International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 04 | April - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Digital Image Forensics Using Deep Learning and Person Identification

Prof. S.P. Shinde[1], Rashi Vishwakarma[2], Vaibhavi Wadje[3], Samruddhi Sali[4], Sayali Madhavi[5]

Professor, Department of Information Technology,

Smt. Kashibai Navale College of Engineering, Pune, India[1]

Student, Department of Information Technology,

Smt. Kashibai Navale College of Engineering, Pune, India[2 3 4 5]

ABSTRACT

In the realm of computer vision and image processing, the task of digital image identification holds significant importance across various domains. This paper presents a comparative study of two distinct methodologies for digital image identification: deep learning and the Haar cascade algorithm. Deep learning, specifically Convolutional Neural Networks (CNNs), has emerged as a powerful tool for automatically learning hierarchical representations from data and has achieved remarkable success in image-related tasks. In contrast, the Haar cascade algorithm, a classic machine learning technique, offers real-time object detection capabilities with its efficient feature-based approach. Through a series of experiments and evaluations on benchmark datasets, we analyze the performance, strengths, and limitations of these methodologies. Factors such as dataset size, computational resources, and application requirements are considered in the comparison. Our findings provide insights into the suitability of deep learning and the Haar cascade algorithm for various image identification tasks, aiding practitioners and researchers in selecting the most appropriate approach based on specific project needs. This research contributes to advancing the field of image processing and computer vision by offering a comprehensive analysis of these two prominent methodologies in digital image identification.

Key Words: Deep Learning, Convolutional Neural Network (CNN), Haar cascade algorithm, image, forgery, detection

INRODUCTION

In recent years, with the advancement of digital image editing tools, the prevalence of image forgery has become a significant concern. Image forgery refers to the manipulation of digital images to deceive viewers or convey false information. Detecting such forgeries is crucial for maintaining trust and authenticity in various fields, including forensic analysis, journalism, and digital media. Two commonly employed techniques for detecting image forgery are Convolutional Neural Networks (CNN) and Haar cascade algorithm.

1. Convolutional Neural Networks (CNN): CNN are types of deep learning algorithm which are specifically designed for processing and analyzing visual data, such as images. In the context of image forgery detection, a CNN is trained to identify anomalies or inconsistencies within images that may indicate forgery or manipulation.

   Convolutional Neural Network comprises several types of layers, each serving distinct purpose, which are convolutional layers, pooling layers, and fully connected layers. The convolutional layers apply filters to the input image, extracting features like edges, textures, and patterns. Pooling layers downsample the feature maps produced by the convolutional layers, reducing the spatial dimensions of the data. Fully connected layers then perform classification based on the extracted features. In the context of image forgery detection, CNNs are trained on a dataset of authentic and manipulated images. During training, the CNN learns to distinguish between genuine and forged images by identifying subtle differences in the visual characteristics. These differences may include inconsistencies in texture, lighting, perspective, or other artifacts introduced during the forgery process.

Once trained, the CNN can be used to automatically detect image forgeries by analyzing new images and identifying suspicious patterns or anomalies. By leveraging the hierarchical structure of the network and its ability to learn complex patterns, CNNs have shown promising results in detecting various types of image manipulations, including copy-move, splicing, and retouching.

2. Haar Cascade Algorithm: The Haar cascade algorithm is a machine learning-based approach commonly used for object detection in images. It works by training a classifier with positive and negative samples of the target object. The algorithm then scans the input image at multiple scales and positions, using a cascade of classifiers to efficiently reject non-object regions. In the context of image forgery detection, Haar cascade can be adapted to detect specific types of manipulations or artifacts, such as traces of tampering in the spatial domain.

