Journal of Computer Science

Research Article

Enhancing Video Tampering Detection Using Dynamic Temporal LSTM With Adaptive CNN

Gurpreet Kour Khalsa, Rakesh Ahuja and Rattan Deep Aneja

Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab, India

Article history
Received: 14-07-2025
Revised: 09-10-2025
Accepted: 08-12-2025

Corresponding Author:
Gurpreet Kour Khalsa
Chitkara University Institute of Engineering and Technology, Chitkara University Punjab, India
Email: er.kaurgurpreet123@gmail.com

Abstract: In the domain of information technology, video tampering detection has become hyper critical principally with the increase in deep fake as everyone is having affordable access to the internet. The long established methods lack in detecting the manipulated content specifically for temporal disordered and variant frames. In order to overcome such issues, the suggested innovative method encompasses Dynamic Temporal Warping (DTW) within the LSTM framework to efficiently focus on these temporal misalignments, which are usually experienced in real-world scenarios. Hence, an adaptive CNN component is introduced to dynamically adjust for frame rate variations, significantly reducing misclassification rates. Moreover, the proposed method is implemented in Python and it outperforms existing approaches, achieving 96.83 accuracy, 96.9 precision, 96.9 recall, 97.3 F1-score and 98% sensitivity, while also maintaining a lower false positive rate of 2%, making it highly effective for real-time tampering detection in deep fake applications.

Keywords: Video Forensics, Safety, Deep Learning, Deep Fake, Temporal LSTM, Resources, Video Security

Introduction

The development of many different advanced image processing techniques has increased digital forgeries in daily life. Digital video is often defined as a series of moving images with less spatial and temporal redundancy. A digital video consists of a high-speed series of still frames (Fadl et al., 2021). In present scenario majority of transmission and exchange of visual information in daily life is made possible by the sophisticated, affordable digital video cameras found in many gadgets, mobile phones and video-sharing websites including Facebook, YouTube and Dailymotion. In legal terms visual data is considered as a strong evidence to substantiate or confirm a witness's testimony (Nguyen et al., 2020). It is impossible to ignore the authenticity of videos in the era of advanced and accessible video editing tools. Information modification is now simple thanks to sophisticated editing tools (Osorio- Arteaga and Giraldo, 2024). With good or evil intentions, videos can be altered by adding or removing objects or events. Traditional face tampering is time-consuming and requires expert video editing tools and knowledge. Image synthesis is now a breakthrough due to developments in deep learning and computer technology (Saini and Ahuja, 2024) Face alteration movies like Deepfake have gained wide popularity on social media and video-sharing sites proving that "seeing isn't always believing." GAN (Fadl et al., 2021) and Auto-Encoders are being used in Deepfake to create and distribute high-quality video-altering content (Chittapur et al., 2020). The authenticity of media has been the subject of several intriguing studies. Nevertheless, the massive and complex volume of multimedia that has to be analyzed makes it difficult to develop an effective multimedia tampering detection system (Tariq et al., 2020). Several techniques have been developed to apply machine learning to the identification of video forgeries. The majority of the first work in picture forgery detection was carried out with a Support Vector Machine (SVM). Later, the issue of video counterfeiting for frame duplication detection was also tackled using neural network based methods (Joshi and Jain, 2020). Convolution neural networks (CNNs), in particular, are deep learning techniques that have seen great success recently because of their potent capacity for large-scale video categorization through automatic feature learning (Zheng et al., 2021; Fadl et al., 2020). Also, deep learning techniques have been quite successful in many different areas. Generative Adversarial Networks (GANs) have been utilized to alter source footage in many ways, including as imitating human facial expressions, manipulating the weather and implementing face-

SCIENCE Publications

swapping. Because so many techniques have been created and since tamper detection is a continually explored field, its performance is far from trustworthy. Existing investigation on this topic lacks comprehensive and universal solutions, allowing space for additional contributions (Saddique et al., 2020). Most recent studies have focused on static tampering detection, however dynamic tampering detection has gained less attention due to its complexity and computing expense (Munawar and Noreen, 2021). It turns out that video forensics makes this task more complex. Video tampering detection faces new challenges due to complex dynamic scene analysis, computational costs, occlusions, changes in perspective, multiple scales, misalignments in the temporal domain, inadequate adaptation to dynamic frame rate variability and temporal feature extraction. These difficulties highlight the need to explore this field of investigation (Rossler et al., 2019; Balasubramanian et al., 2022). Detecting malicious alteration in digital media is important as it becomes more difficult to distinguish between modified and authentic images due to advanced forging techniques (Velliangira et al., 2020). In the realm of public security, an accurate video tampering detection system is becoming more and more essential since advanced forgeries are difficult to identify (Nguyen et al., 2020).

Therefore, this study will develop an effective deep learning-based approach for detecting video tampering, specifically focusing on deep fakes. A deep neural network technique is given for classifying fake videos.

The following methodological and experimental contributions have been achieved by this paper:

- To overcome the temporal misalignments in video tampering detection, Dynamic Temporal LSTM is introduced, this combines DTW within LSTM to accurately align frame sequences despite varying frame rates thereby, reducing false positives and enhancing the accuracy of video tampering detection systems
- To mitigate misclassification errors caused by frame rate variations in video classification Adaptive CNN is introduced, in which an adaptive thresholding algorithm is incorporated within the CNN, allowing dynamic adjustment of thresholds based on observed frame rate variations, thus enhancing the accuracy of distinguishing between original and fake videos

Related Work

The author here proposed a deep learning strategy using transfer learning and customized CNN layers to detect tampered real-time videos with both static and moving backgrounds. The suggested method shows over 99.9% accuracy and effectiveness compared to existing methods, providing trustworthy results with low computing cost and strong detection performance (Koshi and Shyry, 2025). The paper also mentions that forgery detection in videos can be done using conventional methods such as sequential and patch analysis, hierarchical methods, etc. The existing methods were surpassed in the matter of accuracy and delivered reliable results involving low computational cost. Thus, the findings exhibited better results as compared to the state of art methods (Koshi and Shyry, 2025). In present scenario the tampering detection in videos due to machine learning has increased tremendously which ultimately results in dissemination of the false news, harassing an individual and other such delinquent activity (Pandey et al., 2023). The two different methods which have been proposed for detection are passive method and active method. Passive method depends on irregularities in sensor, splicing and compression in frames.

The active methods take into consideration the digital watermarks, digital signatures and fingerprints to assist in video tampering detection. Machine learning based systems have shown promising outcomes in identifying video manipulation by training models on datasets of tampered and real recordings. Challenges that still need to be addressed include detecting tampering in live video broadcasts and identifying deep fake movies (Pandey et al., 2023). Xing et al. (2022) proposed a composite network model using the Siamese network and the bidirectional long short-term memory network auto-encoder to detect tampered frames in videos. The Siamese network calculates the inter-frame distance by calculating the depth features of the frames extracted by VGG-16. The frames depth features are then fed as an input into the BiLSTM Auto-Encoder for frame sequence anomaly detection and localization. The model combines deep learning techniques with frame discrimination and localization methods to improve accuracy. The model is experimented on two different datasets and achieves good results, validating its generalization performance. The results show that the proposed deep learning model achieves higher precision (93.7%) in detecting tamper points compared to classical methods (Xing et al., 2022). Harika et al. *(*2023) proposed method which classifies the video as authentic or tampered by performing patch analysis and error level analysis using video frames and using a deep learning classifier Resnet50. Patch model includes preprocessing the video frames and then separating the patches whereas in error level analysis, the video frames undergo compression level at 90% and then difference is applied. Luan and Damian (2023) depicted the usage of deep learning techniques in tampering detection and its implication in higher education usage. This study focused on resolving the issue of forged images circulated on the social media and bring into notice the need of experts in terms of differentiating these images. Halak et al. (2022) proposes the error level analysis method and uses image metadata to contrast between the original and counterfeited images. For deep learning, a basic neural network comprising two convolutional layers, two dense layers, a max pooling layer, a dropout layer, and one output layer is employed. Girish and Nandini (2023) here proposed the novel video forgery detection that incorporates the UFS-MSRC algorithm & LSTM network for detection of duplication in the regions of the video. The features are being extracted using GoogleNet model and in order to retrieve the background information from the dynamic scene spatiotemporal averaging is being used.

The Unsupervised Feature Selection with Multi-Subspace Randomization and Collaboration (UFS-MSRC) approach reduces training time and increases detection accuracy by choosing discriminative feature vectors. The LSTM network shows its implication in detection of the forgery in different video sequences. The experimental results exhibited that UFS-MSRC with LSTM model achieves high accuracy, with 98.13 and 97.38% accuracy on the SULFA and Sondos datasets. Thereby the results outperform existing models in video forgery detection. Kumar et al. (2022) proposed a method making use of parallel deep neural networks and analytical computations. For the purpose of detecting inter frame video forgery the correlation coefficients are calculated from deep features. The methodology here is divided into two different phases firstly video forgery detection and second one classification. In the detection phase it is determined whether the given video frame is original or manipulated and if the frame is considered not to be real then it is followed by the classification step. The proposed method has been tested upon two different datasets where it achieved accuracy of 91 in VIFD dataset and 90% on TDTV dataset. At the same time the method was being tested for insertion and deletion detection on both the datasets with achieving 8 and 86% accuracy (Kumar et al., 2022). BR et al. (2023) proposed a deep fake detection system which integrates LSTM and Res Net 50 architectures. The major focus of this particular approach is based on the sequential and image based data where in the finest features of both the architectures are being retrieved out of it. In order to perform the comparative analysis, the authors make use of diverse datasets such as Celeb –DF and Face Forensics ++, thereby evaluating the accuracy of the proposed model in terms of detecting deep fakes. The study presented a web frame implementation based on python. Mira (2023) proposed new techniques for deep fake video identification using LSTM and CNN. Moreover, the study also incorporated CNN, XG Boost and the YOLO face detector for deep fake detection and suggests more research into state of art detection techniques. The main idea behind the study was to demarcate between facial video frames. The paper highlights the importance of deep fake identification for protecting data and preventing the spread of misleading information. Irrespective of the tremendous attainment in the field of deep fake detection, there still exists some research gaps that hamper the solidity, flexibility and clarity of the existing systems. The foremost issue is that the existing models specifically focuses on spatial domain analysis like unpredictability in pixels, malfunctions in texture whereas the temporal misalignments in the video frame sequence which are generally caused by variations in the frame and time distortion remain unexamined and which generally lead to higher false positive rates and erroneous tampering detection cues. Furthermore, the variation in frame rate poses great challenge in real world scenarios. The actual recordings get easily affected by the camera jitter, blur and compression, it means there is no uniform frame rate. In contrary to it the general assumption takes by the CNN based classifiers is that there is frame rate which are completely synchronized. Thus, when applied to the actual dynamic video content the accuracy gradually decreases. Finally, the detection capability of the existing systems is limited by the deficit of unified spatial-temporal framework. The reason behind it is the independent treatment of the spatial and temporal features thereby resulting in a Previous study often treat spatial and temporal features independently, resulting in an incomplete knowledge of inter-frame dependencies, which is essential for spotting minute changes in different video clips.

