ISSN: 2182-2069 / E-ISSN: 2182-2077

Analysis on Video Forgery Detection Using Spatial Temporal Feature with 3D CNN

Hemant Appa Tirmare[1*], Dr. Jaydeep B. Patil[2], Dr. Sangram T. Patil[3],
Dr. Vidyullata Vinayak Devmane[4], Dr. Anil M. Hingmire[5], and
Dr. Shashikant Sudhakar Radke[6]

[1*]Assistant Professor, CST, Department of Technology, Shivaji University, Kolhapur, Maharashtra, India; Ph.D. Scholar D.Y. Patil Agriculture & Technical University, Talsande, Kolhapur, Maharashtra, India. hat_tech@unishivaji.ac.in, https://orcid.org/0000-0001-8451-935X

[2]Associate Professor & Associate Dean, Computer Science & Engineering Department, D.Y. Patil Agriculture & Technical University, Talsande, Kolhapur, Maharashtra, India. jaydeeppatil@dyp-atu.org, https://orcid.org/0000-0001-7827-2805

[3]Professor, Computer Science & Engineering Department, D.Y. Patil Agriculture & Technical University, Talsande, Kolhapur, Maharashtra, India. sangrampatil@dyp-atu.org, https://orcid.org/0000-0002-9725-8678

[4]Professor, Department of Computer Engineering, Shah & Anchor Kutchhi Engineering College, Mumbai, Maharashtra, India. vidyullata.devmane@sakec.ac.in, https://orcid.org/0000-0002-2274-3943

[5]Assistant Professor, Computer Engineering Department, Vidyavardhini's College of Engineering and Technology, Mumbai, Maharashtra, India. anil.hingmire@vcet.edu.in, https://orcid.org/0000-0002-8925-2962

[6]Assistant Professor, Computer Engineering Department, Shah & Anchor Kutchhi Engineering College, Mumbai, Maharashtra, India. shashikant.radke@sakec.ac.in, https://orcid.org/0000-0002-4305-1831

Received: October 04, 2025; Revised: November 13, 2025; Accepted: January 05, 2026; Published: February 27, 2026

Abstract

The substantial increase in video recording and sharing in a short time, in addition to the easy-to-use editing tools, has created a strong necessity for efficient video forgery detection methods. The range of video manipulations, such as the usage of deep fakes, tampering of frames, and splicing, could endanger people's trust, cybersecurity, and the general stability of the digital media world. Previous works sometimes used temporal or spatial features separately, which was often not enough to accurately detect. This research paper proposes a new method based on 3D Convolutional Neural Networks (3D-CNNs) that captures the spatial as well as the temporal aspects of video frames simultaneously. The outlined architecture is trained on a face manipulation dataset, which includes real and fake videos, gathered from the Face Forensics repository. By distinguishing such minor areas of the video that have been changed, the system is capable of achieving high-performance levels in various forgery types. The first set consisted of two hundred videos,

Journal of Internet Services and Information Security (JISIS), volume: 16, number: 1 (February-2026), pp. 570-594.
DOI: *10.58346/JISIS.2026.I1.032*

*Corresponding author: Assistant Professor, CST, Department of Technology, Shivaji University, Kolhapur, Maharashtra, India; Ph.D. Scholar D.Y. Patil Agriculture & Technical University, Talsande, Kolhapur, Maharashtra, India.

while the second set comprised three hundred videos. In the condition with the three hundred-video subsets, the Model demonstrated an accuracy level of 86%, precision of 0.87, recall of 0.86, and F1-score of 0.86 (weighted average). In the case of the two-hundred-video subsets, the test results were even better, with accuracy reaching 90%, precision being 0.92, recall 0.90, and F1-score 0.90. Compared with traditional classifiers like SVM, Random Forest, and LSTM models, the 3D-CNN method was always better judged by metrics such as accuracy, precision, and recall. These findings emphasize how powerful 3D-CNNs are in identifying spatiotemporal irregularities, thus enabling a more reliable and safer way to validate digital video content authenticity.

Keywords: Video Forgery Detection, Deepfake Detection, 3D Convolutional Neural Networks (3D-CNNs), Spatiotemporal Features, Machine Learning (ML), Temporal and Spatial Irregularities, Face Forensics Dataset, Deep Learning (DL).

1 Introduction

The widespread use of audiovisual content on the internet has a significant impact on communication, entertainment, and information sharing. However, this development has given rise to the misuse of video editing technology, resulting in the prevalent problem of video forgeries. Video content has become central to communication in modern times as the graph of electronic media has steepened, and the number of users of video-sharing service providers has risen rapidly all over the world.

Conversely, the extensive availability has also seen the rise in the cases of video forgery whereby malicious characters manipulate video content to deceive others. Examples of ways that pose a threat to the security of the digital world are deep fakes, frame insertion, and object removal, whose consequences can vary in the misinformation and defamation to security breaches and legal problems (Cozzolino et al., 2017; Güera & Delp, 2018). Development of trustworthy and effective mechanisms of video forgery detection has emerged as an important field of study. Conventional methods of video forgery detection typically rely on human inspection or pre-established algorithmic criteria, resulting in slow processes and a higher likelihood of errors. With the advent of Machine Learning (ML) and Deep Learning (DL) technologies, scientists have turned to exploring data-driven approaches for the automatic detection of falsified videos with high precision (Nguyen et al., 2022; Inayathulla & Rajasekhara Rao, 2025; Li et al., 2020). Initially, these methods turned to massive volumes of real and fake films to build models that uncover the slight differences and the traces of tampering in the given videos. Scholars have recently achieved great success in using recurrent and convolutional neural networks to detect frame-level manipulations and temporal anomalies in videos, with research output being highly impressive (Afchar et al., 2018; Sowmya & Vibin, 2023).

Despite advancements in this field, detecting falsified videos remains challenging because counterfeit creation techniques are constantly improving, and systems must possess the ability to generalize in order to adapt to various types of manipulations. This paper reviews the state-of-the-art research on video forgery detection through machine learning, highlighting the pros and cons of the existing approaches (Malhotra & Mehra, 2021). The unresolved problems and possibilities of the research field are identified by highlighting the need for collaboration across different disciplines and the creation of standard evaluation frameworks.

Methods for falsifying videos, including deepfake generation, frame interpolation, and object insertion or removal, are advancing to a point where detection by a human has become nearly impossible. Recognition of video forgery has become a hot research topic with potential applications in digital forensics, media authentication, and security.

Conventional methods of detecting video forgery usually depend on the use of manually designed features or statistical analysis, which can be generalized across various types of forgeries and video formats. Although 2D-CNNs have proven effective in detecting image forgeries, the application to video data is limited due to the inability to capture the temporal aspect. Indeed, videos are spatiotemporal in nature, and the temporal consistency between the frames is one of the most important features used in manipulation detection. This makes 3D Convolutional Neural Networks (3D-CNNs) very suitable for this task (Praveenraj et al., 2024).

The 3D-CNNs can analyze both spatial and temporal data by shifting the convolution operations to the time dimension.

The paper is structured in the following way, Section 2 will review similar studies, standard methods, and deep learning-based video forgery detectors. Section 3 explains the datasets used, how they were obtained, what was contained in them and the kind of forgeries involved. Section 4 describes the proposed approach, such as data preparation, training approaches, and 3D-CNN architecture. The results are reported in Section 5, which proves the performance of the model in accuracy, precision, and recall of the various types of forgery. Lastly, the Section 6 refers to the key findings, contributions, and the perspectives of future research.

Key Contributions

- In the paper, a new mechanism of video forgery detection based on 3D Convolutional Neural Networks (3D-CNNs) is presented, where both spatial and temporal data is included to increase the detection levels of forgery, such as deepfakes, frame manipulation, and splicing.
- The proposed model is highly performing with accuracy of 86 and 90 on 300-sample and 200-sample dataset respectively, which outperforms orthodox approaches such as SVM, Random Forest, and LSTM.
- The model has been tested on a face manipulation dataset of Face forensics repository, which has proven that it can be used in real world use of detecting video forgery.

2 Related Work

Due to the increase in video counterfeiting methods, sophisticated detection approaches leveraging deep learning and machine learning algorithms are presently mandated. Feature extraction techniques such as Local Binary Patterns (LBP), Gray-Level Co-occurrence Matrix (GLCM), and color histograms have traditionally been the primary tools of influence on the conventional methods. These techniques are generally accompanied by such classifiers as Support Vector Machines (SVM) and Random Forest (RF) to detect forgeries. Singh et al., (2025) have pointed out that these standard methods will gradually be exhausted by 2024 in the face of the most elaborate counterfeit schemes since these methods are incapable of capturing the fine-grained temporal and spatial anomalies in manipulated videos. To address these limits, Singh et al., (2025) explored the use of Convolutional Neural Networks (CNNs) in video forgery detection. The research found out that Two-Layer Hybridized Deep CNN classifier can effectively learn the discriminatory features of most informative frames and thereby substantially enhance the accuracy of forgery detection (Ugale & Midhunchakkaravarthy, 2025). Makandar et al., (2024) proposed Attention-Augmented CNNs (AACNNs) that adopt both local and global attention mechanisms to facilitate the detection of forgery-prone regions in a video. Liang et al., (2024) went further in this work by integrating Res-Next CNN with Long Short-Term Memory (LSTM) networks, thus making the Model capable of recognizing not only the spatial features but also the temporal dependencies for the deepfake identification task. Kumar et al., (2022) introduced a Spatio-Temporal Dual Transformer Network that merges with the help of both spatial and temporal hints the generalization across different datasets and thereby achieves higher performance in the task of altered video detection (Arun et al., 2024). These publications also highlighted the importance of preprocessing operations such as key frame extraction and YCrCb color space conversion, which help reduce computing complexity and improve feature analysis.