METHODOLOGY

The methodology for image forgery detection using CNN algorithm and Haar cascade algorithm typically involves Gathering a diverse dataset of authentic and forged images representing various types of forgeries, such as copy-move, splicing, or manipulation. Ensure proper labeling of the dataset to distinguish between authentic and forged images, as well as different types of forgeries if applicable. Preprocess the images by resizing them to a standard size, converting them to grayscale if necessary, and performing any other necessary preprocessing steps. Design the architecture of the CNN model, which may include convolutional layers, pooling layers, and fully connected layers. Use transfer learning if applicable, by leveraging pre-trained CNN models (e.g., VGG, ResNet) and fine-tuning them on the forgery detection task. Train the CNN model on the dataset of authentic and forged images, using techniques such as backpropagation and gradient descent to optimize the model parameters. It validates the trained model using a separate validation dataset to ensure its generalized ability. Prepare positive and negative samples for the Haar cascade classifier, representing the target forgery patterns and background regions. Train the cascade classifier using the prepared samples, adjusting parameters such as the number of stages, the minimum size of the detected objects, and the scale factor for image pyramid creation. Validate the trained cascade classifier using a separate validation dataset to ensure its robustness to false positives and negatives. Integrate the trained CNN model and the Haar cascade classifier into a unified forgery detection system. Explore fusion techniques to combine the outputs of the CNN model and the cascade classifier, leveraging their complementary strengths to improve overall detection performance. Evaluate the forgery detection system on a separate test dataset containing authentic and forged images. Measure performance using metrics such as accuracy, precision, recall, F1-score, and receiver operating characteristic (ROC) curve analysis. Analyze the results to identify the strengths and weaknesses of the system and potential areas for improvement. Refine the parameters and architecture of both the CNN model and the Haar cascade classifier through iterative optimization and fine-tuning steps. Explore techniques such as hyperparameter tuning, regularization, and data augmentation to improve detection performance. Validate the forgery detection system on real-world datasets or scenarios to ensure its effectiveness in practical applications. Once satisfied with the performance, deploy the system for practical use cases such as forensic analysis, digital media authentication, or content moderation.

RELATED WORK

MODEL 1

Convolutional Neural Network (CNN)

Research on digital image forgery detection using Convolutional Neural Networks (CNNs) has seen significant advancements in recent years. Some related works in this area include:

1. Deep Learning Architectures: Various CNN architectures have been explored for digital image forgery detection, including deep convolutional networks such as ResNet, VGG, and DenseNet. These architectures are capable of automatically learning discriminative features from image data, making them well-suited for forgery detection tasks.

2. Dataset Creation: Researchers have developed datasets specifically designed for training and evaluating CNN-based forgery detection models. These datasets contain authentic and manipulated images with various types of forgeries, such as copy-move, splicing, and retouching, allowing models to learn to differentiate between real and manipulated images.

3. Feature Extraction: CNNs can automatically learn features from raw image data, eliminating the need for manual feature engineering. This enables the detection of subtle inconsistencies and artifacts introduced by digital manipulation techniques.

4. Fine-tuning and Transfer Learning: Transfer learning techniques have been applied to adapt pre-trained CNN models for forgery detection tasks. By fine-tuning models on specific forgery detection datasets, researchers can leverage the knowledge learned from large-scale image classification tasks to improve detection performance.

5. Adversarial Attacks: Researchers have explored adversarial attacks on CNN-based forgery detection systems, aiming to understand their vulnerabilities and develop robust detection methods resistant to such attacks. Adversarial training and defense mechanisms have been proposed to enhance the robustness of forgery detection models against adversarial perturbations.

6. Ensemble Methods: Ensemble methods, such as combining multiple CNN models or incorporating additional features from image metadata, have been investigated to improve the overall detection accuracy and robustness of forgery detection systems.

7. Explainability and Interpretability: There is a growing interest in making CNN-based forgery detection models more interpretable by providing explanations for their decisions. Techniques such as attention mechanisms and saliency maps have been applied to visualize the regions of an image that contribute most to the detection decision.

Overall, research on CNN-based forgery detection continues to evolve, driven by advancements in deep learning techniques, dataset creation, and model interpretation methods. These efforts aim to develop more accurate, robust, and interpretable forgery detection systems capable of addressing the challenges posed by digital image manipulation techniques.