Methods

In digital multimedia forensics, video tampering detection is considered to be the most indispensable domain that focus on ensuring the authentication and originality of video frames in a scenario where it is very common to manipulate videos. The extensive utilization of forged videos with the intention of malicious activities such as spreading one’s information without their consent, deforming the proof, culminating fabricating tools. That being the case a distinctive approach of Dynamic temporal LSTM with Adaptive CNN, is thrived to address the problem with state of art methods especially focusing on deep fakes (Deo et al., 2023). In the beginning the video is turned into image frames which then undergoes preprocessing to eradicate noise and magnify the contrast. In the second phase the features are extracted making use of auto encoders which makes changes in the facial features. The most important feature for making the tampering detection definitive is discernment of irregularities among the video frames. In spite of this there are variables with the misalignments which pose a serious challenge in the smooth process of tampering detection.

The disordered frames in the temporal domain like variations in the frames or error relating to time sequence vague tampering cues and thereby generate false alarms. The current state of art techniques focusses on spatial inconsistencies including visual remnant or pixel level deviation throughout the frames. These methods commonly ignore the temporal characteristic thus lacking focus on the misalignments in frame timing which in turn gives inaccurate results for detecting fake content disseminated on the social media platforms. The reason behind this is the continuous frame rate variations and irregularities in the frames which remain undetermined. To deal with the provocation of temporal misalignments in video, here Dynamic Temporal LSTM is introduced. In this method, the existing irregularities between the frames extracted and the original video references are detected over the time domain by making use of LSTM which takes into consideration the frame sequences. This is attained by integrating Dynamic Temporal Warping (DTW) within LSTM, which increases detection by lining up the frames sequences irrespective of the frames at a varying rate and with timing errors, guaranteeing precise comparison and identification of tampering cues over time. By amalgamating DTW, the LSTM guarantees the subtle tampering cues capturing and thus decreasing false positives and increasing the accuracy of tamper detection in digital video forensics. Furthermore, it is very difficult to tell the difference between real and fake videos due to frame rate differences between real movies. Additionally, the variations in frame rates among authentic films provide a considerable issue in significantly differentiating between real and fake video frames. The original video frames do have some fluctuations related to the settings of camera, recording condition or polishing the content whereas in the forged videos there are constant abetment in the frame rates. This issue makes it quite difficult to differentiate between the original tampering being done on the frame or some moderate fluctuation. As the current techniques generally assume the variations in the frame to be uniform or same frame rate throughout the clips thereby failing to interpret natural variations in original videos in original videos which in turn leads to misclassifications. In order to address the issue of frame rate variation and increase the accuracy of classification, an adaptive CNN approach is given. By dynamically adjusting the threshold values in response to the observed frame rate variations an adaptive thresholding algorithm is integrated into the CNN to further improve its capabilities. Therefore, by incorporating this hybrid approach, algorithm can differentiate between original and tampered videos with much greater accuracy and thus reducing misclassification errors due to frame rate variability. The proposed model's overall flow diagram is illustrated in the Fig. 1.

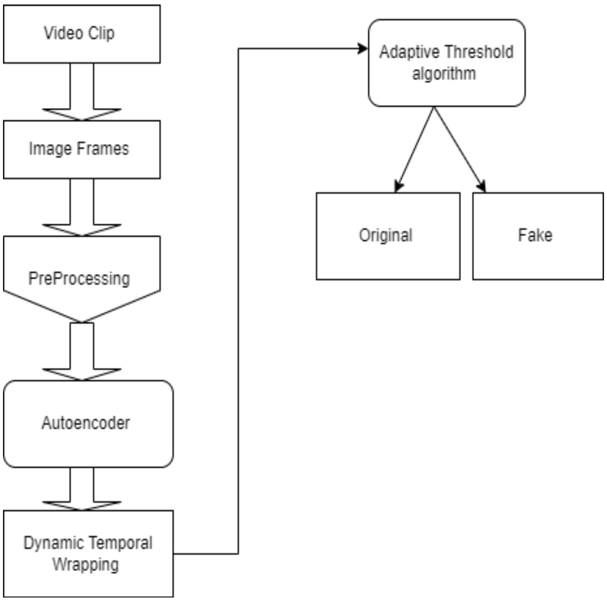

Fig. 1: Overall Flow of the proposed model

The figure shows that the video is first divided into many image frames, then pre-processing is done to improve contrast and eliminate noise. After that, auto encoders are used to compress and decompress face characteristics for feature extraction. Dynamic Temporal LSTM was introduced in which DTW corrects temporal misalignments by analyzing the frame sequence and correcting the frames despite frame rate variations to precisely identify tampering cues. Additionally, to account for the inherent frame rate fluctuations, an adaptive CNN is used, which incorporates an adaptive thresholding technique. The combined methodology proficiently addresses false positives and misclassification mistakes, improving the precision of tamper detection and discriminating between authentic and fake videos. The integrated methodology enhances the accuracy of tamper detection and distinguishes between original and tampered frame by effectively handling these false positives and misclassification errors. The proposed system is exhibited in terms of obvious configuration parameters in order to ensure reliability and technical transparency. The 32 and 64 convolutional filter have been applied consisting of 3x3 kernels for feature extraction in terms of spatial domain and reconstruction for the same, being trained with the Adam optimizer at a learning rate of 0.0001, using a batch size of 32 for 50 epochs. A bidirectional LSTM network with 128 units per direction and a dropout rate of 0.3, optimized with Adam (learning rate = 0.0002), makes up the temporal modeling stage. This setup, integrated with Dynamic Temporal Warping (DTW), aligns frame sequences of length 30 for effective temporal consistency learning. Finally, the Adaptive CNN classifier uses two convolutional layers with 64 and 128 filters (3×3 kernels), followed by a dense layer of 256 units and a dropout rate of 0.4 before the output layer (sigmoid activation). The CNN is trained for 60 epochs, with binary cross-entropy loss, Adam optimizer (learning rate = 0.0001), and validation split of 0.15. These hyperparameter settings were empirically tuned to achieve an optimal balance between accuracy, sensitivity, and computational efficiency (15 seconds per video) on the FaceForensics++ dataset.

After data collection, a video clip consisting of having manipulated or edited content is first entered into the procedure. This video clip is made up of a series of frames that were taken all at once. The video is broken up into several frames. Every frame in the video is a single moment in time captured on video. Once the video is broken down into frames, pre-processing techniques are applied to each frame. This step aims to remove noise, which could be caused by factors like camera sensors or compression artifacts. Median filtering is applied to clean up the frames. Cleaner data boosts specificity by decreasing false positives and decreasing the possibility of identifying fake tampering cues. Contrast stretching is then applied to enhance the contrast by expanding the pixel value range of an image to encompass the whole dynamic range. It entails mapping an image's minimum and maximum pixel values to new values that span the whole range of accessible intensity levels. This essentially broadens the intensity distribution, increasing the contrast between various components in the image as shown in Figure 2. Following preprocessing, each frame undergoes feature extraction using autoencoders, which is explained in the following section.

After preprocessing the compression and decompression of facial and frame features is done with the help of an auto encoder. Autoencoders extract meaningful features from the pre-processed frames and consist of two parts encoder and decoder. The input data consists of video frames containing facial features and expressions. Each frame is typically represented as a matrix of pixel values. The part of autoencoder called as encoder takes each video frame as an input and then constricts the size of facial features into a lower dimensional representation which is termed to as latent space while as retaining essential information. The input frame given as (Y) is being mapped into a hidden representation (Z) by an encoder function (f). Among the important elements that this hidden representation captures are facial landmarks, emotions, and textures. The formulation of the encoder process is shown in the Equation (1) given below:

$$Z = f\left(Y\right) \ = \ Sf \ \left(WY + BY\right) \qquad (1)$$

The *W* in above equation represents the weight matrix, and bias vector *B ε Rn* defines the state of the encoder and nonlinear activation function is represented by Sf. The output represented as *Z* act as compressed version of the facial image which keeps in check only the related information for detection. In order to make the process of analysis easier the dimensionality of facial features is being reduced very effectively by the autoencoder. Followed by this comes the decoder part which takes the compressed representation as an input acquired from the encoder and thus rebuilds the original facial features aiming to minimize the reconstruction loss. The hidden representation *(Z*) is mapped back to reconstructed *Y*' facial feature by decoder function as represented in the following Equation (2):

$$Y = f(g) = Sg(W'Z + Bz) \qquad (2)$$

Here the decoder's activation function is represented by Sg. By curtailing a reconstruction loss function, the autoencoder is being trained on a dataset of facial images. Thereby ensuring that the reconstructed image resembles very closely to the original input which enhances tampering detection.

The loss function measures the difference between the initial input face image *(Y*) and reconstructed images(*Y'*) as shown in the below Equation (3):

$$\theta = \min \theta\, L(Y,Y) = \min \theta L(Y, g(f(Y)) \qquad (3)$$

This method minimizes computational complexity while efficiently capturing the most discriminative facial features. The autoencoders enable more effective processing and analysis by extracting important information from facial features and reducing the dimensionality. The feature extraction is further followed by Dynamic Temporal LSTM which examines the sequence of frames and identify any sort of variations over the time between the extracted frames and the original video frame reference.