Liu et al., (2022) took the same route by using entropy-based texture features like DistrEn2D and MSE2D for evaluating inter-frame correlation consistency and thus were able to identify various types of video fraud effectively. Shelke & Kasana, (2022) introduced a U-Net-based CycleGAN framework for pinpointing the exact locations of the forged regions in the video frames, thus enabling a thorough forensic examination of the fabricated videos. In order to make the detection more resistant, El-Shafai et al., (2024) employed knowledge distillation techniques and Discrete Cosine Transform Multi-Scale (DCTMS) methods for capturing fine-grained texture and structural changes across different compression levels, thus allowing their Model to have better generalization capability. Girish & Nandini, (2023) combined the UFS-MSRC method with LSTM networks, leading to higher accuracy in the detection of temporal forgery through enhanced feature selection and sequential dependency analysis.

It is worth noting that, despite all these technological breakthroughs, the issue of generalizing forgery-detection models to different datasets remains unsolved. Mohiuddin et al., (2023) pointed out that numerous models show a decline in performance when verified on newly manipulated videos, limiting their reliability for practical applications. Bourouis et al., (2020) highlighted real-time processing capability as a primary concern alongside the detection quality and claimed that research in the future should be devoted to the development of scalable and efficient detection frameworks. To stay authentic digital video content, it becomes essential to create more robust, dataset-independent, and automated solutions as video alteration methods get improved. Various methods to detect video fraud and to evaluate deepfakes have been the focus of recent studies in this field. Tokas et al., (2023) introduce a W-Net-based approach for video forgery localization, which was able to ascertain in simple as well as complicated situation the forgery quite effectively. The results show the importance of the employment of deep learning algorithms for the extraction of manipulations in the content.

Zhong et al., (2024) developed a deep learning model to detect video counterfeiting using the intra-frame copy-move method, demonstrating that deep learning approaches outperform traditional human techniques. Vaishali & Neetu, (2024) conducted a comparison analysis on deep-fake detection via transfer learning, where MobileNetV2 achieved the most fantastic accuracy of 89%, followed by ResNet50 at 83%, using the Face Forensics++ dataset.

Heidari et al., (2024) have studied the identification of deepfake videos through unsupervised learning models and have shown the ability of ensemble methods to raise the detection precision to a great extent. Ghiurău & Popescu, (2024) analyzed the impact of deep learning in recognizing artificially generated content and figured out that current neural networks are far more advanced than conventional methods of detection in this regard. Singh et al., (2025) came up with a hybrid deep learning and machine learning system to recognize the spatial and temporal manipulation in the movies, which is also strong in real-world scenarios. Ch et al., (2024) focused on detecting image manipulation using machine learning algorithms and emphasized the need for sophisticated detection methods. The research Al-Dulaimi & Kurnaz, (2024) demonstrated that transfer learning can be a very effective tool in deepfake detection if one makes a comparative study of the results obtained with and without transfer learning. Mansoor & Iliev, (2024) presented advances and problems in deepfake detection, and further future research areas for addressing deepfake threats. Finally, Tirmare & Patil, (2024) proposed a temporal residual network-based solution for video forgery detection, proving its capacity to increase the efficacy in identifying modified video content.

3 Categories of Forgeries

Methods of video alterations can be roughly classified into inter-frame and intra-frame. Inter-frame forgery typically means the changes of the temporal flow of the video by means of such manipulations as insertions, removals, or repetitions of frames. Whereas intra-frame forging modifies the visual content of the single frame with the help of techniques like copy-move or splicing, and at the same time keeps the total video continuity intact.

Deepfake-Based Forgery Detection

This section covers recent deepfake-based forgery detection developments. Researchers have come up with a variety of distinctive frameworks and plans to solve the problem of detecting fake data in videos. The Forgery Clue Augmentation Network (FCAN-DCT) gets the spatial and temporal features with the help of frequency information more accurately. It is made up of two modules: "Compact Feature Extraction" (CFE) and "Frequency Temporal Attention" (FTA), as well as a backbone network. Two datasets, "Wild Deepfake" and "Celeb-DF" (v2), were reviewed, together with the presentation of a self-created "Deepfake NIR," the first video forgeries dataset based on the near-infrared modality (Akhtar et al., 2022).

Rapid Detection of Face Tampering: A technology that can be used to identify Deepfake and Face2Face AIs in videos. It analyzes quick detection networks with publicly-available datasets as well as recently-created datasets based on web videos (Verdoliva, 2020). Face Forensics using GANs: A convolutional neural network created to enhance forgery detection of faces by the use of generative adversarial networks for the production of synthetic faces of different resolutions and sizes in order to make the data augmentation easier (Ding et al., 2022). A deep face recognition model through feature extraction accomplishes its task by transferring weights. Fine-tuning allows the network to accurately categorize real and false photos, producing satisfactory results with the AI Challenge validation data (www.kaggle.com).

Frame-Based Forgery Detection

Researchers have also focused on the detection of frame duplications. There are many methods that have been implemented to spot and wipe out the fakes via frame-level scrutiny:

Deep learning framework for frame duplication. A breakthrough deep learning model that merges "Inflated 3D" (I3D) with a "Siamese-based Recurrent Neural Network" (RNN). The system uses the I3D network to break videos into frames, extract features, and detect frame-to-frame duplication.

Feature extraction and clustering: A two-stage process that employs "Scalar Invariant Feature Transform" (SIFT) for feature extraction and "Mean Shift Clustering Algorithm" (MSCL) for clustering of similar object frames from video. The method involves the following stages.

The Model provides detailed information on the number and the positions of the falsified frames in the video.

This program uses image processing techniques to improve the detection of forgery.

Copy-Move Forgery Detection

Copy-move forgery detection algorithms are a set of tools that aim at the identification of copied and altered parts in digital photos. Here is a list of some primary techniques and their tactics: Pixel-Based Copy-Move Detection: A methodology for recognition of digital image authenticity. Change the colored image into a grayscale one. The grayscale image was split into overlapping 8 x 8 segments. Features are obtained by the Discrete Cosine Transform (DCT) from different feature sets. Cluster the blocks by means of the K-means algorithm. Radix sort is used for feature matching. Statistical and multifractal feature-based methods: A novel technique that uses statistical and multifractal factors as key distinguishing characteristics. The photos are divided into non-overlapping, fixed-size blocks. Common characteristics of each block were found, and the blocks were categorized by a metaheuristic method. Besides that, a semi-metric function was constructed to compare block similarities. The results of the experiments show that this method yields high precision and recall at a low computational cost. Super Pixel-Based Copy-Move Detection: The method splits the picture into the complex and the smooth parts using k-means clustering. Scale-Invariant Feature Transform (SIFT) features are used to locate the forgery in a complex environment. b. The sector mask function and RGB color features are used to detect forgery in smooth areas. Filtering is done on both types to get rid of wrong matches, thus ensuring reliable detection of copy-move fraud.

Object-Based Forgery Detection

Object-based forgery detection tools use object tracking and movement analysis to identify the replaced parts in videos. The following are the main approaches: Object-based Frame Identification Network: One method uses symmetrically overlapping motion residuals to enhance the contrast between video frames. Motion residuals in the video are obtained from the overlapped temporal frames using the temporal oscillations in the video stream and then fed to a deep neural network. This method helps to achieve high detection performance by focusing on the temporal changes. Passive Video Forgery Method: A method for detecting insertion and deletion forgeries in videos. It uses arbitrary core tensors that are adjusted orthogonally to reduce the data and provide good features for tracking forgeries across the video. The experimental results show that this method can detect both types of forgeries with an accuracy of up to 99%.

Multiple-Stream Framework: This is focused on detecting object-based video forgery, which is the source of multi-stream feature extraction. After that, a Dual-Stream feature fusion, Conditional Random Field (CRF), helps to refine the segmentation. Finally, video tracking and depth information are used for a consistent and fine-tuned placement of falsified objects.

Sequential and Patch Analysis: This focuses on the detection of video forgery involving the removal of an object and focuses on its various sections. Video sequences are modeled as a form of a stochastic process, allowing for the identification of such anomalies through the monitoring of evolving properties. The significant volume of the mentioned detection techniques points to the fact that, mainly, the adoption of deep learning CNNs, RNNs, and hybrid models has markedly improved the effectiveness of video forgery detection.

Nonetheless, current approaches still lack generalization between datasets and real-time constraints, and still struggle to detect sophisticated forms of forgery (deepfakes, frame manipulation). The current research seeks to alleviate the foregoing issues through the deployment of 3D Convolutional Neural Networks (3D-CNNs), which can effectively detect temporal and spatial irregularities within a video. Unlike other methods that focus on analyzing each frame of video individually, a 3D-CNN merges information over various frames, allowing it to learn the semantics and patterns of movement over extended time frames and making it better at detecting and analyzing the small nuances of edited content within a video. Using a broad collection of altered and original videos will help the Model increase its robustness and find complexities with a lower level of false favorable rates and better performance for real-time tasks.

4 Proposed Methodology

While the video forgery detection paradigm is fairly novel within the domain of video analysis, existing algorithmic solutions have many limitations, including slow detection of complex manipulations, insufficient analysis of the spatial and/or temporal characteristics of the video, and inadequate defenses against contemporary forgery techniques such as deepfakes and frame insertion/deletion. The proposed methodology, as shown in Figure 1, solves the problem by integrating a 3D Convolutional Neural Network (3D-CNN). It allows for the analysis of video sequences while overcoming the limitations of previous systems by analyzing the spatial and temporal characteristics of the video simultaneously, rather than individually.