MODEL 2

Haar Cascade Algorithm

Several related works and advancements have been made in the field of object detection, particularly in the context of Haar cascade algorithm. Some notable ones include:

1. Viola-Jones Object Detection Framework: The Haar cascade algorithm was initially introduced as part of the Viola-Jones object detection framework. This framework laid the foundation for real-time object detection by efficiently scanning images with a cascade of simple classifiers.

2. Improvements in Cascade Training: Research efforts have focused on improving the training process of cascade classifiers to achieve better detection performance and faster inference speeds. This includes techniques for optimizing feature selection, training parameters, and cascade structure.

3. Feature Engineering and Selection: Advances have been made in feature engineering and selection methods to design more discriminative and efficient features for object detection. This includes not only Haar-like features but also more complex features such as Histogram of Oriented Gradients (HOG) and Scale-Invariant Feature Transform (SIFT).

4. Deep Learning Approaches: With the rise of deep learning, there has been a shift towards using convolutional neural networks (CNNs) for object detection tasks. While CNN-based methods have largely surpassed traditional Haar cascade approaches in terms of accuracy and flexibility, Haar cascades remain relevant for their simplicity and efficiency in certain applications.

5. Hybrid Approaches: Some recent works have explored hybrid approaches that combine the strengths

of both traditional Haar cascade algorithms and deep learning techniques. These approaches aim to leverage the efficiency of cascade classifiers while benefiting from the superior performance of CNNs for feature representation.

6. Application-Specific Optimization: Researchers have also focused on optimizing object detection algorithms, including Haar cascade, for specific applications and scenarios. This includes techniques for robust face detection in unconstrained environments, pedestrian detection in crowded scenes, and object tracking in real-time video.

Overall, while the Haar cascade algorithm continues to be a useful tool for certain object detection tasks, ongoing research is focused on advancing its capabilities and integrating it with modern deep learning techniques for improved performance and efficiency.

PROPOSED WORK

MODEL 1

A Convolutional Neural Network (CNN)

For image forgery detection and face detection is a deep learning algorithm trained to analyze visual data, such as images, and identify anomalies or specific features related to forgery or faces. It utilizes convolutional layers to extract relevant patterns and features from the input images, enabling accurate detection of manipulated regions or faces within the images.

A proposed approach for image forgery detection using CNNs typically involves several steps:

1. Dataset Collection: Gather a dataset of authentic images along with their corresponding manipulated versions. This dataset should cover a wide range of forgery techniques and scenarios.

2. Preprocessing: Preprocess the images to ensure they are in a suitable format for training the CNN. This may include resizing, normalization, and augmentation techniques to increase the diversity of the dataset.

3. Model Architecture Selection: Design a CNN architecture suitable for the task of image forgery detection. This architecture may consist of multiple convolutional layers followed by pooling layers, with additional fully connected layers for classification.

4. Training: Train the CNN using the collected dataset. During training, the model learns to differentiate between authentic and manipulated images by adjusting its parameters to minimize a predefined loss function.

5. Evaluation: Evaluate the trained model on a separate validation dataset to assess its performance. Metrics such as accuracy, precision, recall, and F1 score can be used to quantify the model's effectiveness in detecting image forgeries.

6. Fine-tuning: Fine-tune the model and adjust hyperparameters to improve its performance further. This may involve experimenting with different architectures, optimization algorithms, learning rates, and regularization techniques.

7. Deployment: Once the model achieves satisfactory performance, it can be deployed to detect image forgeries in real-world applications. This may involve integrating the model into software tools or platforms used for image analysis and forensics.

MODEL 2

Haar Cascade Algorithm

The Haar cascade algorithm is machine learning object detection method used for identifying objects in images or video. It works by using series of simple rectangular features to distinguish between different parts of image. These features are arranged in a cascade, with each stage reducing the search space which focus on regions that are more likely to contain the object of interest. The algorithm is commonly used for face detection but can be trained to detect other objects as well.