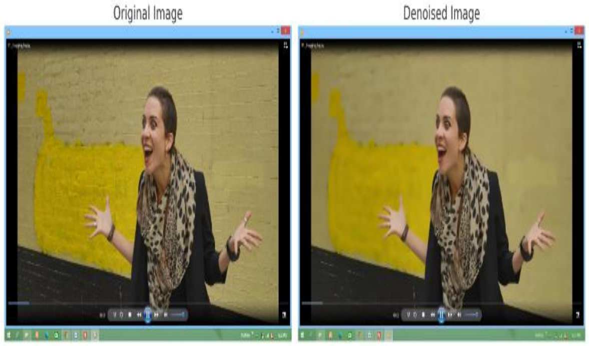

Fig. 2: Preprocessed Frame

The accuracy and reliability of the tampering detection is affected by the misalignments in the temporal domain. Dynamic Temporal Warping (DTW), which dynamically aligns frame sequences and detects subtle tampering cues, is integrated into Long Short-Term Memory (LSTM) networks to compensate for temporal discrepancies. The main input for LSTM is extracted input data. Each input sequence in this instance corresponds to a collection of frames take from a video. Here a novel DTW technique is presented to impart optimal alignment. The DTW characteristics such as resistance to timing errors, its ability to match sequences of varying lengths non-linearly in order to control different frame rates, tracking the temporal dynamics, ensuring the detection of even small tampering signs. In the beginning two sequences are aligned by warping the time axis in order to find out the optimal alignment between them. Given two sequences R = [r1,ri,.rn] represents the reference frames of length n and S= [s1, sj,.sm] representing the extracted frames of length m for analysis, where i and j are the indices of each time step of sequences R and S respectively. The cost of aligning a specific pair of frames from the two sequences is represented by each element of the cost matrix that DTW creates. The similarity measure between the frames is used to compute the cost. Euclidean distance between sequences named as R and S calculates the DTW distance where in *D (r, s)* is represented as the DTW distance as represented in the following Equation (4):

$$D(R,S) = \sum \left(i`, j`\right) \varepsilon M \parallel Ri`- Si` \parallel \tag{4}$$

Equation (4) is used to guarantee precise alignment in spite of timing errors or variations in frame rates thanks to the set of matched pairs M. Then, the optimal alignment path is found using dynamic programming (accumulated cost matrix) to minimize the total cost. DTW adjusts for temporal misalignments by allowing for flexible matching between frames, even if they occur at different points in time. To manage sequences of varying lengths and handle temporal misalignments accurately, DTW applies an asymmetric slope constraint, as described in Equation (5):

$$D(i, j) = \parallel r\text{�} - s\text{�}\parallel + min\, j`\text{€}\{j, j=1, j=2\}\, D(i-1, j`) \tag{5}$$

This assures that the number of local distances is always the same as the number of parts of the reference sequence. This allows for non-linear mapping between time-series data. This means that it stretches or compresses sections of the frame sequences to find an optimal alignment. If two video sequences have differing frame rates, DTW aligns the frames to compensate for the disparities. DTW allows non-linear alignments, thus it can stretch or compress bits of one sequence to match the time of another. After the accumulated cost matrix D has been calculated, the best alignment path is chosen by tracking back from D[m,n] where m and n are the lengths of the reference and extracted sequences, respectively to D[0,0]. The indices of frames in the reference and extracted sequences that are aligned match the alignment path. Further by following the alignment path the aligned sequences R `and S` by can be constructed. Thereby allowing DTW to simultaneously handle video sequences of varying length and thus identifying temporal misalignments accurately. This adaptability of DTW allows it to detect even the small irregularities or tampered cues in the video data and effectively overcoming temporal distortions, accurately aligning frame sequences despite different frame rates or timing errors. By determining the best alignment, DTW guarantees that the sequences' corresponding frames match, lowering false positives and improving tamper detection accuracy.

The aligned sequences are then fed into the LSTM layer, which takes the sequence of frame representations as input and processes them one by one, maintaining an internal state that captures temporal dependencies and detects tampering inconsistencies effectively. At each time step (t), the LSTM processes a frame representation along with the information stored in its memory cell from the previous time step. A forget gate, an output gate, and an input gate are some of the parts that make up the LSTM unit. The forget gate helps the LSTM ignore irrelevant information from previous frames, such as background motion or features that are not related to tampering. This gate discards unnecessary data, reducing the risk of false positives from unrelated content. The input gate controls which new features from the current frame, such as facial landmarks, facial expressions, and frame inconsistencies, should be stored in memory. These features are crucial for identifying tampering. The output gate filters the final output, ensuring that only relevant features, such as misalignments in frame sequences or subtle tampering cues, are passed to the next layer for further analysis. By effectively managing the retention of key features and discarding irrelevant ones, these gates ensure the LSTM is sensitive to genuine tampering cues while avoiding false positives.

The Algorithm for DTW is shown below:

Algorithm 1: Dynamic Temporal Warping (DTW)

```
Input: Reference sequence R = [$r_1$,……..$r_i$,…….$r_n$]
Extracted sequence S=[ $s_1$,…..$s_j$,….$s_m$]
   1. Initialization:
          Create a cost matrix C,C[0,0] =0 and set all other
          elements to infinity.
   2. Compute Local Distances
          For each pair of frames ($r_i$,$s_j$) compute the local
          distance using a similarity measure using equation
          (4)
```

```
    3. Dynamic Programming for Accumulated Cost Matrix
            For $i$ from 1 to n:
                    Compute $D$[i,j] using equation (5)
    4. Backtracking:
            Trace back from $D$[n,m] to $D$[0,0] to find the optimal alignment path
    5. Construct aligned sequences R`and S` by following the alignment path
Output: Aligned Sequences R`and S`
```

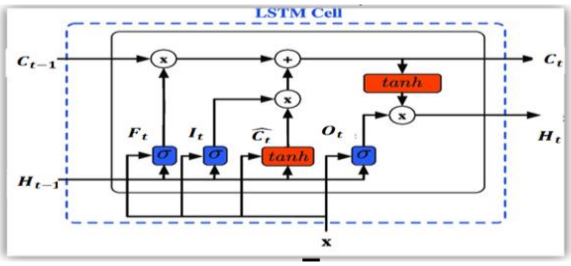

Fig. 3: LSTM Architecture

$$I_t = \sigma\left(W_t H_{t-1} + W_i X_i + B_i\right) \tag{6}$$

$$O_t = \sigma\left(W_o H_{t-1} + W_o X_i + B_o\right) \tag{7}$$

$$Z = f\left(Y\right) \; = \; Sf \; \left(WY + BY\right) \tag{8}$$

$$C\hat{}_{t=} tanh \; \left(Wc\hat{} \; H_{t-1 \; + \; W \; c\hat{} X \; i} + \; B \; c\hat{}\right) \tag{9}$$

Where X_t is the input at the time-step, k_{t-1} is the computed LSTM output at the previous cycle, σ is the symbol for the sigmoid activation function, crucial for controlling the flow of information within the LSTM. It squashes the input values into a range between 0 and 1, effectively determining how much information from the current frame should be passed through the network and retained in memory. H_{t-1} is the hidden states from the previous time step, $C\hat{}_t$ represents the intermediate vector of the cell state, In the LSTM model, each LSTM neural unit will go through a process of C_{t-1} to Ct, where Ct is the current memory content, and C_{t-1} is the memory content of the previous moment B_i, B_f, B_c, and B_0 provide bias vectors, and W is the weight matrix. The new input to the cell's state is selected by the input gate. Using the results of the previous stages, the new C_t is computed as follows in Equation (10):

$$C_t = \; C_{t-1} + \; I_t . C\hat{}_t \tag{10}$$

For each frame, the LSTM predicts the next frame's features and updates its hidden state. At time step t the hidden states of LSTM are designated as Ht, which is expressed in Equation (11):

$$H_t = \Phi\left(W_i, \; \left[\; H_{t-1}, x_t \right] \; + \; B_k\right) \tag{11}$$

LSTM's hidden state carries temporal information across frames, allowing the model to detect inconsistencies between predicted and actual frames, and also, successfully capture the dynamic aspect of frames. During the training phase, the LSTM network is presented with pairs of aligned sequences: One containing genuine reference frames and the other containing potentially tampered frames. The network learns to distinguish between these two types of sequences by adjusting its internal parameters through a process called back propagation through time, where gradients are calculated and used to update the parameters of the network. Each fame sequence is being processed by LSTM by extracting relevant features and thus assimilates the patterns for recognition indicative of modifications. Sudden swaps in the video sequences, motion or texture inconsistency or other deviations from the expected behavior of actual video footage. Given an initial set of frames, the LSTM is used to predict subsequent frames in a sequence after it has been trained. The actual frames are further feed into the LSTM which helps in generated the predicted frames. In order to check for inconsistencies these predicted frames are then compared with the actual frames. The LSTM-DTW model decreases the rate of false positives by aligning frame sequences and compensating for temporal misalignments. Then aligned frames are being processed by LSTM network while focusing on important temporal dependencies and filtering out noise and blur thus decreasing the false positive rate. It can be interpreted that by decreasing temporal inconsistencies, the model increases tamper detection rate and reducing false alarms. Additionally, it correctly detects discrepancies between video frames, enhancing the precision and dependability of tamper detection systems in practical settings.

Classification of Original and Tampered Videos Using Adaptive CNN

To increase video classification accuracy by addressing variability in frame rate differences, an adaptive CNN is introduced, in which CNN is employed for video classification tasks and an adaptive algorithm dynamically adjusts thresholds based on the observed frame rate variations. The adaptive algorithm continuously monitoring the frame rate variations within the video frames, calculating the mean and variance of these variations. Based on these calculations, the algorithm adjusts the classification threshold, allowing the CNN to effectively differentiate between authentic videos and tampered ones, even when there are natural fluctuations in frame rate. In real-world scenarios, a smart phone video exhibit slight frame rate fluctuations between 24-30 fps due to factors like changing lighting conditions or different camera settings [26]. In this case, the algorithm detects the natural variation and adjusts the threshold slightly to accommodate the fluctuations, preventing misclassification of the video as tampered.

On the other hand, if a tampered video is detected where the frame rate remains constant at 30 fps, but the video exhibits significant frame rate manipulation, such as abrupt changes in the frame rate during tampering, the algorithm adjusts the threshold more aggressively to detect these discrepancies as tampering cues [27]. This approach ensures that the model adapts to real-world frame rate changes while still identifying tampering cues. The architectural diagram of Adaptive CNN is illustrated in the Figure 4.

CNNs are extensive deep-learning models that are used for image-video classification applications. It is composed of many layers, including convolutional, pooling, and fully connected layers. Initially, input images are provided to the convolutional layers, where filters are applied to extract different features, and these filters are adjusted throughout the training stage.

