Cureus Journal of Computer Science

Open Access Original Article

Comparative Analysis of Convolutional Neural Network and Support Vector Machine Techniques for Video Forgery Detection

Received 03/26/2025
Review began 04/06/2025
Review ended 05/29/2025
Published 05/30/2025

© Copyright 2025
Elbarougy et al. This is an open access article distributed under the terms of the Creative Commons Attribution License CC-BY 4.0., which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

DOI:
https://doi.org/10.7759/s44389-025-04047-9

Reda Elbarougy [1, 2] , Osama Abdelfatah [3] , Gamal M. Behery [4] , Noha M. El-Badry [3]

1. Department of Artificial Intelligence and Data Science, College of Computer Science and Engineering, University of Ha'il, Ha'il, SAU 2. Department of Information Technology, Faculty of Computer and Artificial Intelligence, Damietta University, New Damietta, EGY 3. Department of Mathematics, Faculty of Science, Damietta University, New Damietta, EGY 4. Department of Computer Science, Faculty of Computer and Artificial Intelligence, Damietta University, New Damietta, EGY

Corresponding author: Noha M. El-Badry, noha_elbadry@du.edu.eg

Abstract

Advanced manipulation techniques like the Copy-Move Attack, which includes copying and changing video information, are posing a growing danger to the integrity of digital assets. The validity of digital media depends on the detection of such forgeries, but current techniques frequently find it difficult to strike a compromise between efficiency and accuracy in complex situations. This study tackles this problem by analyzing and contrasting two machine learning methods for video forgery detection: (1) a deep convolutional neural network (CNN) designed for forgery classification, and (2) a support vector machine (SVM) classifier paired with histogram of oriented gradients (HOG) for feature extraction. Both approaches were evaluated on a dedicated dataset of forged and original video frames from the REWIND database, which focuses on Copy-Move manipulations. Experimental results show that the HOG-SVM approach achieved an accuracy of 87.2% and an F1-score of 0.65, while the CNN method outperformed it with 93% accuracy and an F1-score of 0.95. Our contribution lies in presenting a detailed comparative analysis of these methods, highlighting the strengths and limitations of each. These findings underscore the superior capability of CNNs to learn intricate spatial and temporal features, making them a more robust solution for video forgery detection.

Categories: Application Security, Artificial Intelligence and Machine Learning in the Cloud
Keywords: machine learning, deep learning, cnn, svm, hog, copy-move forgery

Introduction

Video forgery is one of the important challenges that scholars are actively tackling due to its importance and danger in many areas. These practices represent a great danger to the digital world, as it is difficult to identify videos that have been manipulated due to the accuracy of the programs used and their constant updating. These practices represent a great danger to the digital world, as it is difficult to identify videos that have been manipulated due to the accuracy of the programs used and their constant updating. The ability to verify the authenticity of the video is important and a major challenge for researchers in order to ensure the validity of the data published to the public and prevent false information. The need for effective methods to detect video forgery has become more urgent than before due to the development of techniques used in forgery. This confirms the necessity and importance of developing more efficient techniques for detecting video forgery.

The process of forgery is carried out by manipulating objects present in video frames, either removing them or moving them to another place in the same frame or to another frame. Based on the forgery present in the video, the ideal methods for dealing with these manipulations in videos are determined.

Video forgery involves more than one technique, and Splicing and Copy-Move are the most well-known among them. Splicing is the merging of different video clips in order to create one video. More specifically, it involves manipulating the content of a video by adding objects or scenes from different source videos. On the other hand, Copy-Move forgery also involves manipulation, but by copying objects within the same video-adding, removing, or relocating them. While both techniques share the goal of altering video content, they differ in that Splicing introduces external elements, whereas Copy-Move manipulates elements already present in the video.

The danger of this issue is evident from the fact that, as programs and technologies used in forgery operations have advanced, it has become increasingly difficult and challenging for researchers to detect video forgery. This growing sophistication necessitates the development of more robust and effective approaches to ensure the integrity of digital video content.

