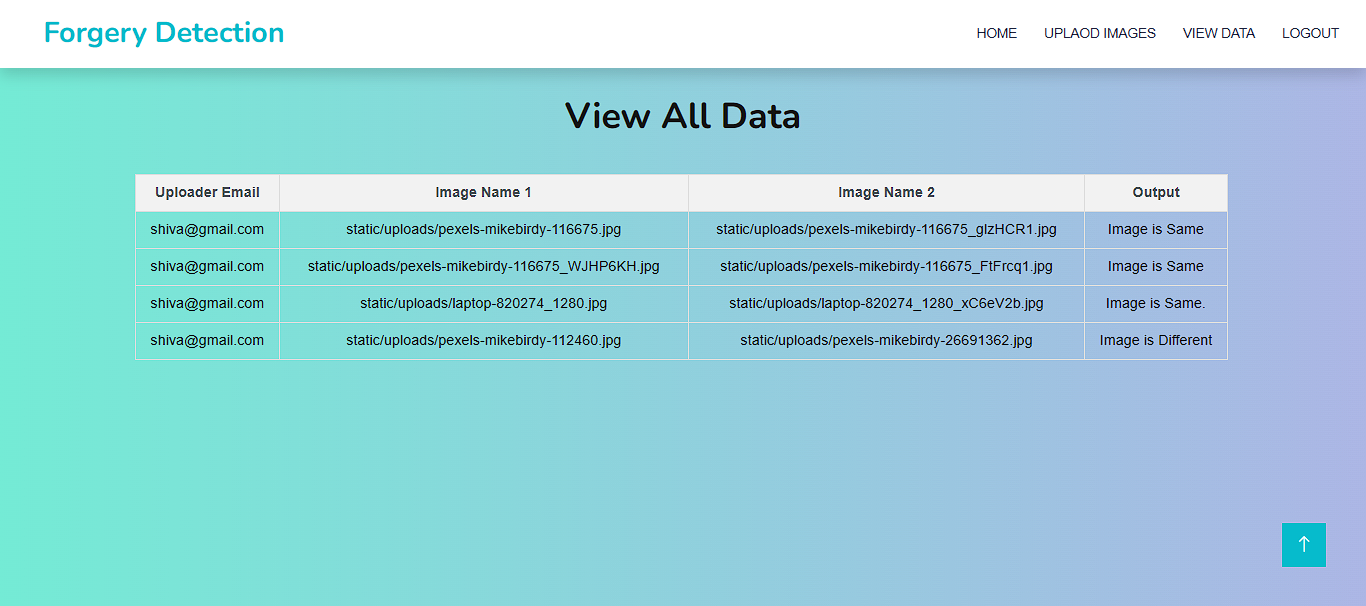
Home Page : user home page



Upload Page: user can upload file here



View Page : user can view all data



# Conclusion

The integrated approach comes to light as the potent and highly efficient forgery detection mechanism based on the phenomenal merging of three excellent algorithms MD5, SHA256 and OpenCV. The capability of multi-layered performance as well as the reliability of detection methodology adds up in providing a very high probable accurate detection of changed images. With MD5 and SHA256 Hashing Algorithm offers an equal dimension for manipulation detection at structure as well as cryptography level. These hashing algorithms help achieve unique fixed-size hash values for every change in original image files, even minute. This will be achieved by recognizing discrepancies by hashing data for the suspect image against the one for the original image kept in a secure database, which would allow for the flagging of forged images. In addition to this, the cryptographic assurance is made effective with pixel-based image processing using OpenCV, thereby enhancing the efficiency of detection.The structural and visual description analyses made possible by OpenCV include very subtle levels of processing that are able to escape 'detection' from the traditional cryptography measures, such as object addition or removal, modification of textures, and tampering with image edges. Thus, this pixel-based detection will give an extra layer of analysis for the comprehensive forgery detection system. Another convenience offered by such a system is the fact that it provides storage of original images in a secure database for reference, thus granting an always present account of genuine but tamper-resistant images. Image integrity is easily affirmed by such systems in environments like legal proceedings, media houses, or digital content, where image credibility is a matter of great difference. It tracks changes of an image to report where it comes from and how far it has been manipulated. The proposed system can competently mitigate the menace of high incidences of false positives, which potentially looms large as an inherent challenge to forgery detection. The combination of techniques to enhance image integrity using cryptographic, with those advanced methods of image processing that employ heavy visual intervention, ensures that one has minimized false positives. Therefore, the proposed system is user-friendly, reliable, and easy for real-world applications, wherein only forged samples are flagged for attention. In short, the proposed system consists of MD5, SHA256, and OpenCV which is one of the world’s best solutions for forgery detection in images. This multi-layered paradigm is made to enhance the accuracy as well as reliability of the framework to provide it an application and implementation in various sectors-mostly from media to law enforcement, from data-forensics to those areas where imaging is critical.The system thus comes with balanced and effective detection of image forgery through its cryptographic methods or image-processing techniques, guaranteeing the authenticity of digital images in this digital era.