These filters slide across the input images, performing element-wise multiplication with local regions of the image and producing feature maps. The filters in the deeper layer's extract more intricate features such as temporal dynamics and frame rates. Usually, a convolutional layer consists of several filters, each of which convolves across the input image to create a feature map. The convolution layer is calculated as follows in Equation (12):

$$F\hat{}_j = \sum_i F_i * M_{i,j} + B_j \tag{12}$$

Where F_i denotes the i^{th} input feature map, $F\hat{j}$ is the j^{th} output feature map, $M_{i,j}$ represents the convolutional kernel, and the bias is represented as B_j. Following each convolutional layer, pooling layers are implemented. These layers execute reduction procedures to lower the spatial dimensions of the feature maps, therefore decreasing computational effort and extracting dominant characteristics. The pooling layer uses the approaches of average pooling or maximum pooling to average or maximize the convolutional layer's output features.

After the pooling layer, a novel adaptive thresholding algorithm is introduced in the CNN. The purpose of this layer is to dynamically adjust thresholds based on observed frame rate variations.

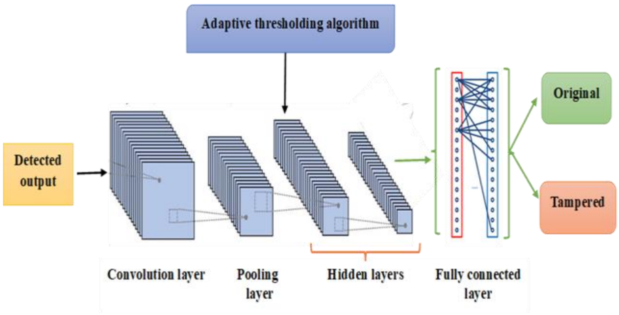

Fig. 4: Architectural diagram of Adaptive CNN

Adaptive thresholding in video classification allows for dynamic adjustment of decision-making thresholds based on the observed frame rate variations, ensuring the model's robustness to temporal inconsistencies. The algorithm first calculates the frame rate *(f)* of the video as the reciprocal of the time difference between consecutive frames. Then detects variations in frame rate in the video input. It tracks variations in each frame rate over time. The frame rate fluctuations based on statistical characteristics are determined, such as mean and variance are used, which are expressed in Equations (13-14):

$$\mathbf{\mu_f = 1/n \sum^n_{i=1} f_i} \tag{13}$$

$$\sigma_f^{\,2} = 1/n \sum^n_{i=1} \left(f_i - \mu_f \right)^2 \tag{14}$$

Where f_i is the sequence of frame rate, n is the total number of frames, σ_f^2 is the variance of frame rate, and μ_f is the mean frame rate. After calculating frame rate variability, the Adaptive Thresholding Algorithm calculates the initial threshold *(T_0)* based on the variability of frame rates, using the following Equation (15):

$$T_0 = k * \sigma_f^{\,2} \tag{15}$$

Where k is a constant factor that can be adjusted based on the desired sensitivity to frame rate variations. For each frame, adjust the threshold (T) based on the current frame rate (f_t) and the previous threshold T_{t-1}, using the following formula (16):

$$Tt = \beta \; X \; f_t + \left(1 - \beta\right) x \; T_{t-1} \tag{16}$$

Where β is a smoothing factor that controls the rate of threshold adjustment. The adaptive threshold T_t is then used within the CNN for classification purposes. The adaptive thresholding algorithm continuously monitors and analyses frame rate discrepancies within the input video frames. Based on this analysis, it dynamically adjusts the decision thresholds of the CNN to better discriminate between authentic and fake videos. This adaptive mechanism allows the CNN to adapt to the inherent variability in frame rates, improving its accuracy in video classification tasks. The adaptive threshold value is then fed into fully connected layers, influencing the classification decision based on the dynamically adjusted thresholds. The output layer produces the final classification decision based on the features learned by the preceding layers. During the classification process, if the feature values obtained from the CNN surpass the dynamic threshold, the video is classified as fraudulent; otherwise, it's classified as authentic. The Adaptive CNN's final output is a classification decision, which indicates whether the input video is original or tempered.

This choice is based on the network's learned features, adaptive thresholds, and classification performance. The proposed model employs an adaptive thresholding mechanism that dynamically adjusts the decision boundary during both training and inference. During training, the threshold is statistically optimized based on the mean and variance of predicted probabilities and frame rate variations across the dataset. During inference, the system continuously recalibrates this threshold in real time according to the observed frame rate, temporal consistency, and output probability distribution for each video segment. This enables the model to maintain high sensitivity under fluctuating recording conditions and minimizes false positives caused by temporal irregularities or compression artifacts. Through this combined approach, this Adaptive CNN offers an effective solution to the problem of distinguishing between original and fake videos in the presence of frame rate variations. By dynamically adjusting decision thresholds based on observed frame rate discrepancies, this approach improves the accuracy of video classification and enhances the reliability of video authentication systems in real-world scenarios.

Results and Discussion

This section encompasses the considerable evaluation of the performance & results of the proposed model and guaranteeing the models enhanced efficiency in terms of deep fake detection. The dataset used for forgery detection in this work is Face Forensics ++. This dataset is made up of 1000 original sequences of the video which have been amended making use of Deep Fakes, Face2face, Face Swap and Neural textures. The source of the video clips is from 977 you tube clips and all of which have predominantly frontal faces without any hidden view. This makes it possible to create realistic looking forgeries using automated tampering techniques. The data is being used as a binary mask for segmentation and classification purposes. The training of the proposed model has been done on first 750 videos out of 1000 and validated with the next 125 and tested with the remaining 125. In order to ensure that all the samples either real or manipulated must be equally represented through the different phases like training, validation and testing the stratified random sampling is used for dataset partition. This perspective stave off the imbalance among class, decrease in the sampling bias and assures the identities shown up in one subset don’t show up in another thereby ensuring the fair and unbiased evaluation of the model’s performance. The Technical University of Munich created the FaceForensics++ dataset, which was made available to the public for use in scholarly research. It can be found at https://github.com/ondyari/FaceForensics, the official project repository.

In this section substantial analysis of the proposed tampered detection method is done especially focusing on accuracy, efficiency and robustness. The performance of the model is evaluated by making use of range of metrics and benchmarks which permit for comprehensive investigation in terms of efficacy for identifying and characterizing videos as real or fake as shown in Figure 5.

As the number of sample increases the accuracy increases in the proposed model.This relationship is illustrated in Figure 5. The proposed model achieves a maximum accuracy of 96.83% with 500 samples whereas the number of samples decreases to 100, and the accuracy drops to a minimum of 70%.The Adaptive CNN reduces misclassifications by dynamically modifying thresholds in response to changes in frame rate, which improves accuracy in differentiating between authentic and tampered videos.

Figure 6 illustrates the precision of the proposed model performance. The proposed model achieves a peak precision of 96.9% with 500 samples, while the precision drops to a minimum of 75% with 100 samples.

Autoencoders enhance the precision of tampered video detection by efficiently emphasizing discrepancies and decreasing false positives through the compression and decompression of face features.

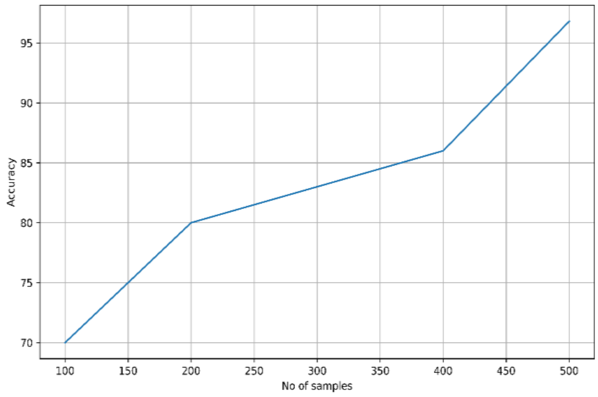

Fig. 5: Accuracy of the proposed model

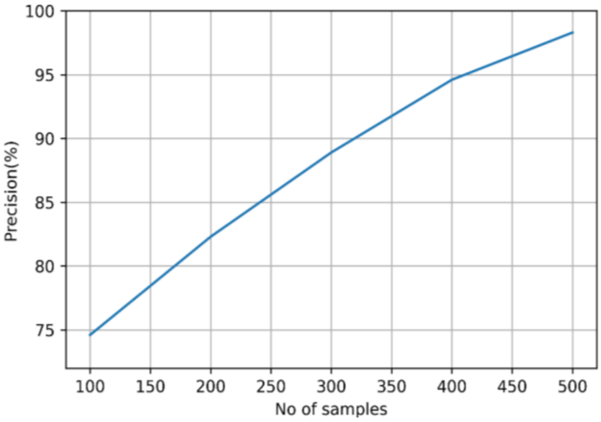

Fig. 6: Precision of the proposed model

The recall of the proposed model is shown in the Figure 7. In 200 samples, the proposed model has a recall of 87.17, 82.3 in 300 samples, 90 in 400 samples, and 96.9% in 500 samples. The greatest recall value achieved by the proposed model is 96.9%. Integrating Dynamic Temporal Warping into LSTM allows the identification of tiny tampering indications over time, which improves recall by minimizing the number of tampered frames.

The F1-score performance of the proposed model is displayed in Figure 8. The proposed framework shows an evident boost in F1-score performance as the number of samples increases. The F1-score achieves a high of 97.3% for 500 samples, on the other hand, the F1-score drops to a lower value of 76.38% with a smaller sample size of 100.By efficiently identifying real cases of tampering and reducing false positives and negatives, the combined strategy optimizes the F1-score while achieving a balance between precision and recall.