How to cite this article
Elbarougy R, Abdelfatah O, Behery G M, et al. (May 30, 2025) Comparative Analysis of Convolutional Neural Network and Support Vector Machine Techniques for Video Forgery Detection. Cureus J Comput Sci 2 : es44389-025-04047-9. DOI https://doi.org/10.7759/s44389-025-04047-9

This study makes three contributions. First, it offers a methodical analysis and comparison of two machine learning-based approaches for video forgery detection: a deep convolutional neural network (CNN) designed for forgery classification and the support vector machine (SVM) classifier combined with histogram of oriented gradients (HOG) for feature extraction. This analysis offers important insights into each method's effectiveness, advantages, and disadvantages in detecting forgeries brought on by Copy-Move Attacks. Second, the suggested CNN-based framework outperforms conventional techniques and shows a greater ability to predict complex forging patterns by detecting forged video frames with greater accuracy. Lastly, the study conducts a thorough empirical evaluation on a benchmark dataset of forged and non-forged video frames, providing robust experimental results and establishing a reference point for future advancements in video forgery detection research.

Numerous researchers have undertaken the task of identifying efficient methods for detecting video forgery. One notable approach involves a fast key point-based feature matching algorithm for Copy-Move forgery detection [1]. This method extracts key points from the image, organizes them, and applies the generalized two nearest-neighbor (g2NN) algorithm to perform localized matching searches around each key point. By avoiding thresholds, this technique excels in detecting forgeries in areas with uniform or low texture. Furthermore, the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm is employed for clustering, effectively identifying fabricated regions.

Building on earlier work, scientists have investigated cutting-edge methods to improve video forgery detection. For instance, some studies [2-6] proposed fast key point-label groups to efficiently assign high-dimensional features extracted from videos to specific groups, thereby achieving improved accuracy. Additionally, a method utilizing the scale-invariant feature transform (SIFT) structure was introduced for feature extraction from Copy-Move forged videos, offering a robust spatial detection mechanism.

Significant progress was achieved by Pandey et al. [3], who integrated noise residue correlation methods for temporal analysis with SIFT for spatial detection, resulting in a more holistic strategy for video forensics. However, this method exhibited certain limitations, such as increased computational time required to complete the necessary steps before identifying video forgeries. Meanwhile, Su et al. [4] directed their focus toward identifying duplicated regions within video frames. They introduced an algorithm that employs exponential furious momentum to detect these duplicated regions, further contributing to the fight against video counterfeiting.

In other efforts, some academics created a forensic technique to identify digital video resampling-based intra-frame forgeries, including splicing and upscale-crop [5]. Their method successfully detects post-production changes in video frames by combining noise-inconsistency analysis and pixel-correlation examination, along with other key forensic techniques such as motion detection and artifact analysis. Through this combination, the system overcomes the specific shortcomings of each individual technique, improving the precision of tampered region detection. They tested multiple codecs, bit-rates, scaling factors, noise levels, and tampered region sizes to validate the system across a wide range of films and photos, showcasing its efficacy in various forensic scenarios.

Additionally, in an earlier attempt, some researchers addressed issues in smooth areas and regions exposed to rotation, scaling, or multiple pasting by introducing a two-step key point-based technique to detect Copy-Move forgeries [6]. Initially, they employed BRIEF (Binary Robust Independent Elementary Features) with FAST (Features from Accelerated Segment Test) descriptors for key points in textured regions and SIFT for key points in smooth areas. Next, the generalized second nearest-neighbor approach was used to match the points. These matches were then refined by morphological processing and the structural similarity index. For improved localization of forged areas, linear spectral clustering was then used. Experiments conducted on many datasets demonstrated that, in comparison to other approaches, this technique reduces computational complexity and provides better precision, recall, F-measure, and visual outcomes.