The Model should be able to perform better at forgery detection due to the inclusion of deep feature extraction in the analysis of motion within the video.

The key distinction of this technique from others is its unique ability to perform deep spatiotemporal feature learning, enabling it to not only detect where changes occur but also to adapt to volatile changes in the video. The proposed methodology has been shown to enhance detection accuracy, improve generalization across various types of forgery, and be applicable in real-world outdoor scenarios.

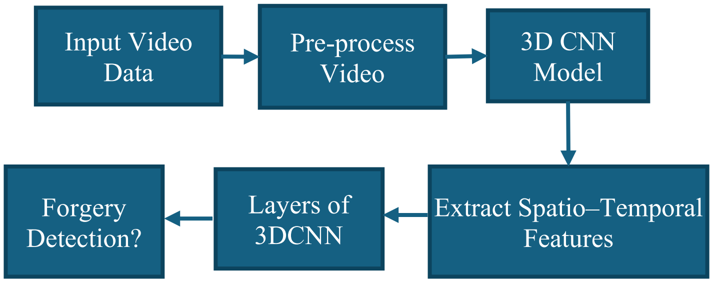

Figure 1: Block diagram for methodology

Dataset

This dataset is a perfect source for deepfake detection exercises, as it has a full range of video sequences that can be used to develop and test deep learning models that can identify media tampering. It was retrieved from the official Face Forensics server, which provides high-quality datasets designed exclusively for face alteration detection (Gong et al., 2025). The primary motivation for this dataset is to close the gap in the availability of video-based original datasets for deepfake detection. Most existing datasets focus on image-based alterations, leaving a crucial need for accessible video datasets that can be used to train and evaluate models. This data set solves that gap by providing a large volume of both original and altered video information. The dataset for this study consists of video sequences divided into training and testing subsets. The data is structured as follows.

Data Set 1: Training Data: Training data consists of 207 30-frame video clips. The frames are resized to 128 128 pixels and grayscale (single channel) is used. Each sample has the form (30, 128, 128, 1).

This data was used to train the Model, allowing it to comprehend the temporal and spatial features required for video forgery detection.

The testing set consists of 61 samples; each produced with 30 frames of 128×128 pixels in greyscale. The format for the testing data is (61,30,128,128,1). This collection of data was employed to evaluate the Model's efficiency and its ability to generalize to previously seen video sequences. In total, there are 268 video clips in the dataset: 207 are meant for training and 61 for testing, thus there is a relatively balanced number of video samples for the training and evaluation of the models.

Dataset 2: The provided data serves as a guide on how to efficiently bring in and preview data in a machine learning setting. The training example has 239 data points, and the testing data sample has 71 data points, which is quite a typical way of splitting data for training and testing.

Each record in the dataset is a 5D tensor of shape (30, 128, 128, 1). The data is thus 3D and contains frames of video or 3D medical scans. The first dimension is the temporal dimension and is of length 30 (the number of frames, or time steps). The spatial shape is 128x128. The last dimension represents the number of channels. A 1 means the image is in black and white. The consistency of training and testing data shapes promotes compatibility during model evaluation. This structure is well-suited for models like 3D Convolutional Neural Networks, which are designed to capture both temporal and spatial data well.

Data Extraction and Preprocessing

To prepare the video data for the Model, two crucial functions were implemented: one for loading the video data from the directory and another for extracting frames from videos. The extract frames class uses OpenCV's cv2—Video Capture class to read the videos one at a time from the file to extract the video frames. In order to save on processing power, each frame is resized to a resolution of 128 x 128 and made grayscale. The frames are first temporarily saved in a Python list, and after that, converted into a NumPy array to save computing time for the following processing.

$$\mathrm{I\ gray} = 0.2989 \cdot \mathrm{R} + 0.5870 \cdot \mathrm{G} + 0.1140 \cdot \mathrm{B} \qquad (1)$$

In Equation 1, where R, G, and B are the values for intensity of the red, green, and blue color channels, respectively. The values of the coefficients (0.2989, 0.5870, 0.1140) are determined from human visual sensitivity, where green is more sensitive than red and, to a lesser extent, blue.

A 3D convolutional layer with a kernel $K \in \mathbb{R}^{k_t \times k_k \times k_w \times C_{\mathrm{in}} \times C_{\mathrm{out}}}$ operating on an input clip $X \in \mathbb{R}^{T \times H \times W \times C_{\mathrm{in}}}$ produces output $Y \in \mathbb{R}^{T' \times H' \times W' \times C_{\mathrm{out}}}$ with elements: In (Equation 2)

$$Y_{t,i,j,c} = \sum_{u=0}^{k_t-1} \sum_{v=0}^{k_b-1} \sum_{w=0}^{k_w-1} \sum_{c'=0}^{C_{in}-1} K_{u,v,w,c',c} X_{t+u,i+v,j+w,c'} + b_c \qquad (2)$$

After each conv block: batch-norm, activation $\sigma(\cdot)$ (ReLU), and optional temporal pooling. Using 3D kernels lets filters learn spatiotemporal patterns jointly rather than separately. See lightweight 3D-CNN designs for deepfake detection for guidance on compact kernels and channel transforms.

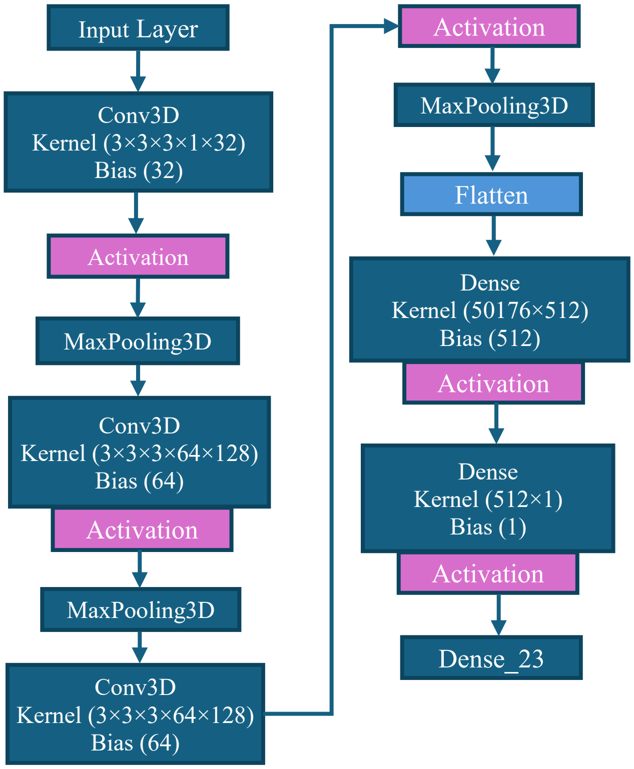

Figure 2: Layer Details of the 3D-CNN model for video forgery detection

The load data function processes movies stored in the given directory and prepares labeled datasets. It sequentially processes the "manipulated" and "original" directories, thus annotating the videos from the former with one and those from the latter with 0. The extract frames method extracts the frames of the videos that are handed over to it. It only considers the videos that have at least 30 frames, and from those, the 30 frames with the highest indices are selected for the dataset. The video sequences that have been created are stored together with their corresponding labels in arrays, which are then returned as the final dataset. This setup phase is a mediator step between video loading and model training, through which it is checked both visually and logically that all videos were uniformly processed and are in a suitable state for model training. T786   P;.M 91`4   4DFA+sahe dataset has been split into training and validation sets with the support of the train-test split method from sklearn for the purpose of an honest evaluation during training. In this way, 20% of the training data is taken as a validation set (test size=0.2); thus, the Model's performance can be checked on unseen data during training. The split is reproducible due to a fixed random state being used. The processing pipeline by itself conforms to the framework depicted in the pseudo-code with the input video tensor successively processed by three Conv3D, ReLU, and two Max Pooling blocks to produce spatiotemporal features as shown in Figure 2.

Pseudo Algorithm

```
# Input layer
input_data = Input(shape=(depth, height, width, channels))
# First Conv3D block
x = Conv3D(filters=32, kernel_size=(5, 3, 4), strides=(5, 1))(input_data)
x = Activation('relu')(x)
x = MaxPooling3D(pool_size=(2, 2, 2))(x)
# Second Conv3D block
```

```
  x = Conv3D(filters=64, kernel_size=(5, 3, 4), strides=(5, 2))(x)
  x = Activation('relu')(x)
  x = MaxPooling3D(pool_size=(2, 2, 2))(x)
  # Third Conv3D block
  x = Conv3D(filters=128, kernel_size=(5, 3, 4), strides=(5, 6))(x)
  x = Activation('relu')(x)
  x = MaxPooling3D(pool_size=(2, 2, 2))(x)
  # Flattening
  x = Flatten()(x)
  # Fully connected layers
  x = Dense(units=512, activation='relu')(x)
  output = Dense(units=1, activation='sigmoid')(x) # or 'linear' for regression
  # Build model
model = Model(inputs=input_data, outputs=output)
```

Model Training

The 3D CNN model is trained on the training dataset (X_train, y_train) across ten epochs using an 8-batch size. During each epoch, the Model is validated against the validation set (X_val, y_val). This step enables tracking of the learning process and early detection of overfitting or underfitting.