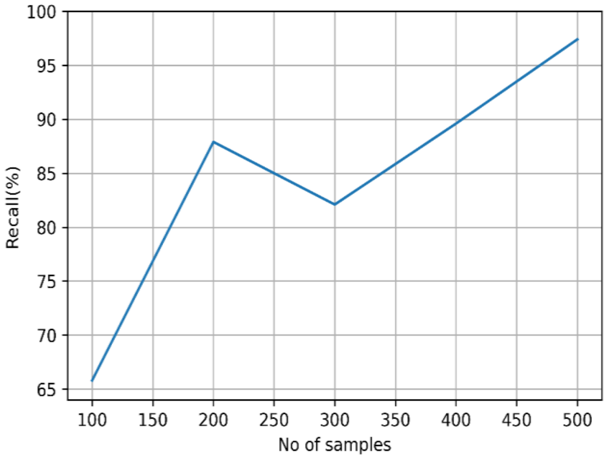

Fig. 7: Recall of the proposed model

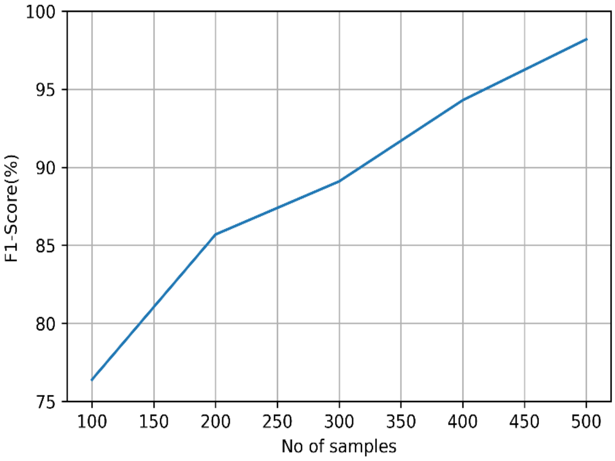

Fig. 8: F1-score of the proposed model

The detection accuracy of the proposed model is revealed in Figure 9. In 100 samples, the proposed model has a detection accuracy of 70, 83.2 in 200 samples, 90.15 in 300 samples, 93.4 in 400 samples, and 97.3% in 500 samples. Here CNN retains high accuracy due to dynamic adaptation in frame rates.

Figure 10 demonstrates the sensitivity performance of the proposed model. The proposed model achieves a high sensitivity value of 98% and a low sensitivity value of 72.38% when the number of sample values is 500 and 100 respectively. When samples increase sensitivity of the proposed model also increases. The dynamic temporal LSTM focuses on the sequence of frames to detect discrepancies, ensuring that even slight tampering indications are detected, increasing the model's sensitivity.

The specificity of the proposed model is depicted in above Figure 11. In 100 samples, the proposed model has a specificity of 67.5, 85 in 200 samples, 92.4 in 400 samples, and 97.5% in 500 samples. The proposed model attains a low specificity of 67.5 and maximum specificity of 97.5%. Pre-processing techniques that effectively reduce noise and boost contrast increase the specificity of the detection system by decreasing the probability of non-tampered frames being highlighted.

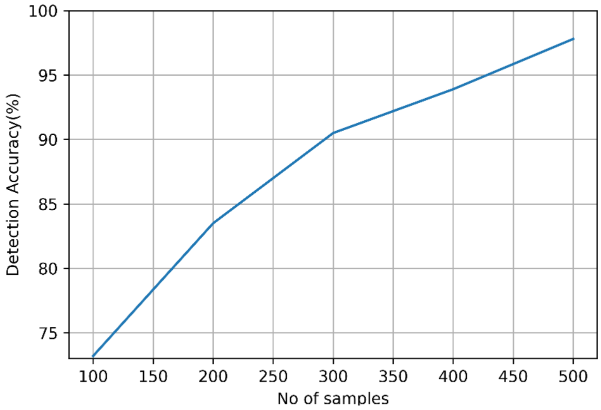

Fig. 9: Detection accuracy of the proposed model

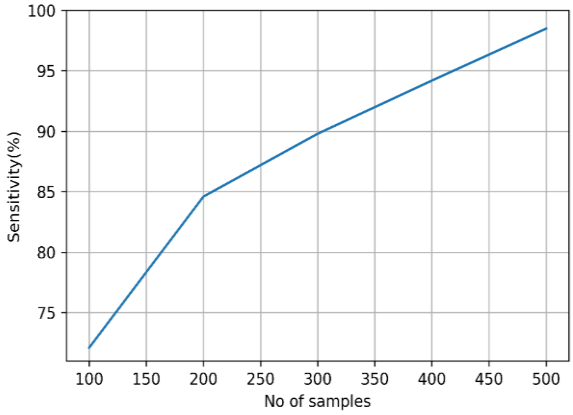

Fig. 10: Sensitivity of the proposed model

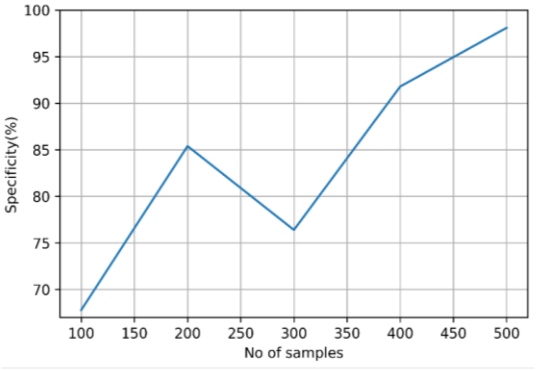

Fig. 11: Specificity of the proposed model

Figure 12 depicts the false positive rate of the proposed model. The proposed model achieves a maximum false positive rate of 5 and a minimum false positive rate of 2% when the number of samples is 100 and 500 respectively. By resolving temporal misalignments through the dynamic temporal LSTM, the system lowers the false positive rate by reducing the number of false positives.

The proposed model's AUC is depicted in Figure 13. The suggested model obtains the greatest AUC value of 0.99 when there are 500 samples, while it reaches the lowest AUC value of 0.65 when there are 100 samples. The enhanced adaptive threshold ability increases the AUC results in order to differentiate between original and forged clips.

The frame alignment performance of the proposed model is depicted in Figure 14. The error in the frame misalignment was 10 units before applying DTW and after the deployment of DTW it significantly reduced to just 1-2 units which represents quite significant reduction. Here the main role play is of DTW which aligns tampered video frames and compensate for frame rate variations and in turn guaranteeing more precise detection. By making corrections in these temporal misalignments, the proposed model increases precision in detecting precise tampering cues resulting into more genuine tampering detection.

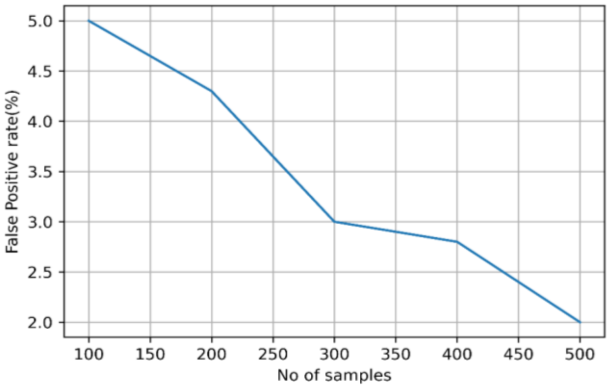

Fig. 12: False positive rate of the proposed model

Additionally, the results depicted in the Figures 15 and 16 highlight the proposed model's stability and adaptability across a dynamic range of video situations

The model has attained highest accuracy of 96.83 and sensitivity of 98% in regulated circumstances.

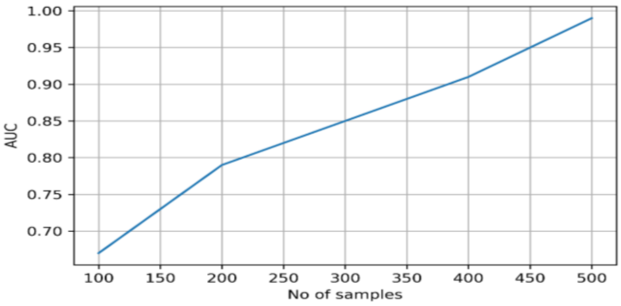

Fig. 13: AUC of the proposed model

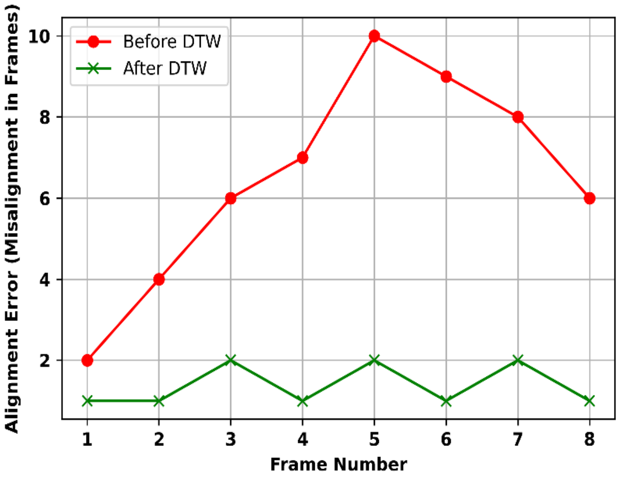

Fig. 14: Frame alignment before and after DTW

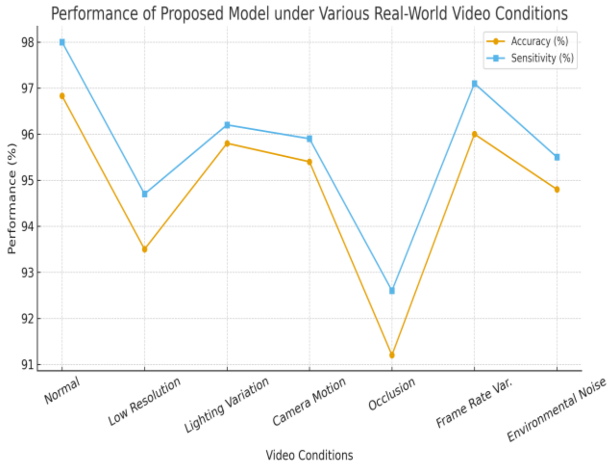

Fig. 15: Performance of the Proposed Model under Various Real-World Video Conditions

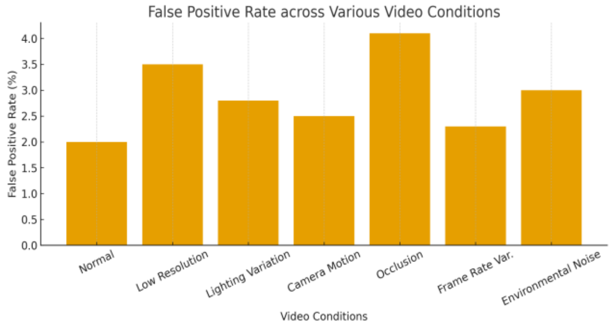

Fig. 16: False Positive Rate across Various Real-World Video Conditions

The model when exposed to circumstantial distortions like low resolution, variations in lighting and motion delay, the performance of system is maintained within a limited margin thereby manifesting its persistence. In order to efficiently compensate for the inconsistencies in temporal domain and camera motion, the dynamic temporal LSTM integrates with the Dynamic Temporal warping DTW and Adaptive CNN dynamically adjusts threshold value to fit in with variations in frame rate and other lighting changes. Consequently, the system shows compatible detection capability even when minor deviations in the authentic videos are there. Even the slightest decrease in the model's performance is observed underneath heavy scenarios based on occlusions where the facial visibility is either completely or partially invisible and thereby reducing the accessibility of trustworthy spatial cues. However, the false positive rate is still less than 4.1%, indicating high classification accuracy. To conclude it the results, illustrate that the proposed hybrid framework is well suited for implementation in real world digital forensic scenarios competent enough to differentiate between authentic and tampered videos in a variety of recording settings.