The ability of the variation of prediction footprint (VPF) to identify double compression and determine group of pictures (GOP) sizes in films with bi-directional frames (B-frames) has continuously been improved by researchers [7]. They proved that using motion vectors to improve VPF capture was both efficient and warranted by integrating rate-distortion analysis into a general double compression model. Additionally, they mitigated B-frame-induced shifts in the VPF to solve the problem of imprecise GOP size estimations. In terms of accuracy in double compression detection, GOP size estimation, and processing time reduction, their suggested generalized VPF method fared better than current methods.

In another study, Wu et al. introduced the first end-to-end approach for Copy-Move forgery detection (CMFD) using deep neural networks [8]. Their method begins with feature extraction and then calculates self-correlation among elements from the final layer. This is followed by pointwise feature extraction and decoding operations, which enable pixel-level detection with greater precision than traditional methods. The success of deep learning in CMFD has spurred the development of advanced models like Busternet, which extends earlier work by incorporating a dedicated branch for detecting tampered regions, further

improving Copy-Move forgery detection accuracy.

In order to discover similarities, many CMFD techniques now in use mostly rely on high-level features from the final layer, which is frequently less successful. Semantically independent characteristics from previous layers are usually advantageous for CMFD [9-11]. Zhu et al. developed ARNet, a residual refinement network that makes use of adaptive attention, to get around this restriction [12]. In order to calculate self-correlations and extract similar regions, ARNet combines the final three layers of the feature extraction module, each of which has undergone adaptive attention processing. The last two layers of the feature extraction module were similarly subjected to channel attention by Chen et al. who independently computed self-correlation for each layer before combining the findings for decoding and the identification of related regions [13].

Building on these advancements, Liu et al. proposed a two-stage CMFD method called Proposal SuperGlue [14]. This method calculates self-correlations for the last three layers individually and then fuses the features for the subsequent detection stage. More recently, Zhang et al. introduced a CNN-Transformer model that combines shallow-layer features from CNNs with transformer-based representations [15]. This fusion improves feature representation and enhances the identification of similar regions in forgery detection tasks.

As research interest in video forgery detection has grown [16-20], deep neural networks, including DenseNets [21] and Residual Networks [22], have been widely explored. However, these architectures often struggle with complex forgery types, posing challenges in accurately classifying videos as authentic or forged [17,23]. Traditional approaches, which rely on expert-designed, handcrafted features, have proven insufficient, highlighting the need for more sophisticated, adaptive solutions [24].

Compared to conventional techniques, machine learning has many advantages, especially in the analysis of videos. Machine learning models have the ability to automatically identify patterns and features from unprocessed video data, in contrast to traditional methods that mostly rely on domain-specific expertise and manual feature engineering [25,26]. As a result, less expert involvement is required, and the models may more readily adjust to a variety of datasets. For example, while techniques like SVMs and k-nearest neighbors are frequently employed for classification tasks, algorithms like CNNs are skilled at capturing complex patterns. Richer and more accurate representations are produced by these models' ability to interpret both temporal sequences and spatial relationships found in video material [27].

Furthermore, end-to-end training is made possible by a number of machine learning approaches, which do away with the need for intermediary manual processing stages by mapping input video sequences straight to desired outputs. This increases the effectiveness of video analysis and streamlines procedures. These models are enhanced through the utilization of extensive labeled datasets and techniques such as feature extraction and transfer learning, which allow them to generalize effectively to new data and adapt seamlessly to various video processing tasks. Whether using supervised, unsupervised, or reinforcement learning techniques, machine learning has become a flexible and potent way for analyzing and comprehending videos.

In this study, we present a thorough comparison of two different machine learning methods for identifying Copy-Move forgeries in films. In order to demonstrate each approach's usefulness in forensic analysis, we want to clarify its advantages, disadvantages, and underlying mechanisms. By weighing the benefits and drawbacks of each approach, we aim to provide an unbiased assessment that considers critical factors such as accuracy, computational efficiency, robustness against various types of forgeries such as Copy-Move and splicing, and scalability in practical applications. This analysis, which is based on scientific rigor, aims to produce significant findings that can guide further investigation and enhance forgery detection techniques. Ultimately, this paper aims to contribute to the field by offering a well-rounded perspective on the effectiveness of these techniques, helping researchers select the most appropriate approach based on their specific needs and constraints.