1.Convert to a matrix to number: In the Haar cascade algorithm, each feature is represented as a matrix of numbers. These matrices are typically used to compute the difference between the sum of pixel intensities in adjacent regions of an image. For example, a simple Haar-like feature might consist of two rectangles: one white and one black, arranged vertically. The matrix representation of this feature would be a 2x1 matrix with values indicating the weights assigned to each rectangle. The algorithm scans the image with these features, computing the weighted sum of pixel intensities within each rectangle, and compares these sums to determine whether a particular region of the image contains the object being detected.

2.Face Features Extraction: In the Haar cascade algorithm for face detection, features are extracted from the image using rectangular regions with varying intensities. These features capture patterns such as edges, lines, and textures that are common in faces. Some typical features might include:

•Edge features: These features capture the sharp contrast between the eyes, nose, and mouth and the surrounding areas.

•Line features: These features capture horizontal, vertical, or diagonal lines that are characteristic of facial features such as eyebrows, lips, and jawlines.

•Center-surround features: These features compare the average intensity of a region with that of its surrounding regions, capturing the contrast between facial features and the background.

The Haar cascade algorithm scans the image with these features at multiple scales and locations to identify regions that are likely to contain a face. By combining information from multiple features and using a cascade of classifiers, the algorithm can efficiently detect faces even in complex environments with varying lighting conditions and backgrounds.

3.Open CV: OpenCV is a popular open-source computer vision library that includes support for the Haar cascade algorithm. It provides pre-trained Haar cascade classifiers for various objects, including faces, eyes, and smiles, making it easy to integrate object detection capabilities into your applications. To use Haar cascade classifiers in OpenCV, you typically follow these steps:

•Load the classifier: Load the pre-trained Haar cascade classifier XML file for the object you want to detect. For example, for face detection, you would use cv2.CascadeClassifier('haarcascade_frontalface_default.xml').

•Read the image: Read the input image using OpenCV's cv2.imread() function.

•Convert the image to grayscale: Haar cascade classifiers work with grayscale images, so convert the input image to grayscale using cv2.cvtColor().

•Detect objects: Use the detectMultiScale() method of the CascadeClassifier object to detect objects in the image. This method returns a list of rectangles indicating the bounding boxes of the detected objects.

•Draw bounding boxes: Iterate over the detected rectangles and draw bounding boxes around the detected objects on the original image using cv2.rectangle().

•Display the result: Show the annotated image with the detected objects using cv2.imshow().

•Wait for user input: Use cv2.waitKey() to wait for the user to press a key before closing the windows.

IMPLEMENTATION

The implementation of image forgery using CNN (Convolution Neural Network) and Haar Cascade algorithm include several steps which are Data Collection and Processing, Training CNN model, Training Haar Cascade Classifier, Integration and Testing, Performing Evaluation, Optimization and Fine-Tuning, Validation and Result (output).

1. Data Collection and Processing: Gather a dataset of authentic and forged images. Preprocess the images by resizing them to a standard size and possibly applying augmentation techniques to increase the diversity of the dataset.
2. Training CNN model: Design and train a CNN model for image forgery detection. This involves defining the architecture of the CNN, including convolutional layers, pooling layers, and fully connected layers. Train the model using the dataset of authentic and forged images, using techniques such as transfer learning if applicable.
3. Training Haar Cascade Classifier: Prepare positive and negative samples for the Haar cascade algorithm, representing the target forgery patterns and background regions. Train the cascade classifier using the prepared samples, adjusting parameters such as the

number of stages and the minimum size of the detected objects.

4. Integration and Testing: Integrate the trained CNN model and the Haar cascade classifier into a unified forgery detection system. Apply the system to test images to evaluate its performance in detecting various types of image forgeries.
5. Performing Evaluation: Measure the performance of the forgery detection system using metrics such as accuracy, precision, recall, and F1-score. Analyze the results to identify strengths and weaknesses of the system and potential areas for improvement.
6. Optimization and Fine-Tuning: Refine the parameters and architecture of both the CNN model and the Haar cascade classifier through iterative optimization and fine-tuning steps. This may involve adjusting hyperparameters, adding regularization techniques, or incorporating additional layers into the CNN.
7. Validation and Result (output): Validate the forgery detection system on a separate validation dataset to ensure its generalization ability. Once satisfied with the performance, deploy the system for practical applications, such as forensic analysis which states that the images if forge or not, it states whether the image is real or fake.