Comparative Analysis of the Proposed Model

In this section the comparative analysis of the proposed model with the current state of art is being done. The results were based on several metrics, including accuracy, recall, precision, false positive rate, sensitivity, specificity, error rate, and F1-score. The achieved outcome was explained in detail by comparing it with existing models such as ResNet, MobileNet-v2, SVM, KNN, FOA-SVNN and ECNN (Enhanced CNN).

Figure 17 compares the accuracy of the proposed model with different existing models such as SVM, MobileNet, and ResNet. The accuracy achieved by the proposed model is 96.83 as compare to the existing models like SVM with 90.25%, for Mobile Net 93.1 and 86.8% for ResNet. Thereby the existing model achieves the highest accuracy using adaptive CNN approach by continuously adjusting the frame rates and avoiding misclassification by some other sources.

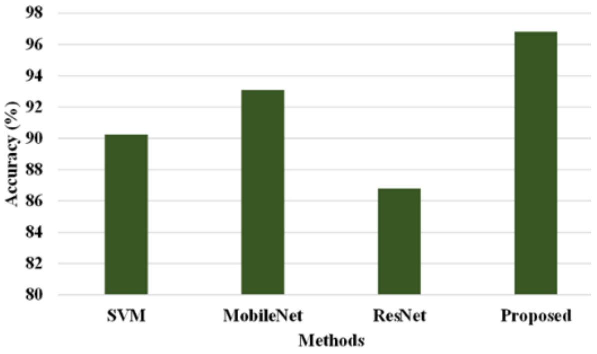

Fig. 17: Comparison of the accuracy of the proposed model with existing models

Further the model's sensitivity is being compared to other existing models as shown in Figure 18. In comparison to the sensitivity of the other models such as SVM, Mobile net and ResNet were 80,87 and 90% is achieved, the proposed model shows the maximum sensitivity of 98% by making use of dynamic temporal LSTM which in turn improves sensitivity by addressing the major issue of frame misalignment between frames of the video and considered to be very crucial for the detection of tampered content in the videos. By aligning frame sequences over time, the DTW technique fixes these misalignments and allows the model to precisely capture minute changes in the video.

The specificity of the proposed model with the existing models is shown in Figure 19. The proposed method achieves the specificity of 97.5 outperforming the SVMs 80, and both MobileNet and ResNet's 91%. The possibility of detecting specificity in videos is increased by directing the problems in temporal misalignments. This trait of the model aligns frame sequences thereby reducing the risk of false positives and making sure of the correct classification of actual frames turning the model to be much definitive. Thus, illustrating how the model is best suited for differentiating between original and tampered frames.

Figure 20 compares the computation time of the proposed model with the existing models such as SVM, Mobile Net and ResNet. With the fastest computation time of 15 seconds, the suggested model outperforms the existing models where Mobile Net and ResNet shows the computation time of 31 seconds and 33 seconds. Taking the variations in the frame rate as the base, the proposed method enhances the computation time by adjusting the threshold values. This enables CNN to classify videos more quickly and with less computational overhead by eluding recalculations when there are no notable frame rate differences thus resulting in faster classification and reduced computational overhead.

Figure 21 depicts the comparison of the error rate of existing models such as SVM, MobileNet, and ResNet, with the proposed model. With the lowest error rate of 6 the proposed model notably outperforms the other models. In contrary to it the existing models such as Mobile Net, ResNet and SVM have higher error rates of 9.8, 10.6 and 14.02.

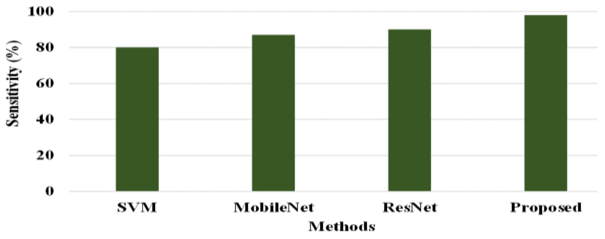

Fig. 18: Comparison of the sensitivity of the proposed model with existing models

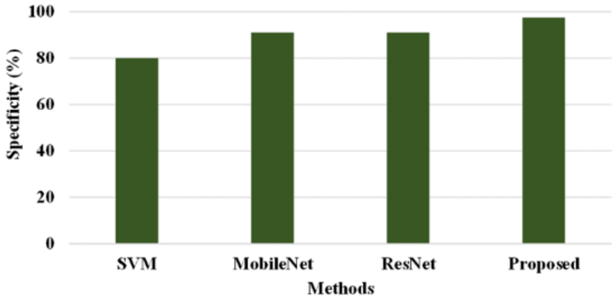

Fig. 19: Comparison of specificity of the proposed model with existing models

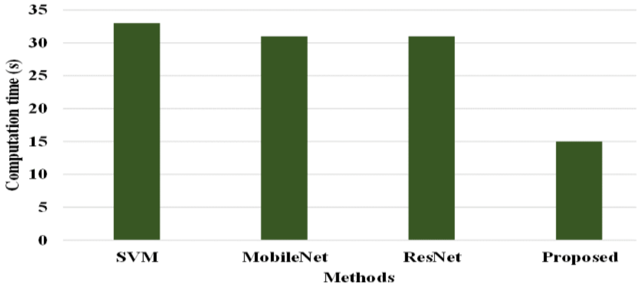

Fig. 20: comparing the computation time of existing models with the proposed model

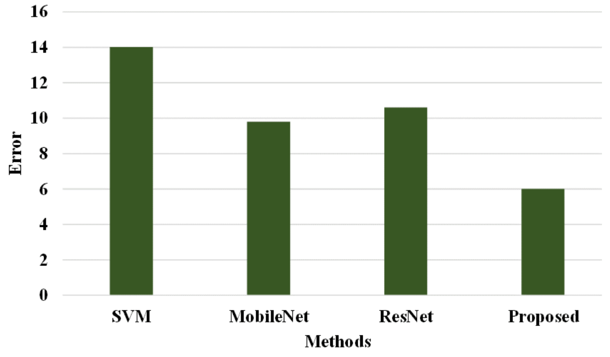

Fig. 21: Comparison of error rate of existing models with the proposed model

The features of adaptive CNN incorporated with the adjusting thresholds reduce s the misclassification errors caused by frame rate variations in the actual scenario. In comparison to the current methods, this reduces the error rate and misclassifications by compensating for natural frame rate variations and guaranteeing that the model can differentiate between real and manipulated videos.

Further the AUC of the proposed model is compared with the existing models such as SVM, MobileNet and ResNet which are depicted in Figure 22. The model achieves the remarkable AUC of 0.99. In comparison to SVM the AUC achieved is 0.784 whereas MobileNet and ResNet AUCs of 0.945 and 0.921, respectively. The dynamic temporal LSTM with Adaptive CNN employs the adaptive thresholding based on the current frame rate and thereby increasing the model's capability of distinguishing between original and fake clips and which eventually increases the AUC. Thus, in contrast to the existing methods the proposed model exhibits superior performance in accurately identifying tampered cues.

In addition to it, the false positive rate of the proposed model is compared to the existing model as depicted in Figure 23. The existing models exhibit the false positive rate of 4 in SVM, 6.67 in Mobile Net and 3.33% in Res Net. Whereas the proposed model attains the lowest false positive rate at just 2% representing its efficiency in lowering the imprecise classifications as compare to the other models discussed. Irrespective of the timing differences the model very correctly adjusts the misalignments and thereby reduces the inconsistencies and false positive rates by capturing these minor altercations.

The comparison of true positive rate of proposed model with the existing models is shown in the Figure 24. The proposed model overcomes the existing models like SVM, MobileNet, and ResNet in terms of true positive rate, with an amazing value of 98% while the SVM ,MobileNet and ResNet depicts low true positive rate of 97.78,97.78 and 95%. This improvement in the proposed model is caused by the Dynamic Temporal LSTM, combined with Dynamic Temporal Warping (DTW), which improves the True Positive Rate (TPR) in video tampering detection by handling temporal misalignment and reducing false positives. DTW adjusts frame timing, allowing LSTM to accurately capture tampering cues and detect subtle variations in sequences, thereby reducing false positives.

The F1-score of the proposed model is compared with existing models such as ECNN, FOA-SVNN, SVM, KNN, and NN which is shown in Fig. 25. Relatively, the existing models such as SVM, KNN and NN perform ata very low F1-score of 78.24, 70.27, and 79.56%, while on the contrary ECNN, FOA-SVNN show good performances with an F1-score of 97.09, and 94.9%, respectively. On the other hand, the proposed model notably outperforms the existing models by attaining highest F1-score of 97.3%. Here the model uses dynamic temporal LSTM with Dynamic temporal warping which in turn increases the F1 score by taking into consideration the misalignment among the frames and thereby decreasing the false negatives. Thus, managing the variations existing in the frame rates and guaranteeing accurate categorization.

Figure 26 exhibits the collation among precision of existing models with the proposed model. The existing model such as MobileNetV2 reveals 82.9% of precision and Squeeze Net depicts 81.1% of precision.