Materials And Methods

The proposed method is divided into two main parts, each of which uses a different machine learning algorithm to identify video frame Copy-Move forgeries. In the first step, an SVM classifier is applied, and the HOG feature extractor is used to extract features from the video frames. HOG is very helpful in identifying the minute alterations that take place during counterfeiting since it efficiently extracts edge, shape, and texture information from the pictures [28]. The SVM identifies the frames as either forged or non-forged after receiving the retrieved features. Because SVM can handle high-dimensional data and determine the best hyperplane to divide the two classes with the greatest margin, it is a good fit for this task [29].

In the second part of the method, a CNN is used to classify the video frames into forged or non-forged. Because CNNs use several convolutional layers to directly learn hierarchical feature representations from raw pixel input, they are effective tools for analyzing images and videos. The CNN can identify intricate

patterns and irregularities in the video frames that can point to manipulation by utilizing spatial hierarchies. Because the CNN automatically learns pertinent features during training, this end-to-end learning method has the advantage of avoiding human feature extraction. By combining CNN for deep learning-based detection and SVM with HOG for feature-based classification, the suggested approach compares and capitalizes on the advantages of both approaches in identifying video forgeries [15].

First, all video clips are transformed from color to grayscale in order to simplify the computation and lower the method's total complexity [27]. Because grayscale frames only have one channel (intensity) as opposed to colored frames' three channels (red, green, and blue), this transition drastically reduces the quantity of data that must be processed. The technique may nonetheless preserve crucial structural and textural information, which is crucial for identifying fabricated regions, by concentrating just on the intensity values and removing extraneous color information. Faster processing times are achieved without compromising the accuracy needed for Copy-Move forgery detection, thanks to this reduction in data dimensionality. Additionally, the grayscale conversion ensures that the method is less sensitive to variations in lighting and color inconsistencies across the video frames, improving the robustness of the detection process. Figure *1* shows the proposed method in full.

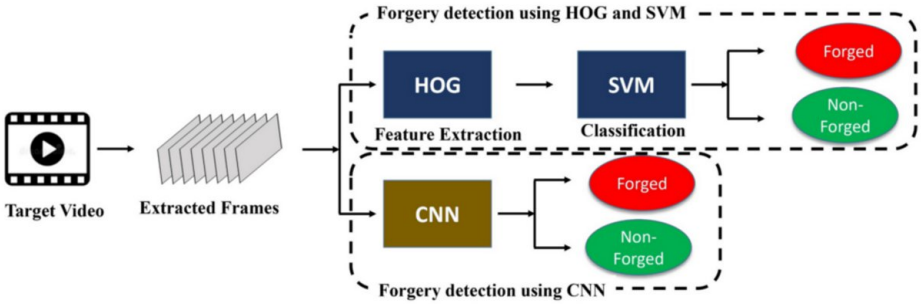

FIGURE 1: Schematic diagram of the proposed method

CNN, Convolutional Neural Network; HOG, Histogram of Oriented Gradients; SVM, Support Vector Machine

HOG and SVM

In computer vision, HOG is a well-known feature extraction method that is very helpful for object identification and detection. By gathering edge and gradient-based information, HOG is essential when applied to video frames to determine whether they are legitimate or counterfeit. This method works particularly well for identifying minute structural alterations in the frame, including anomalies brought about by counterfeit. HOG's efficacy stems from its capacity to depict local object appearances and forms using gradient distributions that are calculated at various orientations [30].

The computation of picture gradients is the first step in the HOG feature extraction process. The gradient values, when applied to a grayscale frame, draw attention to the pixel-level intensity variations that correlate to edges and local texture patterns. Mathematically, for a frame I, the gradients in the horizontal and vertical directions Gx and Gy are typically computed using finite difference methods. These gradients provide essential information about the direction and magnitude of intensity changes in the frame. The magnitude of the gradient is given by

$$M = \sqrt{G_x^2 + G_y^2} \quad (1)$$

and the orientation is computed as

$$\theta = arctan(G_y/G_x) \quad (2)$$

These two quantities encapsulate the local structural properties of the frame, which are crucial for detecting manipulations.