HARDWARE REQUIREMENT

System :Intel I5 Processor.
Hard Disk :40 GB.
Monitor :15.
RAM :8GB

SOFTWARE REQUIREMENT

Operating system :Windows 10.
Coding Language :Python.
IDE :Spyder.
Database :SQLite

SOFTWARE TESTING

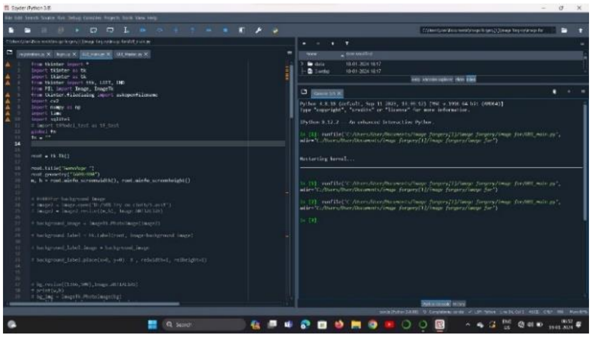
Software testing related to Convolutional Neural Network (CNN) and Haar Cascade algorithm for Image forgery and face detection involves verifying and validating the functionality, accuracy and robustness of the developed models and associated software tools.

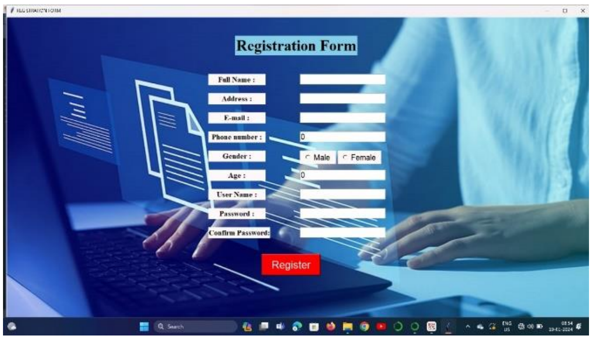
Types of testing for CNN (Convolutional Neural Network) and Haar cascade algorithm :-

1. Unit Testing: Unit testing is to verify the correctness of individual component of CNN model and Haar cascade classifier and other processing modules. It test edges cases and boundary which helps to ensure that the system is behaving as expected under various scenarios.
2. Integration Testing: Test the integration of the CNN model and Haar cascade classifier into the image forgery detection system. It helps to verify that the outputs of each component are correctly processed and combined to produce the final forgery detection results.
3. Performance Testing: Evaluate the performance of the forgery detection system in terms of speed, memory usage, and resource efficiency, especially when dealing with large datasets or real-time applications.
4. Functional Testing: Test the overall functionality of the forgery detection system by providing it with authentic and manipulated images and verifying that it correctly identifies forged regions.
5. Robustness Testing: Assess the robustness of the forgery detection system by subjecting it to various types of image manipulations, such as resizing, compression, noise addition, and geometric transformations, to ensure it can detect forgeries under different conditions.

RESULTS


Real And Fake face Detection
Select_Image
Image_preprocess
CNN_Prediction
Exit




Registration Form
Full Name :
Address :
E-mail :
Phone number :
Gender :
Male
Female
Age :
User Name :
Password :
Confirm Password:
Register