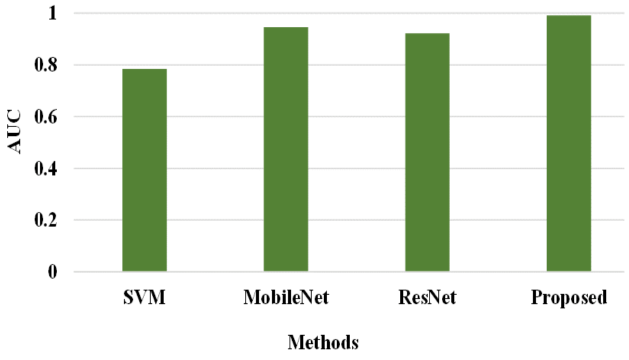

Fig. 22: Comparison of AUC of existing models with the proposed model

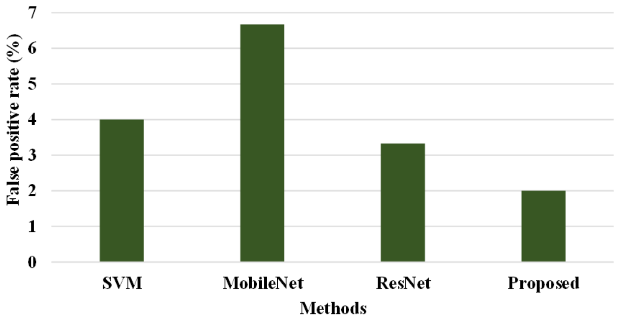

Fig. 23: Comparison of false positive rate of the proposed model

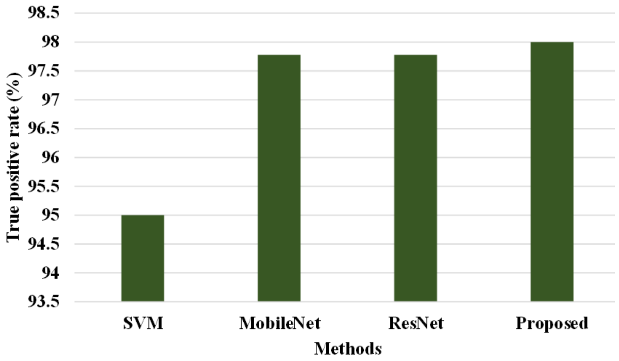

Fig. 24: Comparison of the true positive rate of the proposed model

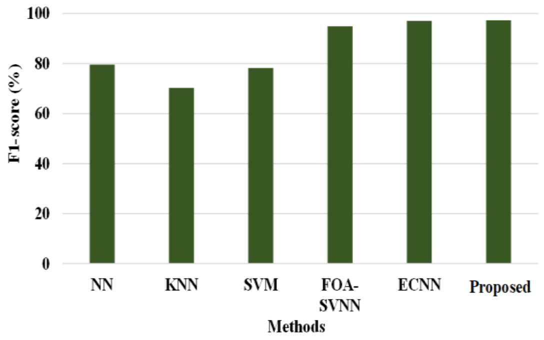

Fig. 25: Comparison of F1-score of existing models with the proposed model

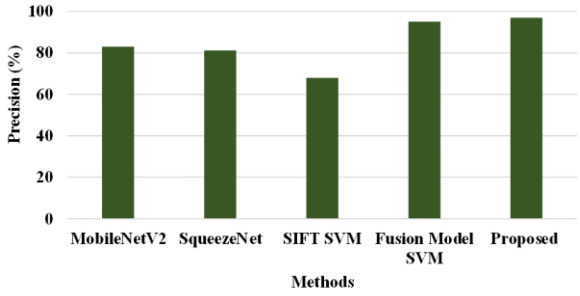

Fig. 26: Comparison of precision existing models with the proposed model

At the same time the current state of art methods like SIFT and SVM demonstrate great decrease in performance with 67.9% of precision. Compared to the current approaches, it is especially effective for dynamic scenes because it lowers false positives, enhances model performance, and reaches a peak precision of 96.9%.

Figure 27 exhibits the comparison of recall measure with the proposed model. Here MobileNet, SqueezeNet, SIFT, SVM, and Fusion Model SVM give recall values of 81.2, 79.9, 69, and 96.1% recall whereas the proposed model achieves great recall rate that is 969%. By addressing temporal misalignments between video frames, the Dynamic Temporal Warping (DTW) integration within the Dynamic Temporal LSTM detects subtle tampering signs and greatly improves recall.

Further the classification success CS of proposed model is compared with the current existing models as shown in Figure 28. CS values of 84.2, 83.6, and 87.5% are attained by the current techniques, which include CNN, LSTM, and CNN + LSTM. The proposed model transcends as compare to other existing models by attaining the CS rate of 91%. By incorporating DTW, frame sequences despite different rates and timing errors, performs exceptionally well in classification. Thus, allowing for precise comparison over time and detection of temporal inconsistencies.

In Figure 29. comparison of the proposed model's Cross Validation Accuracy is done with the existing model. In the figure it is shown that existing methods like CNN, LSTM, and CNN + LSTM, achieves CVA value of 91.6, 90.8, and 93.2%. On the contrary the proposed model achieves CVA of 95.2%. The proposed technique based on Dynamic Temporal LSTM handles temporal misalignments and frame rate variations in an efficient way and thereby improving cross validation accuracy. This hybrid approach increases the ability of model in order to accurately detect tampered cues and maintain time domain dependability. The confusion matrix depicted in Figure 30 interprets classification result of proposed model on Face Forensics ++ dataset.

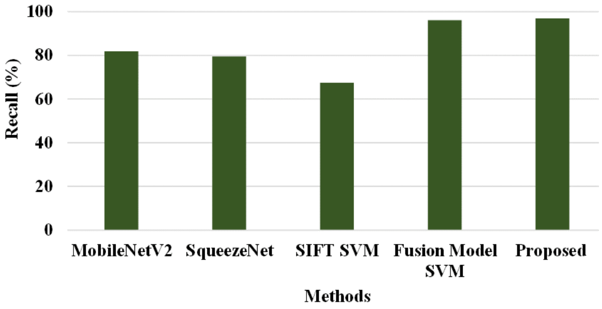

Fig. 27: Comparison of recall of the proposed model with existing models

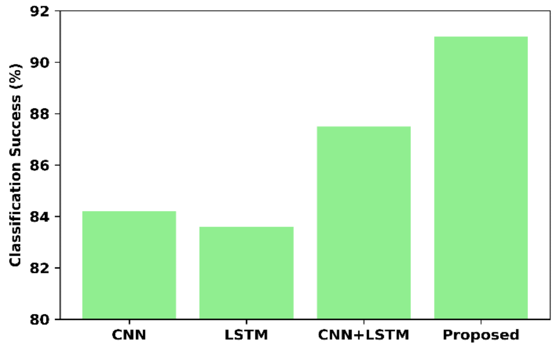

Fig. 28: Comparison of classification success of the proposed model with existing models

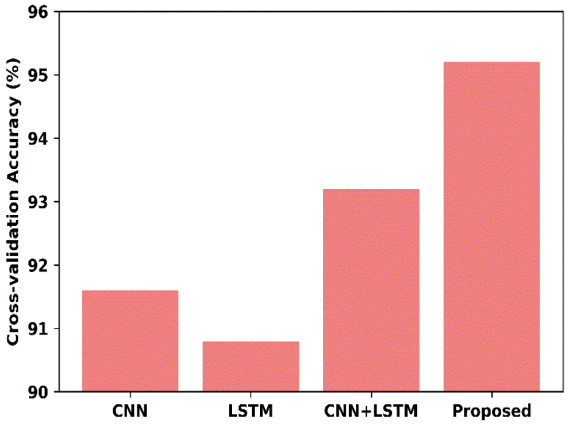

Fig. 29: Comparison of cross-validation accuracy of the proposed model with existing models

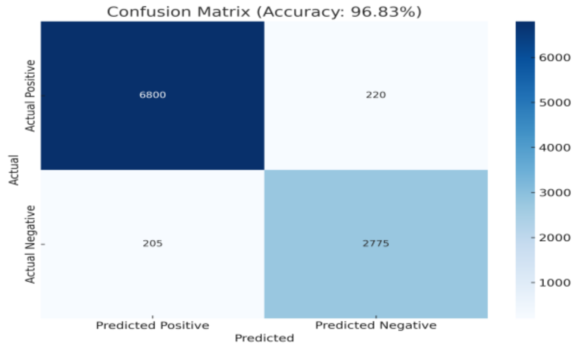

Fig. 30: Confusion Matrix

Additionally, the feasibility of real time deep fake model is shown by run time analysis as depicted in the Figure 31 and 32. There a comparative run time evaluation between CNN, LSTM and CNN+LSTM models is done under identical hardware settings (NVIDIA RTX 3090 GPU, 32 GB RAM).

The CNN model depicts the latency of 12ms/ frame which is followed by LSTM latency of 18ms/frame. Whereas the combined approach of CNN+LSTM shows high latency of 25ms/frame due to temporal management. Near real-time performance at 30 frames per second is made possible by the time per frame staying below 33 ms.

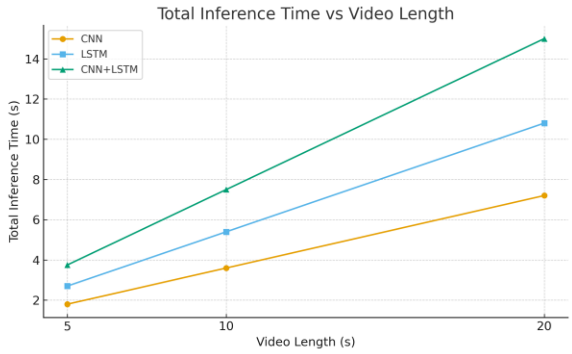

Fig. 31: Average inference time per frame

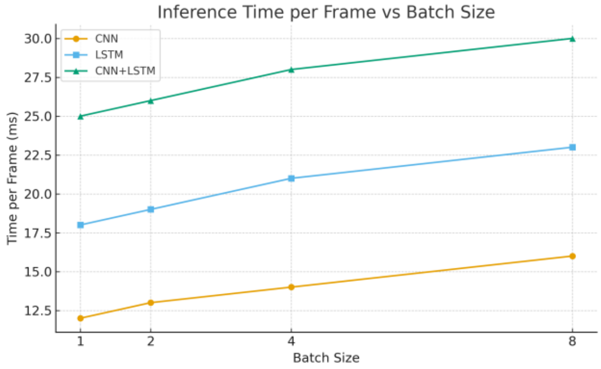

Fig. 32: Total inference time per frame

However, the time per frame remains below 33 ms, enabling near real-time performance at 30 FPS. In terms of Total Inference Time versus Video Length, The total inference time scales linearly with video duration for all models. The processing of video in proposed hybrid model is done in less time thus maintaining the scalability which shows 30 FPS video processed in about 7.5 seconds. The results show that the suggested approach of CNN +LSTM maintains a good balance between accuracy and real time scalability making it accurate for real time detection of deep fake detection. In order to ensure the statistical meaning of proposed model, a paired statistical significance test is conducted against the existing baseline models such as CNN, LSTM and CNN+LSTM. The balance among the existing models suggested both higher performance and statistically significant improvement. Multiple experimental deployments using a paired t-test ($\alpha = 0.05$) ensure the accuracy and F1 score improvements of the proposed Dynamic Temporal LSTM amalgamated with Adaptive CNN with ($p<0.05$) across all baseline models. The outcomes substantiate the solidity and dependability of the proposed architecture under diverse video conditions

Conclusion and Future Scope

In conclusion the hybrid approach based on Dynamic temporal LSTM with adaptive CNN has made a remarkable advancement in the domain of video tampering detection specifically the deep fake detection, where the data is very much vulnerable to the modifications by easily available editing software tools thereby striking a balance among the originality and integrity of the video data. The proposed method has provided an adequate basis for tampering detection by taking into consideration the parameters such as temporal misalignment and fluctuations in the frame rate. The major superiority of the given approach is incorporating the DTW into LSTM framework for looking into even minor tampering signs in the video frames. Irrespective of the variation in the frame rates, the usage of DTW aligned the frame rates and thereby allowing comparisons and tampering detection cues. In addition to it the continuous moderation in the threshold value with respect to the fluctuations in the frame has notably enhanced the performance of the model and escalated the video classification accuracy. According to Table 1, the model's success in identifying tampered videos while reducing false positives is demonstrated by its highest sensitivity of 98, specificity of 97.5, accuracy of 96.83, and F1-score of 97.3%.