The frame is separated into tiny, non-overlapping spatial areas known as cells after the gradients have been calculated. The gradient orientations are binned into a predetermined number of ranges (bins) in every cell. Generally, the 0° to 180° orientation range is separated into 9 bins, each of which represents a specific range of angles. The contribution of each pixel is weighted by the amplitude of the gradient and goes to the bin that corresponds to its orientation. This captures the local distribution of edge directions in HOG gradients for every cell. Capturing the structural and shape-related information in a confined area

that is especially susceptible to alterations brought about by forgery is the purpose behind this stage.

A normalizing method is used to ensure invariance to variations in light and shadowing. Usually made up of 2 × 2 units, cells are arranged into bigger areas known as blocks. Each block's feature vector is created by concatenating its histograms, and it is subsequently adjusted to lessen the impact of regional contrast fluctuations. L2-norm and L1-norm normalization are popular normalization techniques that guarantee the final feature vector is resistant to changes in illumination and concentrate more on the object's structure than on brightness fluctuations. The dependability of the retrieved characteristics is improved by this normalizing technique, especially in situations where lighting conditions may differ across video frames.

The creation of the HOG descriptor is the last phase in the HOG process. A single feature vector representing the entire frame is created by concatenating the histograms of each block in the frame. Compact and computationally efficient, this descriptor preserves the important texture and shape details of the frame. Because fabricated regions frequently display aberrant edge patterns or anomalies in the local texture, HOG's sensitivity to edge orientations makes it particularly helpful for forgery detection. These descriptors function as highly discriminative features for separating genuine frames from fabricated ones when used with machine learning models like SVMs. The features taken from one of the video frames are displayed in Figure 2.

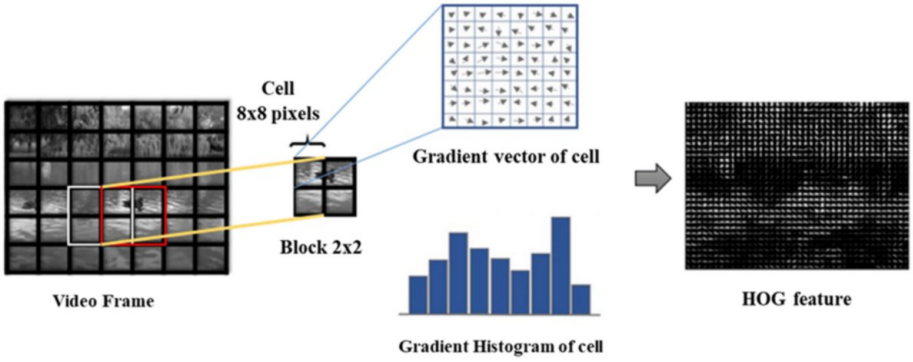

FIGURE 2: HOG feature extractor and its result on one of the video frames

HOG, Histogram of Oriented Gradients

A popular supervised learning technique for classification tasks, such as identifying faked video frames, is the SVM. By establishing a decision boundary that divides the forged and non-forged frames according to the extracted features, SVM becomes an effective tool for detecting minute variations between these two classes when combined with feature extractors such as HOG. To ensure that the classifier performs effectively when applied to unseen data, SVM's main objective is to identify the best hyperplane in the feature space that optimizes the margin between the classes [31].

Following the extraction of the HOG features from every video frame, SVM is used in the context of forgery detection to categorize the frames according to the patterns found in the feature vectors. SVM is tasked with identifying if a frame is forged or not. Each frame is represented as a point in the high-dimensional HOG feature space. By identifying the hyperplane that best divides the two classes of frames, the classifier accomplishes this. For a binary classification problem, such as distinguishing forged from authentic frames, the decision function of the SVM is given by

$$f(x) = w.x + b \quad (3)$$

where:

w is the weight vector that defines the orientation of the hyperplane.

x is the feature vector of the data point being classified. (in this case, the HOG feature vector of a video frame).

b is the bias term that shifts the hyperplane.