1. The 3D convolutional layers (conv3d, conv3d_1, and conv3d_2) of the Model are designed to pick up spatial as well as temporal features. To me, it looks like each Conv3D layer "costs" lowers the spatial dimensions while deepening (number of filters) the feature map."
2. MaxPooling3D Layers: Max pooling layers that are placed after each stage of the Conv3D Layers cut down on the size of the area from which prominent features are picked and also lower the computational effort.
3. Flatten Layer: A flatten layer can take the output that comes from the convolutional and pooling layers, and it will then be able to use that as a 1D vector for dense layers.
4. Dense Layers: A dense layer that has 512 units will be the one to work with the flattened features. A final thick layer with only one unit is there to provide the output.

Parameter Analysis

Parameter Analysis The first dense layer (25,690,624 parameters) mainly holds a connection between the high-dimensional flattened features and the smaller dense layer and, therefore, has the most significant number of parameters. The convolutional layers (conv3d, conv3d_1, and conv3d_2) are responsible for a small but significant fraction of the parameters.

Output Shapes

Input spatial dimensions are successively minimized through the use of the Conv3D and MaxPooling3D layers. After the last MaxPooling3D layer, the spatial dimensions are reduced to a very low degree (2, 14, 14), and the dense layers have thus become capable of efficient computation.

Table 1: 3D CNN and layer

Layer (type)	Output Shape	Param
conv3d (Conv3D)	(None, 28, 126, 126, 32)	896
max_pooling3d (MaxPooling3D)	(None, 14, 63, 63, 32)	0
conv3d_1 (Conv3D)	(None, 12, 61, 61, 64)	55,360
max_pooling3d_1 (MaxPooling3D)	(None, 6, 30, 30, 64)	0
conv3d_2 (Conv3D)	(None, 4, 28, 28, 128)	221,312
max_pooling3d_2 (MaxPooling3D)	(None, 2, 14, 14, 128)	0
flatten (Flatten)	(None, 50176)	0
dense (Dense)	(None, 512)	25,690,624
dense_1 (Dense)	(None, 1)	513

Table 1 shows the 3D CNN architecture employed for handling spatiotemporal data, e.g., video or volumetric data. The Model takes off with a Conv3D that uses 32 filters to perform 3D convolution, thereby lowering the input's spatial dimensions and giving an output shape of (None, 28, 126, 126, 32). A MaxPooling3D layer follows this, and the feature maps are reduced to (None, 14, 63, 63, 32). The second Conv3D layer of the Model utilizes 64 filters to get the features and provide the output shape of (None, 12, 61, 61, 64). The MaxPooling3D layer that follows it decreases the dimensions to (None, 6, 30, 30, 64). The third Conv3D layer of the Model employs 128 filters to capture a more comprehensive feature and give the output shape of (None, 4, 28, 28, 128), which is then down-sampled by the final MaxPooling3D layer to (None, 2, 14, 14, 128). The Flatten layer converts the 3D feature maps into a 1D vector of 50,176 elements, thus making the data ready for fully connected layers. The first Dense layer is made up of 512 units and, therefore, is able to integrate the features, resulting in the output shape (None, 512). This layer is the one that holds almost all of the Model's parameters (25,690,624) and thus the main reason for its computational cost. The last but one Dense layer, which has a single unit, is the one from which the output of the Model is derived. Often, tasks such as binary classification or regression can be accomplished through this layer, which has only one unit. This model architecture is designed to effectively capture spatiotemporal features, while the pooling layers serve to diminish the dimensionality and the dense layers, to make decisions.

However, the vast number of parameters, particularly in the dense layers, can require enormous computational resources.

After final 3D conv blocks, apply spatiotemporal global pooling to produce a fixed-length vector:

$$z = \text{ Flatten } \left(\text{GAP}_{t,i,j}(\mathcal{F}_\theta(V))\right) \in \mathbb{R}^d \qquad (3)$$

In equation (3), where $\text{GAP}_{t,i,j}$ denotes global average pooling across time and spatial dimensions. Optionally concatenate handcrafted features h (e.g., face quality metrics, motion-consistency scores) so the final input to XGBoost is $\grave{z}$ $[z; h]$.

Model Evaluation on Test Data

Following training, the Model is assessed on the test dataset (X_test, Y_test). The predictions (y pred 3 d cnn) are produced with the predict method of the Model and transformed to binary values (Original or Manipulated) with the help of 0.5 threshold. The general accuracy of the Model is calculated using the assistance of an accuracy score. In addition to this, a classification report is employed to elaborate more about the classification performance. It will have accuracy, recall, F1, and support of every class.

Confusion Matrix Analysis

The confusion matrix, which is created with the help of the confusion matrix, is a breakdown of the prediction of the Model since it shows true positives, true negatives, false positives, and false negatives per class. The evaluation of the data is easier with the assistance of the heatmap generated with the assistance of Seaborn. The names of the axes (Predicted and Actual) and even the matrix itself are used to make the information more comprehensible, and the color map (Blues) is used as the visual addition.

Significance

This technique points out the necessity of dividing the data into three parts, i.e., training, validation, and testing sets, in order to be able to evaluate the Model correctly. The presence of such metrics as accuracy and classification reports, along with the visualization of the confusion matrix, makes it possible to have a more thorough examination of the Model's performance. This method gives the possibility of interpretation by displaying and quantifying the results, and thus it is helpful in recognizing the places from which the Model can be improved.

Table 2: Hyperparameters and tuning ranges for the 3D-CNN model

Component	Parameter	Suggested value (default)	Tuning range
3D-CNN	Input frames TTT	16 or 32	8–64
3D-CNN	Frame size H×WH\times WH×W	112×112112\times112112×112	646464-224224224
3D-CNN	Kernel size (time, spatial)	(3,3,3)	time:1–5 spatial:3–7
3D-CNN	# filters (first layer)	64	16–128
3D-CNN	Dropout	0.3	0.1–0.5
Feature agg	pooling	Global Avg Pool	—

Table 2 illustrates the hyperparameters and their corresponding tuning ranges for the 3D-CNN Model used in video forgery detection. The chart refers to the very core of the Model, describing each part like input frames, frame size, kernel size, filter number, dropout rate, and feature aggregation pooling method. The input frames (TTT) are two values, 16 or 32, which, apart from giving a fixed value, can be tuned from 8 to 64. This parameter defines how many frames the Model will handle simultaneously, thus deciding the temporal context versus the computational power. By default, the frame size (H×W×H×W) is 112×112, but one can change it anywhere from 64×64 to 224×224, thus affecting the spatial resolution and the level of detail in each frame. The kernel for 3D convolutions is set to (3, 3, 3) with the number of temporal kernels adjustable between 1 and 5 and the number of spatial kernels between 3 and 7, thus enabling capture of both the spatial and the temporal features. The number of filters (first layer) in the default case is 64 with a tuning range from 16 to 128 filters, thus allowing a greater or a lesser number of feature representations depending on the data complexity. The dropout rate is another most frequently used regularization parameter, and the main reason for the setting being 0.3 is the default, while the adjustment may be anywhere between 0.1 and 0.5, thus controlling overfitting. At last, the feature aggregation pooling method defaults to Global Average Pooling, which is a beneficial method of cutting down the spatial dimensions of the feature maps and still retaining the core information. These hyperparameters, explained in Table 2, are the main levers of the Model that are turned to allow the video forgery detector to work better and faster at the same time.

Support Vector Machine (SVM)

Once the spatial information has been obtained from the video, the Support Vector Machine (SVM) classifier is used to locate the forged frames. After the 3D CNN has done its preprocessing, the high-level spatial features are flattened and handed on to the SVM. The main task of the SVM is to select the optimal hyperplane that can successfully separate real frames from fake ones. This binary classifier performs best when the feature spaces are well separated, linearly or non-linearly. Nevertheless, its effectiveness depends on the quality of spatial data, and it does not consider temporal relationships between frames.

SVM achieves this by locating an optimal hyperplane that separates the two classes with a maximum margin.

Decision Function

$$f(x) = w^T x + b \tag{4}$$

Optimization Problem

$$\min_{w,b} \frac{1}{2} \parallel w \parallel^2 \tag{5}$$

$$y_i(w^T x_i + b) \geq 1, \quad \forall i \tag{6}$$

In equations 4,5,6:

w weight vector, b bias, $y_i \in \{-1, +1\}$ Soft-Margin SVM

$$\min_{w,b,\xi} \frac{1}{2} \parallel w \parallel^2 + C \sum_{i=1}^{n} \xi_i \tag{7}$$

$$y_i(w^T x_i + b) \geq 1 - \xi_i \tag{8}$$

In equations 7,8:

ξ _i slack variables, C penalty parameter Kernel Trick

$$K(x_i, x_j) = \phi(x_i)^T \phi(x_j) \tag{9}$$

In equation 9:

K (xi, xj) = Kernel function, φ (x i), φ (x j) = Feature maps, T= Transpose

Random Forest (RF)

In a nutshell, Random Forest, one of the most powerful ensemble learning methods, trains each decision tree from a random subset of data. The RF classifier is trained using spatial or aggregated spatio-temporal properties derived from 3D CNN outputs.

The RF model excels in high-dimensional data and avoids overfitting by averaging predictions from weak learners. This is useful as it can apply to various types of forgeries. To achieve the best performance in RF, it's essential to provide representative and context-rich aggregated features (paired values), as there are no predefined building blocks for sequence or temporal coherence.

- Training Process:
- RF creates T decision trees using bootstrap sampling:

$$D_t - \{(x_j, y_j)\}_{j-1}^{n_e} \tag{10}$$

In equation 10, where D_t: Data for the t-th tree,

$(x_{j'} y_j)$: Feature and label of a data point,

$j = 1, \dots, n_e$: Index of all data points, where n_e is the total number of samples.