# Future Enhancement

With respect to future developments, it is possible to make more enhancements to the image forgery detection system that addresses its efficiency and application. While the current-day integration of MD5 and SHA256 hashing algorithms and OpenCV library provides a very strong backbone for forgery detection, this could be further improved if advanced machine learning algorithms come into play. The amalgamation of deep learning models, especially the Convolutional Neural Networks (CNNs), can be used into some very complex forgery detection. CNNs are best known for their powers of distinction when put through fine, pixel-level manipulation that would defeat traditional hashing with CNNs so that a stronger detection could be a chance against their highly skilled image alteration. Moreover, another advancement for the system could be a support from a layer of metadata analysis. Metadata, which comprises timestamps (creation dates), GPS locations, and other camera settings, are oftentimes embedded into image files, therefore giving clues concerning the credibility of an image. Inconsistencies or alterations that may remain unnoticed by the naked eye could be traced by the system through metadata analysis, thereby validating the image integrity and subsequently strengthening the system against informal manipulation. Perhaps the next expected enhancement could be real-time video forgery detection. In the context of rising usage of video content, the manipulation of video frames would provide viability to news footage verification and further safeguard video evidence in legal matters. It can work on the same detection basis of subtractive analysis for every subsequent frame such as flagging any scene heavily spliced or tampered. For greater reach, mobile platforms can be optimized. A mobile version would then allow users to check images on the go, thus rendering convenience and further embedding it into everyday situations.

Another opportunity would be using blockchain as a mechanism to store and verify the authenticity of images in an unalterable manner. The very nature of the unchanging blockchain will secure its own environment more for the set of image forgery detection; thus it becomes a great instrument for sectors that need an assured image verification medium: media, legal, and forensic.

# References

1. Bhatia, P., & Ghosal, S. (2018). "Image Forgery Detection Using Hybrid MD5 and SHA Algorithms." International Journal of Computer Science and Engineering, 7(5), 112-121.
2. Chen, S., & Zhang, Y. (2020). "A Survey on Image Forgery Detection Techniques Based on Visual Features." Journal of Digital Forensics, 12(3), 55-68.
3. Zhang, J., & Wang, Z. (2019). "Forgery Detection in Digital Images Using Hashing Functions and OpenCV." Journal of Image Processing and Computer Vision, 34(2), 98-106.
4. Sharma, S., & Kumar, A. (2017). "Detection of Image Tampering Using MD5 Hashing and Pixel Comparison." International Journal of Computer Vision and Image Processing, 8(2), 76-83.
5. Singh, V., & Arora, R. (2016). "Deep Learning Models for Image Forgery Detection." Proceedings of the International Conference on Machine Learning, 45(4), 234-240.
6. Zhao, L., & Liu, X. (2021). "SHA256-Based Image Authentication and Forgery Detection System." IEEE Transactions on Image Processing, 30(1), 23-34.
7. Patel, H., & Mehta, P. (2018). "OpenCV-Based Real-Time Forgery Detection in Digital Images." Journal of Real-Time Systems, 40(6), 1005-1017.
8. Khan, M., & Ahmed, M. (2017). "Combining Image Hashing with Visual Analysis for Forgery Detection." International Journal of Digital Imaging and Forensics, 18(2), 210-219.
9. Li, X., & Yu, L. (2019). "Enhanced Image Forgery Detection with SHA256 and Visual Feature Matching." International Journal of Image and Graphics, 26(3), 189-200.
10. Singh, A., & Gupta, S. (2020). "Image Forgery Detection Using Hashing and Pixel Analysis." Journal of Information Security, 9(5), 112-119.
11. Kumar, A., & Rani, S. (2018). "Hybrid Image Forgery Detection Using SHA256 and Feature Extraction." Journal of Computer Vision and Image Processing, 20(2), 85-96.
12. Kaur, A., & Arora, N. (2020). "Comparative Analysis of Image Forgery Detection Techniques: A Survey." International Journal of Computer Applications, 12(6), 45-58.
13. Sharma, R., & Mehta, M. (2019). "Security Enhancements in Image Forgery Detection Using OpenCV." International Conference on Artificial Intelligence and Image Processing, 23(4), 512-524.
14. Pati, S., & Nayak, P. (2017). "Image Forensics Using OpenCV and Cryptographic Hashing." Journal of Computer Security and Cryptography, 11(1), 101-115.
15. Ahmed, R., & Singh, V. (2021). "Image Forgery Detection Using Cryptographic and Visual Analysis." Computer Vision and Image Analysis, 28(7), 303-311.