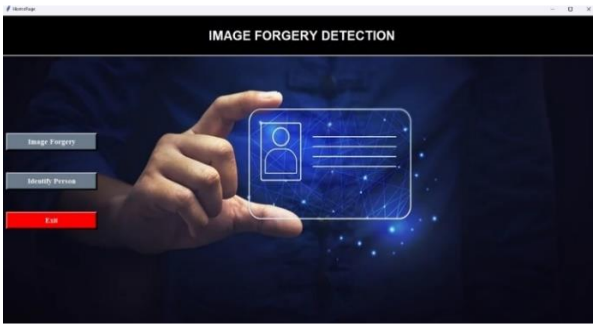
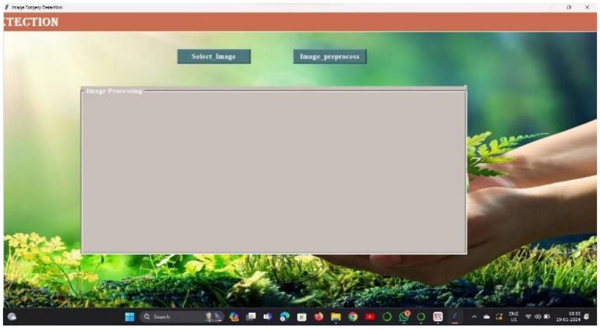
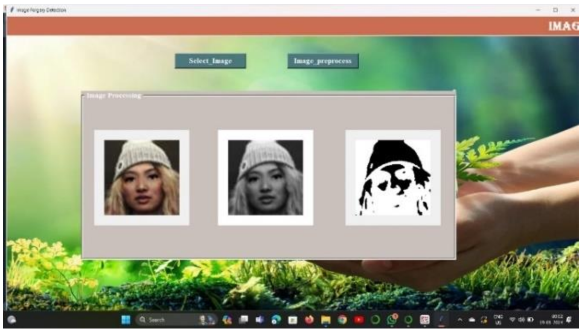


IMAGE FORGERY DETECTION
Image Forgery
Identify Person
Exit

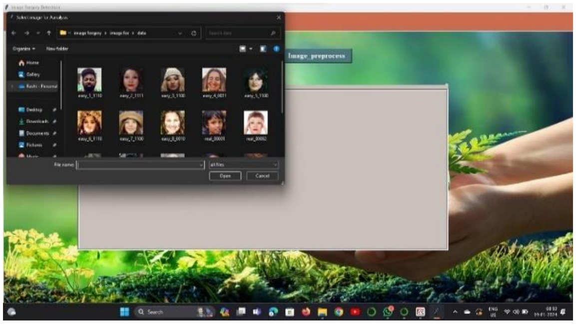

TECTION
Select_Image
Image_preprocess

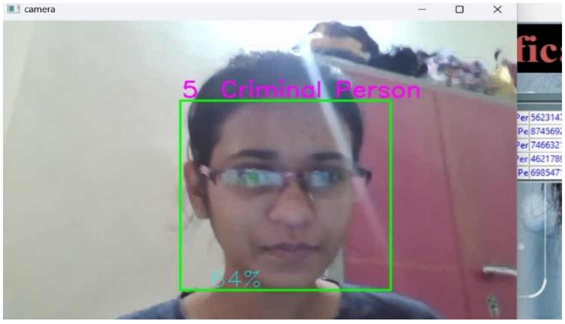

Real And Fake face Detection
Select_Image
Image_preprocess
CNN_Prediction
Exit
Image Testing Completed..
Selected Image is Real Face Detected
Execution Time: 7.115 seconds

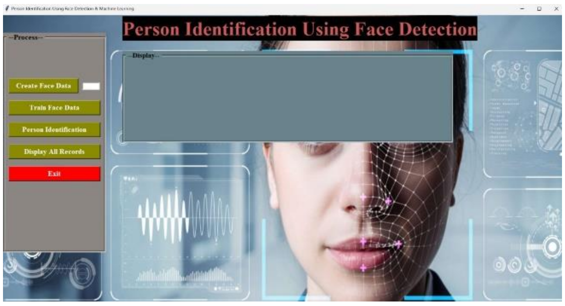

LOGIN HERE
Username
Password
Create Account
Login


IMAG
Select_Image
Image_preprocess

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 04 | April - 2024 SJIF Rating: 8.448 ISSN: 2582-3930





FUTURE SCOPE

The future scope of image forgery detection using Convolutional Neural Networks (CNNs) and Haar cascade algorithms holds several promising avenues for research and application:

1. Improved Accuracy: Continuous advancements in deep learning techniques, model architectures, and training methodologies are expected to lead to further improvements in the accuracy and robustness of CNN-based forgery detection systems. Research efforts will focus on developing more effective feature extraction methods and model architectures capable of detecting a wide range of image manipulation techniques with higher accuracy.