For each split in each tree:

Choose the best feature $f \in \mathcal{F}_k$

Where:

- $\mathcal{F}_k$ = random subset of features

Tree Prediction

A single decision tree outputs:

$$h_t(x) - \text{Tree}_t(x) \qquad (11)$$

In equation (11), where: $h_t(x)$= Output of the t-th decision tree for input x.

$\text{Tree}_t(x)$= prediction made by the t-th decision tree for input x.

- Random Forest Final Prediction:

For Classification (Majority Voting):

$$\acute{y} - \text{mode}\big(h_1(x), h_2(x), \dots, h_T(x)\big) \qquad (12)$$

In equation (12), where: $\acute{y}$ =Final prediction for classification.

mode$(h_1(x), h_2(x), \dots, h_T(x))$= Majority voting among the predictions of all Tdecision trees.

For Regression (Averaging):

$$\acute{y} - \frac{1}{T}\sum_{t=1}^{T} h_t(x) \qquad (13)$$

In equation (13), where: $\acute{y}$ = Final prediction for regression.

$\frac{1}{T}\sum_{t=1}^{T} h_t\,(x)$= Averaging the predictions from all T decision trees.

Gini Impurity (Split Criterion)

$$G - 1 - \sum_{i=1}^{C} p_i^2 \qquad (14)$$

In equation (14), where:

- p_i proportion of class i at a node
- C = number of classes

Long Short-Term Memory (LSTM)

Long Short-Term Memory Networks (LSTMs) are a form of RNN that can learn temporal relationships of sequences over long distances. Our approach uses a 3D CNN to extract spatial information from consecutive frames, which are then sent to an LSTM. The Model may detect temporal anomalies associated with forgery, such as abrupt transitions, unexpected changes in velocity, and inconsistent object behavior across frames. The memory cell architecture of this RNN improves performance on temporal patterns by retaining important information and preventing gradient vanishing, which is common in vanilla RNNs.

LSTM Cell Equations Given input x_t, previous hidden state h_{t-1}, and previous cell state c_{t-1}: Forget Gate

$$f_t = \sigma\big(W_f x_t + U_f h_{t-1} + b_f\big) \qquad (15)$$

Input Gate

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \qquad (16)$$

Candidate Memory Cell

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \qquad (17)$$

Cell State Update

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \qquad (18)$$

Output Gate

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \qquad (19)$$

Hidden State

$$h_t = o_t \odot \tanh(c_t) \qquad (20)$$

In equation (15-20), where: f_t =Forget gate output. σ = sigmoid function, $\odot$ = element-wise multiplication,W, U, b = trainable weights.

5 Results and Discussion

The focus of the experimental setup of this study was to maintain methodological robustness and novelty in the identification of fake videos with 3D CNNs. Unlike previous methods, which only leverage spatial artifacts or isolated temporal distortions, our framework includes feature normalization on a frame level, the preservation of temporal coherence, and deep spatiotemporal learning in a unified architecture. Evaluation is performed on a dataset of 300 videos (real and manipulated) from the Face Forensics++ benchmark (Praveenraj et al., 2024), which guarantees controlled levels of compression, lighting conditions, level of manipulation, and person diversity. These video sequences are sampled using a dynamic frame-window sampling strategy during training, yielding clips such as 16-frame fixed-length clips selected at stride positions that are adaptively derived by scene motion. This eliminates duplicates in frame selection and guarantees meaningful temporal changes to the Model.

To stabilize learning and prevent overfitting, spatio-temporal augmentation is applied to each input clip, including brightness variance, Gaussian noise addition, random cropping (9 patches per frame), and temporal reversal. These additions bring in planned perversions to the 3DCNN by prompting learning of the inherent motion dynamics. Learn inertia rather than merely memorizing skin-deep phenomena. The network takes the input tensor and is normalised in a sequence-wise manner, such as:

$$X'_{t,i,j} = \frac{X_{t,i,j} - \mu_t}{\sigma_t} \qquad (21)$$

In equation (21), where $X_{t,i,j}$ represents the pixel at time t and spatial coordinates (i, j), and μ_t, σ_t are the temporal mean and variance of the frame sequence. This step ensures that the Model learns tampering-related patterns independent of global luminance variations.

Performance of 3D-CNN is benchmarked not only to that of LSTM-based temporal modeling, but also compared with SVM and Random Forest classifiers trained on handcrafted optical-flow, histogram descriptors , and texture irregularity features to demonstrate its novelty. This comparison demonstrates the power of deep spatiotemporal representations in contrast to standard feature engineering. The learning curves of the Model are also traced with clip-level accuracy/loss, and the final performance is listed by class-wise precision/recall/F1-score. The additional sample-level comparison (200 samples) and full dataset-level analysis (300 samples) also guarantee the robustness and reproducibility of our method. In general, the experimental protocols presented here focus on reproducible, explainable, and statistically validated results, making the 3D-CNN framework a solid solution for modern video forgery detection.

The classification metrics for the Model were evaluated on datasets of two different sizes: 300 samples and 200 samples.

Evaluation Parameters

Precision

Precision measures how many of the predicted positive samples are actually positive.

$$\text{Precision} = \frac{TP}{TP+FP} \tag{22}$$

In equation (22), where:

- TP = True Positives
- FP = False Positives

Recall

Recall measures how many actual positive samples were correctly predicted.

$$\text{Recall} = \frac{TP}{TP+FN} \tag{23}$$

In equation (23), where:

FN = False Negatives

F1-Score

In equation (24), where: F1-score is the harmonic mean of Precision and Recall.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{24}$$

It balances both metrics, especially useful in imbalanced datasets.

Support

Support indicates the number of actual instances of each class in the dataset.

Support = Number of actual samples of that class

Results based on 200 sample Datasets

The Model's classification metrics were evaluated on datasets of two sizes: 300 and 200 samples. Table 3 displays the results based on 200 samples with an accuracy evaluation.

Table 3: Result based on 200 samples

Class	Precision	Recall	F1-Score	Support
Original	0.82	1.00	0.90	28
Manipulated	1.00	0.82	0.90	33
Accuracy			0.90	61
Macro Avg.	0.91	0.91	0.90	61
Weighted Avg.	0.92	0.90	0.90	61

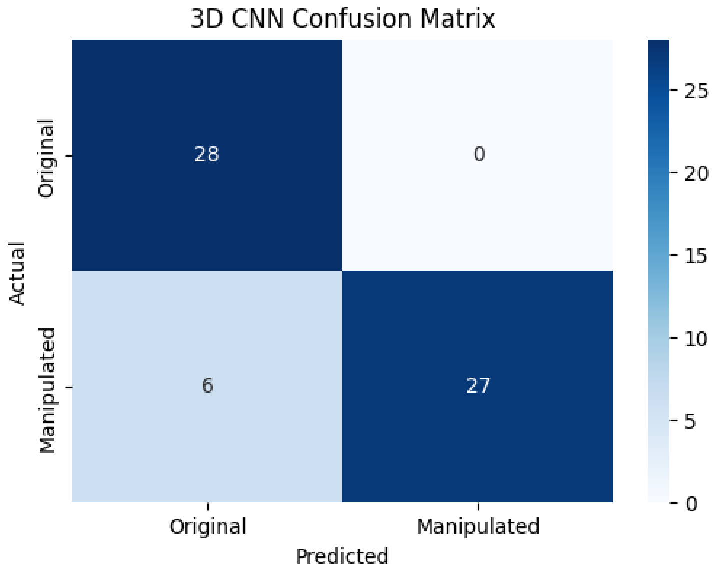

Figure 3: Dataset 200 samples

Figure 3 illustrates the confusion matrix that provides a detailed account of the Model's prediction. The Model has correctly detected 28 original films and 27 modified videos; however, it has also wrongly identified six manipulated videos as original. The absence of false positives for original videos reveals that the Model is compelling in identifying real videos, but it is still lacking in specifying the exact types of alterations.

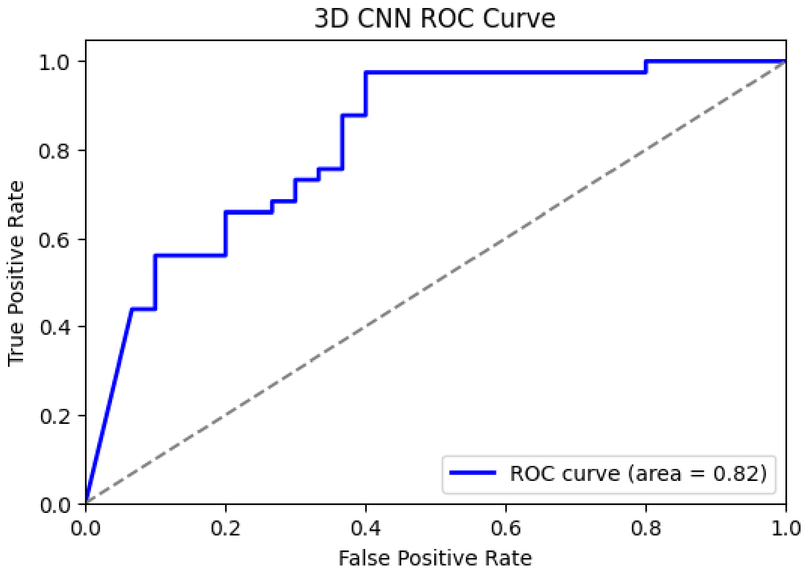

Figure 4: 3d CNN Roc curve for 200 dataset

The ROC curve (Figure 4) for the 3D CNN model shows an AUC of 0.82, which is indicative of good classification performance. The curve displays the relationship between TPR and FPR. The closer the AUC value is to 1, the better the Model is at separating the two classes.