Table 1: Comparison of proposed model with existing methods

Metrics	500 Samples	100 Samples
Accuracy	96.83%	70%
Precision	96.9%	75%
Recall	96.8%	87.17%
F1-Score	97.3%	76.38%

Additionally, the proposed model exhibits the considerable refinement in the computational time and rate of error as compare to the state of art techniques, thus making sure about high accuracy and efficiency in tamer detection. In terms of its robust performance particularly while dealing with these misalignments and variations in the frame rate, the model turns out to be efficient tool in tackling the dissemination of the forged content for various malevolent activities. In connection with the models' hands on usage in the field of media affirmation, content moderation and criminal examination it has turned out to be very capable. The model provides an efficient way to validate videos and recognize any false information present thereby assisting legal procedures by tackling both temporal and spatial inconsistencies. The performance of the model can be further refined to handle long videos and much more complex tampering irrespective of the systems robustness in terms of handling temporal misalignment and frame rate variation. The limitations stated in the current study is that Video-level classification is carried out, which uses aggregated frame features to classify a whole video as either authentic or tampered. The precise frames or facial areas where manipulation takes place are not specifically identified or highlighted. This restricts its use in forensic investigations where accurate localization of tampered segments is necessary. In addition to it only Face-swapping and expression-based manipulations are the main subjects of the study. It excludes more recent generative models that are quickly appearing in real-world settings, like multi-modal forgeries involving audio-visual synchronization or diffusion-based deepfakes.

In future, techniques like Faster R-CNN and Deep Q-Networks will improve video content detection for subtle edits, extending their impact to multimedia forensics, media verification, content moderation, and digital evidence analysis in criminal investigations, thereby enhancing the applicability of these techniques.

With respect to most recent findings in video tampering detection the future work must be focused on integrating transformer architectures. These vision based transformers help in captivating long range dependencies in the video streams thereby improving interpretability. Secondly the proposed model is validated on Face Forensics ++ dataset which prominently contains frontal and high quality facial data. The validation can be extended to more recent datasets such as DFDC, Celeb-DF and Wild Deepfake which represent real world with conditions such as motion blur, occlusions and diverse lighting.

Acknowledgment

We would like to thanks our institution for providing computational resources and environment required to carry out our research work.

Funding Information

The authors have not received any financial support or funding to report.

Author’s Contributions

Gurpreet Kour Khalsa: Wrote the original draft of the manuscript.

Rakesh Ahuja: Contributed to proof reading of the manuscript.

Rattan Deep Aneja: Contributed to the comparative results, detailed evaluations, ablation studies, and cross-dataset validations to strengthen the robustness and reliability of our findings.

Ethics

The authors don’t have any ethical issues that may arise after the publication of this manuscript.

References

Balasubramanian, S. B., Kannan, R. J., Prabu, P., Venkatachalam, K., & Trojovský, P. (2022). Deep fake detection using cascaded deep sparse auto-encoder for effective feature selection. *PeerJ Computer Science*, *8*, e1040. https://doi.org/10.7717/peerj-cs.1040

Balasubramanian, S. B., Kannan, R. J., Prabu, P., Venkatachalam, K., & Trojovský, P. (2022). Deep fake detection using cascaded deep sparse auto-encoder for effective feature selection. *PeerJ Computer Science*, *8*, e1040. https://doi.org/10.7717/peerj-cs.1040

BR, S. R., Pareek, P. K., Bharathi, S., & Geetha, G (2023). Deepfake Video Detection System Using Deep Neural Networks. *Proceeding of the IEEE International Conference on Integrated Circuits and Communication Systems (ICICACS)*. 2023 IEEE International Conference on Integrated Circuits and Communication Systems (ICICACS), Raichur, India. https://doi.org/10.1109/icicacs57338.2023.10099618

Chittapur, G., Murali, S., & Anami, B. (2020). Tempo Temporal Forgery Video Detection Using Machine Learning Approach. *Journal of Information Assurance & Security*, *15*(4), 144–152.

Deo, S., Mehta, S., Jain, D., Tiwari, C., Thorat, A., Mahara, S., Gonge, S., Joshi, R., Gite, S., & Kotecha, K. (2023). Video Tampering Detection Using Machine Learning and Deep Learning. *Advanced Computing, 1782*, 444–459. https://doi.org/10.1007/978-3-031-35644-5_36

Fadl, S., Han, Q., & Li, Q. (2021). CNN spatiotemporal features and fusion for surveillance video forgery detection. *Signal Processing: Image Communication*, *90*, 116066. https://doi.org/10.1016/j.image.2020.116066

Girish, K., & 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

Halak, B., Hall, C., Fathir, S., Kit, N., Raymonde, R., & Vincent, H. (2022). Intelligent Tamper Detection Systems using Machine Learning. *Proceeding of the IEEE International Conference on Design & Test of Integrated Micro & Nano-Systems (DTS)*, 1–5. https://doi.org/10.1109/dts55284.2022.9809885

Harika, L. P., Gayathri, D. B., & Priya, R. S. (2023). Video Tampering Detection in Real Time. *Intelligent Sustainable Systems*, *665*, 351–364. https://doi.org/10.1007/978-981-99-1726-6_27

Joshi, V., & Jain, S. (2020). Tampering detection and localization in digital video using temporal difference between adjacent frames of actual and reconstructed video clip. *International Journal of Information Technology*, *12*(1), 273–282. https://doi.org/10.1007/s41870-018-0268-z

Koshi, L., & Shyry, S. P. (2025). Detection of tampered real time videos using deep neural networks. *Neural Computing and Applications*, *37*(11), 7691–7703. https://doi.org/10.1007/s00521-024-09988-1

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

Luan, T., & Damian, M. A. E. (2023). A Study on the Application of Deep Learning-based Media Tampering Detection Technology in Higher Education Teaching Resource Protection. Contemporary Education and Teaching Research, 4(6). https://doi.org/10.47852/bonviewcetr232011480604

Mira, F. (2023). Deep Learning Technique for Recognition of Deep Fake Videos. *Proceeding of the IEEE IAS Global Conference on Emerging Technologies (GlobConET)*, 1–4. https://doi.org/10.1109/globconet56651.2023.10150143

Munawar, M., & Noreen, I. (2021). Duplicate Frame Video Forgery Detection Using Siamese-based RNN. *Intelligent Automation & Soft Computing*, *29*(3), 927–937. https://doi.org/10.32604/iasc.2021.018854

Nguyen, H. X., Hu, Y., Ahmad Amin, M., Gohar Hayat, K., Thinh Le, V., & Truong, D.-T. (2020). Detecting Video Inter-Frame Forgeries Based on Convolutional Neural Network Model. *International Journal of Image, Graphics and Signal Processing*, *12*(3), 1–12. https://doi.org/10.5815/ijigsp.2020.03.01

Osorio-Arteaga, F., & Giraldo, E. (2024). Adaptive Neural Network Identification for Robust Multivariable Systems. *IAENG International Journal of Applied Mathematics*, *54*, 68–76.

Pandey, S. K., Kumar, L., Kumar, G., Kumar, A., Singh, K. U., & Singh, T. (2023). An Overview of Video Tampering Detection Techniques: State-of-the-Art and Future Directions. *Proceeding of the International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, 171–175. https://doi.org/10.1109/cises58720.2023.10183511

Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Niessner, M. (2019). FaceForensics++: Learning to Detect Manipulated Facial Images. *Proceeding of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 1–11. https://doi.org/10.1109/iccv.2019.00009

Saddique, M., Asghar, K., Bajwa, U. I., Hussain, M., Aboalsamh, H. A., & Habib, Z. (2020). Classification of Authentic and Tampered Video Using Motion Residual and Parasitic Layers. *IEEE Access*, *8*, 56782–56797. https://doi.org/10.1109/access.2020.2980951

Saini, P., & Ahuja, R. (2024). Robust and Secure Video Authentication: A Hash-Based Watermarking Approach in IAENG. *International Journal of Computer Science*, *54*, 1291–1308.

Tariq, S., Lee, S., & Woo, S.-S. (2020). A convolutional LSTM based residual network for deepfake video detection.

Velliangira, S., & Premalatha, J. (2020). A Novel Forgery Detection in Image Frames of the Videos Using Enhanced Convolutional Neural Network in Face Images. *Computer Modeling in Engineering & Sciences*, *125*(2), 625–645. https://doi.org/10.32604/cmes.2020.010869

Xing, Q., Luo, Y., Zhang, Z., & Zhang, F. (2022). Video Inter-frame Tampering Detection Based on SN-VGG+BiLSTM-AE Composite Model. *Proceedings of the 2022 10th International Conference on Information Technology: IoT and Smart City*, 80–87. https://doi.org/10.1145/3582197.3582210

Zheng, Y., Bao, J., Chen, D., Zeng, M., & Wen, F. (2021). Exploring Temporal Coherence for More General Video Face Forgery Detection. *Proceeding of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 15044–15054. https://doi.org/10.1109/iccv48922.2021.01477