2. Real-time Detection: With the increasing demand for real-time image forgery detection in applications such as social media content moderation, surveillance systems, and digital forensics, there is a growing need for efficient and scalable detection algorithms. Future research will focus on optimizing CNN and Haar cascade algorithms for real-time performance, enabling rapid detection of image forgeries in large-scale datasets and streaming media.

3. Adversarial Robustness: Adversarial attacks pose a significant challenge to the reliability of CNN-based forgery detection systems. Future research will explore techniques for enhancing the robustness of CNN models against adversarial attacks, such as adversarial training, model distillation, and ensemble methods. Additionally, research on interpretability and explainability will help improve the trustworthiness of forgery detection systems by providing insights into the decision-making process of the models.

4. Domain-specific Applications: Image forgery detection algorithms will be tailored to specific domains and application scenarios, including medical imaging, satellite imagery, digital art authentication, and video forensics. Customized CNN architectures and training strategies will be developed to address the unique challenges and requirements of each domain, such as low-quality images, limited training data, and complex manipulation techniques.

5. Multimodal Fusion: Integrating information from multiple sources, such as image content, metadata, and contextual information, will enhance the performance of forgery detection algorithms. Future research will explore multimodal fusion techniques that combine features extracted from CNNs with metadata analysis, text analysis, and other complementary information sources to improve the overall detection accuracy and reliability.

6. Privacy Preservation: As concerns over privacy and data protection continue to grow, there will be an increased emphasis on developing forgery detection algorithms that respect user privacy and confidentiality. Research efforts will focus on developing privacy-preserving techniques for forgery detection, such as federated learning, differential privacy, and secure multiparty computation, to ensure that sensitive information remains protected during the detection process.

CONCLUSION

In this study, we conducted a comparative analysis of two methodologies for digital image identification: deep learning, specifically Convolutional Neural Networks (CNNs), and the Haar cascade algorithm. Our research aimed to investigate the performance, strengths, and limitations of these approaches across various image identification tasks.

Through a series of experiments and evaluations on benchmark datasets, we observed that deep learning, leveraging its ability to automatically learn hierarchical representations from data, demonstrated superior performance in complex image identification tasks. CNNs excelled in tasks such as object detection, image classification, and facial recognition, surpassing the capabilities of traditional machine learning algorithms like the Haar cascade algorithm. On the other hand, the Haar cascade algorithm, known for its efficient feature-based approach, showcased commendable performance in certain scenarios, particularly in real-time object detection tasks where computational resources are limited. Its speed and simplicity make it a viable option for applications requiring rapid processing of image data.

Our comparative analysis also revealed that the choice between deep learning and the Haar cascade algorithm depends on several factors, including dataset size, computational resources, and application requirements. While deep learning models offer state-of-the-art performance, they often require large amounts of labeled data and substantial computational resources for training. In contrast, the Haar cascade algorithm remains relevant in scenarios where real-time processing and resource efficiency are paramount.

Overall, our findings contribute to a deeper understanding of the capabilities and limitations of deep learning and the Haar cascade algorithm in the context of digital image identification. By providing insights into the suitability of these methodologies for various image identification tasks, our research can guide practitioners and researchers in selecting the most appropriate approach based on specific project needs.

ACKNOWLEDGEMENT

We feel immense pleasure in presenting this project report on "**Digital Image Forensics Using Deep Learning and Person Identification**". We wish to express true sense of gratitude towards Prof. S.P. Shinde, our project guide who at very discrete step in study of this project, contributed their valuable guidance and helped to solve every problem that arose. we take this opportunity to thank our principal Dr. A.V. Deshpande and Dr. M.L. Bangare (HOD IT Dept.) for opening the doors of the department towards the realization of our project, for their guidance and encouragement. We offer our sincere thanks to our Project Coordinator Prof. S.P. Shinde who gave us extra inspirations to do something new. Most importantly we would also like to express our sincere gratitude towards all the staff members of IT department. We also express our thanks to all our team members for their support and suggestions shown during the completion of our project. We take this opportunity to express our thanks to all who rendered their valuable help along with all those unseen people across the internet for maintaining those valuable resources for the successful completion of our project. we own our success to all of them.