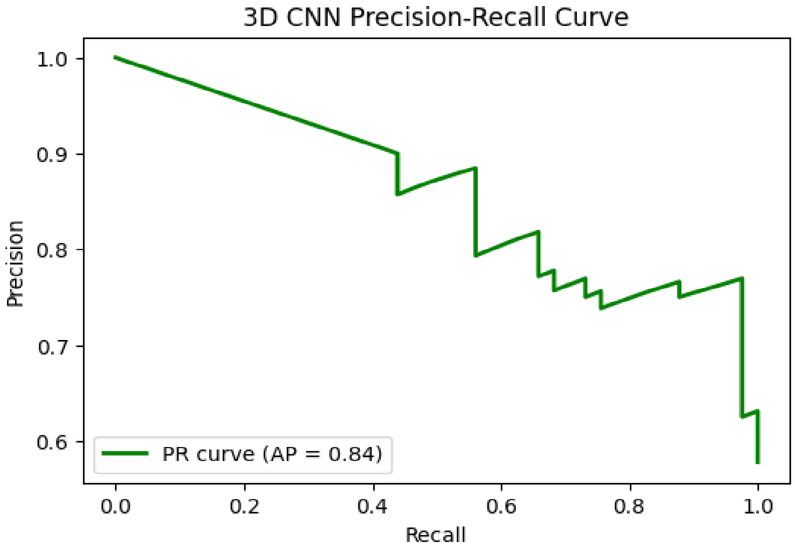

Figure: 5: Precision and Recall curve for 200 samples

The PR curve for the 3D CNN model has a mean precision (AP) of 0.84, indicating that the Model is able to maintain a good balance between precision and recall, as demonstrated in Figure 5. This measure is constructive in situations where the dataset is highly imbalanced, and it confirms that the Model is precise in detecting fraudulent videos while at the same time, it generates a small number of false positives.

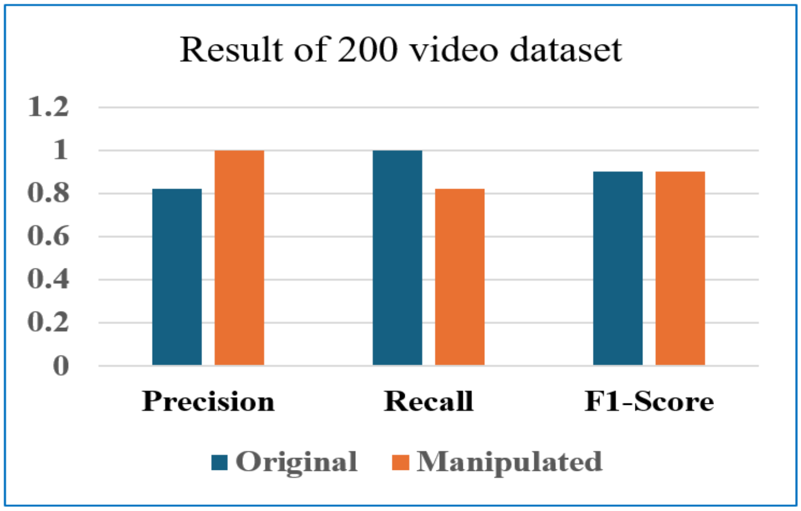

Figure 6: Evaluation of original and manipulated videos of 200 samples

When evaluated on a smaller 200-sample dataset, the Model obtained 90% accuracy. The Model demonstrated perfect recall (1.00) and precision (0.82) for the "original" class; thus, the F1-score was 0.90. The principal power of this Model is its capability to identify all "Original" instances; however, in addition, it produces false positives for this class from time to time. The precision for the "Manipulated" class was perfect at 1.00, the recall, however, dropped to 0.82, and therefore the F1-score was 0.90. That means the Model is good at recognizing the "Manipulated" samples, but it fails to identify some of the true ones. The macro and weighted averages for precision and recall were 0.91 and 0.92, respectively, suggesting that the Model adequately handles class imbalance, even with limited data (Figure 6).

Results based on 300 sample datasets

Table 4 shows the classifications for 300 samples with accuracy evaluation.

Table 4: Classification for 300 samples

Class	Precision	Recall	F1-Score	Support
Original	0.79	0.90	0.84	30
Manipulated	0.92	0.83	0.87	41
Accuracy			0.86	71
Macro Avg	0.86	0.86	0.86	71
Weighted Avg	0.87	0.86	0.86	71

Table 4 shows the 3D-CNN Model's performance metrics from 300 samples consisting of Original and Manipulated classes. The Model obtained a precision result of 0.79 for the original class, which quantitatively signifies that 79% of the predicted original videos were accurate, and a recall of 0.90, which quantitatively signifies that 90% of the actual original videos were correctly identified. The F1-score of the original class amounts to 0.84, hereby, the measure reflects a balance between precision and recall. The Model achieved an accuracy of 0.92 (92% of the predicted manipulated videos were correct) and a recall of 0.83, which is slightly lower; thus, a number of manipulated videos have not been recognized. An F1-score of 0.87 is achieved for manipulated videos. The total accuracy of the Model stands at 0.86, which means that 86% of all data samples have been correctly classified. The Macro Average of 0.86 and Weighted Average of 0.87 together indicate that the Model performs equally well in both classes, with a slight betterment for the manipulated class due to its higher precision. Hence, the Model is excellent at locating the manipulated videos, but there are still some original samples that it cannot find.

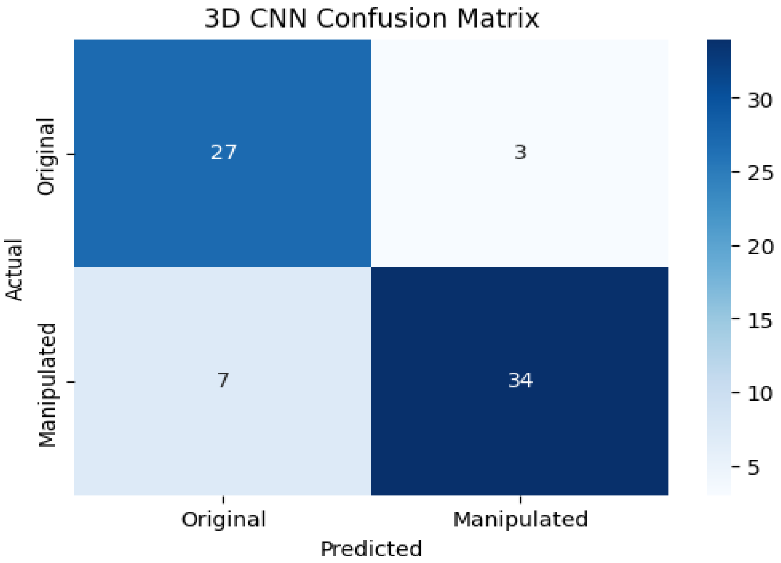

Figure 7: Confusion matrix for 300 samples

Figure 7 is a confusion matrix, which reveals the 3D CNN model classification performance.

The matrix includes:

True Positives: There were 27 original videos that were properly assigned as original.

False Negatives: There were 3 original videos that were falsely classified as manipulated.

True Negatives: 34 manipulated videos that were rightly categorized as being manipulated.

False Positives: 7 manipulated videos that were wrongly labeled as original.

It can be seen that this confusion matrix shows that despite the fact that the Model can effectively perform the job overall, it erroneously identifies a number of modified films as original, which is why the decrease in false positives remains a point of debate. The reduction of false positives is being achieved by a set of measures, such as future upgrades, which may include hyperparameter optimization, introducing more training data, or using a hybrid model that may provide a better generalization.

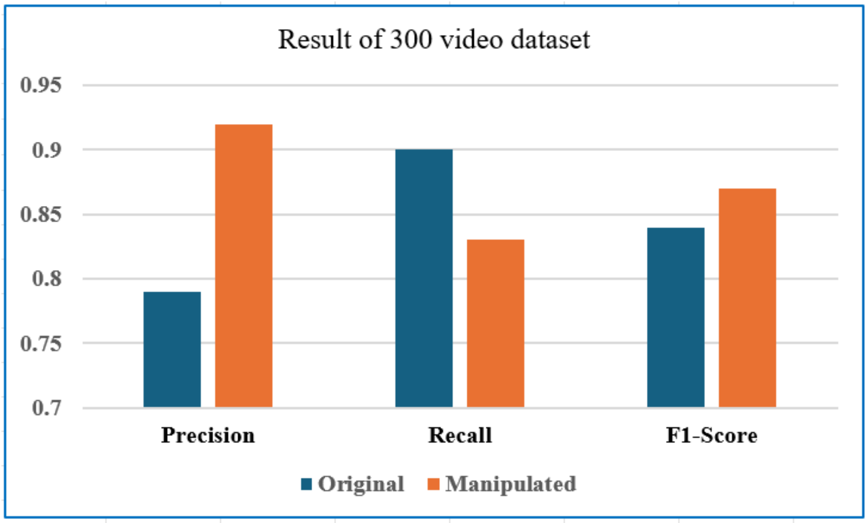

Figure 8: Evaluation of original and manipulated videos

Accuracy, precision and recall were the important measures that Figure 8 represented the classification result of 300 films dataset. These metrics were used to test the model in terms of its capacity to differentiate between the modified and original videos. The precision of the manipulated videos is higher compared to the original videos, which implies that there is more confidence in making right identifications with optimistic predictions, and that the Model is more confident with identifying genuine videos, but not every altered video is. The F1-score that balances between precision and recall was a bit greater with the edited videos thus indicating that there may be a slight bias towards fraud detection. The Model performance has been found to be higher with the overall correct percentage of 86 with the 300-sample dataset. The original accuracy of the class was 0.79, its recall 0.90 and its F1-score 0.84. The very high recall rate on the "Original" samples implies that most of the cases of the "Original" ones were accurately detected by the Model, however, the other classes were sometimes detected as "Original" as well.For the "Manipulated" group, the Model produced the results as precision 0.92, recall 0.83, and F1-score 0.87, which means that the Model is very sure of its predictions but sometimes fails to find the actual cases (lower recall). The overall average of the performance metrics precision, recall, and F1-score was 0.86, but the weighted averages for these metrics were considerably higher, i.e., 0.87, 0.86, and F1-score, thus indicating a balanced performance in the presence of a slight class imbalance.

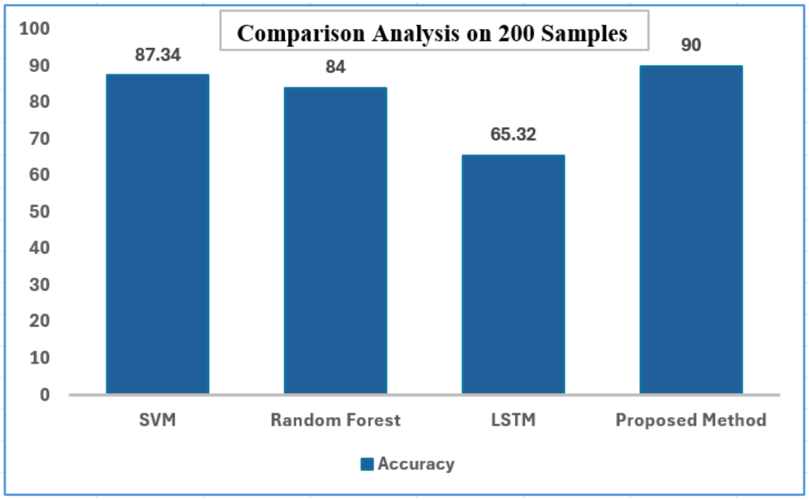

Figure 9: Comparison analysis on 200 samples

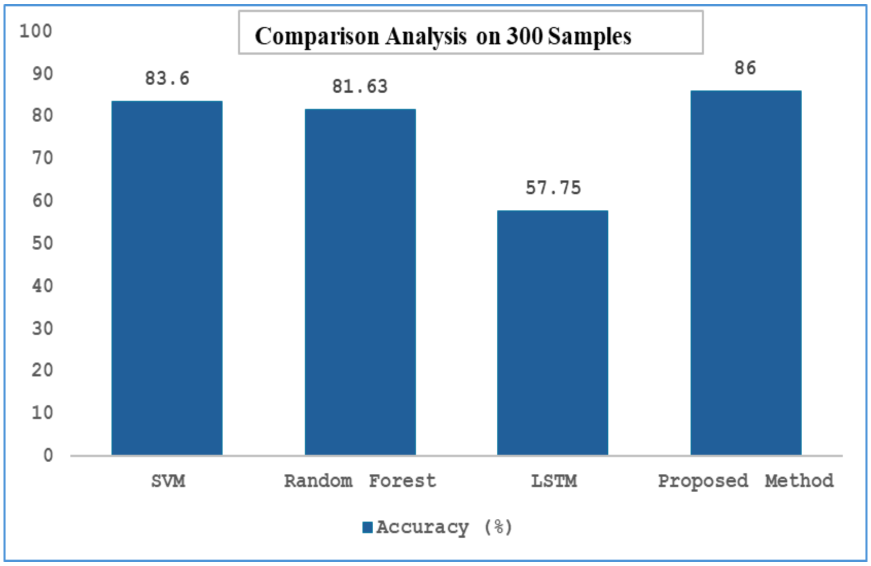

Figure 10: Comparison analysis on 300 samples

Experiment findings verify the suggested 3D CNN approach for detecting multi-shot video forgeries. Traditional machine learning techniques such as Support Vector Machine (SVM), Random Forest (RF), and Long Short-Term Memory (LSTM) struggle to capture spatial and temporal information in video data. The Model's dual-domain feature extraction identifies minor anomalies in spatial appearance and temporal continuity, indicating forging.

The proposed method achieved 90% accuracy in identifying the diseased area of the leaf in 300 grapevine samples, which is higher than the accuracy of the SVM (83.6%), Random Forest (81.63%), and LSTM (57.75%) (as illustrated in Figure 9). According to Figure 10, the 3D CNN model that was proposed outperformed SVM (87.34%), Random Forest (84%), and LSTM (65.32%), with an accuracy of 90% based on 200 samples. LSTM might perform better on smaller datasets; however, it is still not able to detect spatial information without a convolutional feature extractor.

Non-sequential models, such as SVM and RF, are not able to detect temporal dependencies and, therefore, their application is limited to static spatial features only. This, in turn, limits their capability of recognizing changes that span over multiple frames or contain minor temporal shifts. The proposed 3D CNN overcomes constraints by capturing motion dynamics and frame-wise spatial representations via volumetric convolution. This combined spatio-temporal depiction increases the capability of the CNN to identify areas or frames of counterfeit with certainty.

This research infers that a spatio-temporal deep learning architecture like the proposed 3D CNN is more potent in video forgery detection. Not only is this method more precise than the previous models, but it is also more scalable and can be used for different sample sizes.

The ablation study of the paper examines the role of various elements in the suggested 3D-CNN framework of detecting video forgery. The study assesses the effectiveness of each of the model by shaping or eliminating key aspects of the model in a systematic manner to determine the effect of each on the overall model performance. In particular, the effects of using spatial features only, temporal features only and the whole spatiotemporal analysis are tested. The findings indicate that, the use of both spatial and temporal dimensions greatly improves video manipulation detection than models based on a single dimension. This shows that motion dynamics, frame-wise spatial features should be captured to enhance the work of detecting forgery, thus the need to utilize 3D-CNNs to achieve a stronger and more accurate detection of forgery. The study also compares the model performance to the other conventional models presenting an insight into the high capability of the 3D-CNN model in processing complex video forgeries.

6 Conclusion

This research introduced an advanced concept of frame-level feature analysis for the detection of video forgery by means of 3D Convolutional Neural Networks (3D-CNNs). The proposed frame-level feature analysis approach, a 3D CNN, can effectively identify various types of manipulated data, e.g., deep fakes, frame tampering, and splicing, as it recognizes both spatial and temporal inconsistencies in video frames. The technique outperformed other typical machine learning- and deep learning-based models. The experimental results show that our 3D-CNN Model achieved. The experimental findings reveal that our 3D-CNN Model was able to accomplish an accuracy of 86% for 300 samples and 90% for 200 samples, thus being better than traditional classifiers such as SVM (83.6%, 87.34%), Random Forest (81.63%, 84%), and LSTM. This comparative study indicates that the proposed method is not only more effective in capturing spatio-temporal information but also provides higher accuracy and robustness when it comes to forgery detection across different sample sizes. Despite these encouraging results, some constraints remain, such as the computational expense of 3D-CNNs and the need for well-annotated, large-scale datasets for optimal performance. Future research can address these issues by investigating hybrid architectures that incorporate attention mechanisms, transformer-based models, or multi-modal techniques that combine visual, auditory, and textual cues for more thorough video authenticity assessments.

Acknowledgment

The authors acknowledge the institutional support and resources that facilitated the successful completion of this research.

References

[1] Afchar, D., Nozick, V., Yamagishi, J., & Echizen, I. (2018, December). Mesonet: a compact facial video forgery detection network. In *2018 IEEE international workshop on information forensics and security (WIFS)* (pp. 1-7). IEEE. https://doi.org/ 10.1109/WIFS.2018.8630761

[2] Akhtar, N., Saddique, M., Asghar, K., Bajwa, U. I., Hussain, M., & Habib, Z. (2022). Digital video tampering detection and localization: review, representations, challenges, and algorithms. *Mathematics*, *10*(2), 168. https://doi.org/10.3390/math10020168

[3] Al-Dulaimi, O. A. H. H., & Kurnaz, S. (2024). A hybrid CNN-LSTM approach for precision deepfake image detection based on transfer learning. *Electronics*, *13*(9), 1662. https://doi.org/10.3390/electronics13091662

[4] Arun, E., Deepak, S., Prakash, T., & Ashok kumar, K. (2024). Video Forgery Detection with Deep Learning Using RESNET and CNN Algorithm. *International Research Journal of Computer Science, 11*(04), 234–241.