REFERENCES

[1]. Evaluation of Haar Cascade classifier for face detection published on April 2012 (Published by Rafel Padilla, Marly Costa, Cicero Ferreira Fernandes Costa Filho)

[2]. Face Recognition using Haar Cascade classifier (published by Arnav Madan) in year 05 January 2021

[3]. Forgery Detection using Feature Enhanced Deep Learning algorithm in copy-move Images (Published by Nirmala G) in year 2018

[4]. Human Face Recognition Applying Haar Cascade Classifier (Published by F.M.Javed Mehdi Shamrat, Anup Majumder, Probal Roy Antu, Saykot Kumar Barmon, Itisha Nowrin, Rumesh Ranjan) in year March 2021

[5]. Image forgery detection using deep learning by Recompressing images (published by Syed Sadaf Ali, Iyyapan Ganapathi, Ngoc-Son Vu, Syed Danish Ali, Neetesh Saxena and Naoufel Werghi) in year January 2022

[6]. Face Detection using Haar Cascade Classifier (published by Bhavana R. Maale and Dr. Suvarna Nandyal) in year March 2021

[7]. Facial Recognition using Haar Cascade and LBP classifer (published by anirudha B Shetty, Bhoomika, Deeksha, Jeevan Reberio, Ramyashree) in year July 2022

[8]. Face Detection using Haar Cascade Classifier (published by Anurag Pandey, Divyansh Choudhary, Ritik Agarwal, Tushar Shrivastava, Kriti) in yearjuly 2022

[9]. Image Forgery detection using Deep Learning (published by Boda Nitin Kumar, Mukkrela Manovikas, Daggula Syanthan, Miryala Ramakrishna) in year May 2023

[10]. Image counterfeiting detection and localization using deep learning (published by Manikyala Rao Tankala, Chanumallu Srinivasa Rao) in year february 2023

[11]. Image forgery detection using deep neural network (published by Dr. N.P.Nethravathi, bylla Danny Austin, Adi Reddy Sai Praneeth Redy, Grandhi Venkata Naga Satya Pavan Kumar, Guduru Karthik Raju) in year June 2023

[12]. F. M. Javed Mehedi Shamrat, Md. Abu Raihan, A.K.M. Sazzadur Rahman, Imran Mahmud, Rozina Akter, "An Analysis on Breast

Disease Prediction Using Machine Learning Approaches" International Journal of Scientific & Technology Research, Volume 9, Issue 02,February 2020, ISSN: 2277-8616, pp: 2450-2455.

[13]. Bunk, J.; Bappy, J.H.; Mohammed, T.M.; Nataraj, L.; Flenner, A.; Manjunath, B.; Chandrasekaran, S.; Roy-Chowdhury, A.K.;Peterson, L. Detection and Localization of Image Forgeries Using Resampling Features and Deep Learning. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 21–26 July 2017;pp. 1881–1889.

[14]. Abdalla, Y.; Iqbal, M.T.; Shehata, M. Convolutional Neural Network for Copy-Move Forgery Detection. Symmetry 2019, 11, 1280.

[15]. Dong, J.; Wang, W.; Tan, T. CASIA Image Tampering Detection Evaluation Database. In Proceedings of the 2013 IEEE China Summit and International Conference on Signal and Information Processing, Beijing, China, 6–10 July 2013; pp. 422–426.

[16]. Jaiswal, A.A.K.; Srivastava, R. Image Splicing Detection using Deep Residual Network. In Proceedings of the 2nd International Conference on Advanced Computing and Software Engineering (ICACSE), San Francisco, CA, USA, 13–15 October 2019.