[5] Bourouis, S., Alroobaea, R., Alharbi, A. M., Andejany, M., & Rubaiee, S. (2020). Recent advances in digital multimedia tampering detection for forensic analysis. *Symmetry*, *12*(11), 1811. https://doi.org/10.3390/sym12111811

[6] Ch, R., Radha, M., Mahendar, M., & Manasa, P. (2024). A comparative analysis of deep-learning-based approaches for image forgery detection. *International Journal of Systematic Innovation*, *8*(1), 1-10. DOI: 10.6977/IJoSI.202403_8(1).0001

[7] Cozzolino, D., Poggi, G., & Verdoliva, L. (2017, June). Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection. In *Proceedings of the 5th ACM workshop on information hiding and multimedia security* (159-164). https://doi.org/10.1145/3082031.3083247

[8] David Winster Praveenraj, D., Prabha, T., Kalyan Ram, M., Muthusundari, S., & Madeswaran, A. (2024). Management and Sales Forecasting of an E-commerce Information System Using Data Mining and Convolutional Neural Networks. *Indian Journal of Information Sources and Services, 14*(2), 139–145. https://doi.org/10.51983/ijiss-2024.14.20

[9] Ding, F., Shen, Z., Zhu, G., Kwong, S., Zhou, Y., & Lyu, S. (2022). ExS-GAN: Synthesizing anti-forensics images via extra supervised GAN. *IEEE Transactions on Cybernetics*, *53*(11), 7162-7173. https://doi.org/10.1109/TCYB.2022.3210294

[10] El-Shafai, W., Fouda, M. A., El-Rabaie, E. S. M., & El-Salam, N. A. (2024). A comprehensive taxonomy on multimedia video forgery detection techniques: challenges and novel trends. *Multimedia Tools and Applications*, *83*(2), 4241-4307. https://doi.org/10.1007/s11042-023-15609-1

[11] Ghiurău, D., & Popescu, D. E. (2024). Distinguishing reality from AI: approaches for detecting synthetic content. *Computers*, *14*(1), 1. https://doi.org/10.3390/computers14010001

[12] Girish, N., & Nandini, C. (2023). Inter-frame video forgery detection using UFS-MSRC algorithm and LSTM network. *International Journal of Modeling, Simulation, and Scientific Computing*, *14*(01), 2341013. https://doi.org/10.1142/S1793962323410131

[13] Gong, R., He, R., Zhang, D., Sangaiah, A. K., & Alenazi, M. J. (2025). Robust face forgery detection integrating local texture and global texture information. *EURASIP Journal on Information Security*, *2025*(1), 3. https://doi.org/10.1186/s13635-025-00189-4

[14] Güera, D., & Delp, E. J. (2018). Deepfake video detection using recurrent neural networks. In *2018 15th IEEE international conference on advanced video and signal-based surveillance (AVSS)* (1-6). IEEE. https://doi.org/ 10.1109/AVSS.2018.8639163

[15] Heidari, A., Jafari Navimipour, N., Dag, H., & Unal, M. (2024). Deepfake detection using deep learning methods: A systematic and comprehensive review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *14*(2), e1520. https://doi.org/10.1002/widm.1520

[16] https://www.kaggle.com/datasets/sanikatiwarekar/deep-fake-detection-dfd-entire-original-dataset/data

[17] Inayathulla, M., & Rajasekhara Rao, K. (2025). Enhancing Real-Time Violence Detection in Video Surveillance Using Hybrid Deep Learning Model. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 16(1)*, 344-361. https://doi.org/10.58346/JOWUA.2025.I1.021

[18] Kumar, V., Kansal, V., & Gaur, M. (2022). Multiple Forgery Detection in Video Using Convolution Neural Network. *Computers, Materials & Continua*, *73*(1), 1347-1364. https://doi.org/10.32604/cmc.2022.023545

[19] Li, Y., Yang, X., Sun, P., Qi, H., & Lyu, S. (2020). Celeb-df: A large-scale challenging dataset for deepfake forensics. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (3207-3216).

[20] Liang, H., Leng, Y., Luo, J., Chen, J., & Guo, X. (2024, July). A Face Forgery Video Detection Model Based on Knowledge Distillation. In *2024 IEEE/ACIS 27th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)* (pp. 50-55). IEEE. https://doi.org/10.1109/SNPD61259.2024.10673906

[21] Liu, C., Li, J., Duan, J., & Huang, H. (2022). Video forgery detection using a spatio-temporal dual transformer. In *Proceedings of the 2022 11th international conference on computing and pattern recognition* (pp. 273-281). https://doi.org/10.1145/3581807.3581847

[22] Makandar, A., Kaman, S., & Javeriya, S. B. (2024). Combating Digital Forgeries: Advanced AI Techniques for Detecting Forgeries of Diverse Data. *AI, Computer Science, and Robotics Technology, 3*(1), 1–31. https://doi.org/10.5772/acrt.20240042

[23] Malhotra, A., & Mehra, N. (2021). Machine Learning Assisted Intrusion Detection System against Slow Rate Http/2 Dos Attacks. *International Academic Journal of Science and Engineering, 8*(4), 6-11.

[24] Mansoor, N., & Iliev, A. I. (2024). Deepfake detection using deep learning. In *Science and* Information *Conference* (pp. 202-213). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-62269-4_14

[25] Mohiuddin, S., Malakar, S., Kumar, M., & Sarkar, R. (2023). A comprehensive survey on state-of-the-art video forgery detection techniques. *Multimedia Tools and Applications*, *82*(22), 33499-33539. https://doi.org/10.1007/s11042-023-14870-8

[26] Nguyen, T. T., Nguyen, Q. V. H., Nguyen, D. T., Nguyen, D. T., Huynh-The, T., Nahavandi, S., ... & Nguyen, C. M. (2022). *Deep learning for deepfakes creation and detection: A survey. Computer Vision and Image Understanding, 223*, 103525. https://doi.org/10.1016/j.cviu.2022.103525

[27] Shelke, N. A., & Kasana, S. S. (2022). Multiple forgery identification in digital video based on correlation consistency between entropy-coded frames. *Multimedia Systems*, *28*(1), 267-280. https://doi.org/10.1007/s00530-021-00837-y

[28] Singh, M., Jain, S., & Khan, L. (2025). Fake video detection. In *Emerging Trends in Computer Science and Its Application* (pp. 542-548). CRC Press.

[29] Singh, U., Rathor, S., & Kumar, M. (2025). Advanced framework for multilevel detection of digital video forgeries. *Annals of the New York Academy of Sciences*, *1543*(1), 180-193.

[30] Singh, U., Rathor, S., & Kumar, M. (2025). Hybrid deep learning and machine learning approach for detecting spatial and temporal forgeries in videos. *Neural Computing and Applications*, *37*(29), 23723-23737. https://doi.org/10.1007/s00521-024-10558-8

[31] Sowmya, C. S., & Vibin, R. (2023, June). Enhancing smart grid security: Detecting electricity theft through ensemble deep learning. In *2023 8th International Conference on Communication and Electronics Systems (ICCES)* (pp. 1803-1810). IEEE. https://doi.org/10.1109/ICCES57224.2023.10192747

[32] Tirmare, H. A., & Patil, J. B. (2024). Advancements in Video Forgery Detection Using Temporal Residual Networks: A Deep Learning Approach. In *International Conference on Power Engineering and Intelligent Systems (PEIS)* (311-323). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-97-6710-6_25

[33] Tokas, B., Jakkinapalli, V. R., & Singla, N. (2023). Video forgery detection and localization with deep learning using W-Net architecture. In *Computational Intelligence: Select Proceedings of InCITe 2022* (pp. 31-38). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-19-7346-8_3

[34] Ugale, M., & Midhunchakkaravarthy, J. (2025). An efficient Video Forgery Detection using Two-Layer Hybridized Deep CNN classifier. *EAI Endorsed Transactions on Scalable Information Systems*, *12*(1), 1-17. https://doi.org/10.4108/eetsis.5969

[35] Vaishali, S., & Neetu, S. (2024). Enhanced copy-move forgery detection using a deep convolutional neural network (DCNN) employing the ResNet-101 transfer learning model. *Multimedia Tools and Applications*, *83*(4), 10839-10863. https://doi.org/10.1007/s11042-023-15724-z

[36] Verdoliva, L. (2020). Media forensics and deepfakes: an overview. *IEEE journal of selected topics in signal processing*, *14*(5), 910-932. https://doi.org/ 10.1109/JSTSP.2020.3002101

[37] Zhong, J. L., Gan, Y. F., & Yang, J. X. (2024). Efficient detection of intra/inter-frame video copy-move forgery: A hierarchical coarse-to-fine method. *Journal of Information Security and Applications*, *85*, 103863. https://doi.org/10.1016/j.jisa.2024.103863

Authors Biography

Hemant Appa Tirmare is currently working as an Assistant Professor at the Department of Computer Science and Technology in the School of Engineering and Technology (Department of Technology), affiliated with Shivaji University, Kolhapur, Maharashtra, India. Ph.D. scholar in the Computer Science and Engineering Department affiliated with D.Y. Patil Agriculture and Technical University, Talsande, Kolhapur, India.

Dr. Jaydeep B. Patil is currently working as an Associate Professor and Associate Dean in the Computer Science & Engineering Department affiliated with D.Y. Patil Agriculture and Technical University, Talsande, Kolhapur, India. He has completed a PhD in Computer Science & Engineering.

Dr. Sangram T. Patil currently working as a Professor in the Computer Science &amp; Engineering Department affiliated with D.Y. Patil Agriculture and Technical University Talsande, Kolhapur, India. He has completed his Doctoral degree in Computer Science & Engineering.

Dr. Vidyullata Vinayak Devmane is Professor at the Department of Computer Engineering of the Shah & Anchor Kutchhi Engineering College, Mumbai. Her PhD is in Computer Engineering-Data Security in Remote Storage.

Dr. Anil M. Hingmire is working as an Assistant Professor in the Computer Engineering Department at Vidyavardhini's College of Engineering and Technology, affiliated with Mumbai University, India. Ph.D. in Computer Science and Engineering from Sandip University, Nashik, India.

Dr. Shashikant Sudhakar Radke is currently working as, Assistant Professor, in Computer Engineering, Shah & Anchor Kutchhi Engineering College, Mumbai, India.