SVM's core idea is margin maximization. The distance between the hyperplane and the nearest data points

- also referred to as support vectors - from either class is the margin. SVM seeks to maximize this margin, as a larger margin generally leads to better generalization on unseen data. Mathematically, this is framed as a convex optimization problem, where the classifier aims to $\min \|w\|^2$ subject to the constraint that the data points are correctly classified. For a training set consisting of n labeled examples (x_i, y_i), where x_i is the feature vector and $y_i \in (-1, 1)$ is the label indicating whether the frame is forged or not, the optimization problem can be written as:

$$\min_{w,b} 1/2 \|w\| \; subject \; to \; y_i(w \cdot x + b) \geqslant 1 \forall i \quad (4)$$

This formulation ensures that the classifier correctly separates the data while maximizing the margin. Figure 3 shows the classification process using SVM, and Algorithm 1 presents the detailed steps of the detection process.

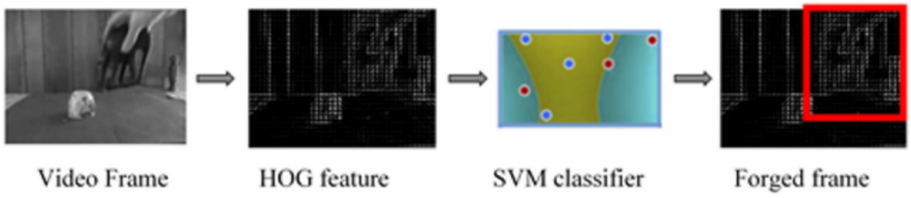

FIGURE 3: Classification process using the combination of HOG and SVM

HOG, Histogram of Oriented Gradients; SVM, Support Vector Machine

In summary, the SVM classifier plays a crucial role by analyzing the feature patterns extracted from video frames. Using the HOG features that highlight important edges and textures, the SVM learns to distinguish between forged and non-forged regions. It identifies patterns typically associated with forgery and classifies new frames based on their similarity to these learned patterns, thereby improving the accuracy of forgery detection.

Algorithm 1:

Input:

A video suspected of forgery.

Step 1: Extract Frames

Break the video into individual frames.

Convert each frame to grayscale to reduce complexity.

Step 2: Extract Features with HOG:

For each frame:

Analyze edges and gradients.

Create a HOG descriptor (a summary of edge directions in small parts of the frame).

Step 3: Train the SVM:

Use labeled data (forged and non-forged frames) to teach the SVM model how to identify forgeries based on the HOG descriptors.

Step 4: Detect Forgery:

For new video frames:

Extract HOG features.

Feed these features into the trained SVM model to predict if they are forged or not.

Step 5: Output Results:

Highlight which frames or parts of the video are likely forged.

CNN implementation

CNNs play an essential role in identifying counterfeit frames, especially those affected by Copy-Move attacks, as they are adept at spotting irregularities and inconsistencies in visual information. CNNs are excellent at identifying particular markers connected to Copy-Move actions because they use hierarchical feature extraction, which allows them to capture intricate features and patterns. When it comes to identifying anomalies in duplicated or shifted portions of a video frame, these networks' profound comprehension of spatial relationships is essential [15,16].

Even when efforts are made to merge altered content smoothly, CNNs can differentiate between genuine and fraudulent regions due to their expertise in pattern recognition. These networks are adaptable to many assault types because they can generalize their insights to handle multiple forging scenarios through training on a variety of datasets. CNNs provide end-to-end solutions for forgery detection systems by identifying and locating Copy-Move assaults by processing complete frames or sequences.

The frames that have been extracted, with dimensions of $w \times h$ pixels, are processed through a sequence of transformations in a CNN that is intended to differentiate between genuine and forged frames. This network comprises three convolutional layers, each followed by a ReLU activation function and a pooling layer. The detailed workings of the network are outlined in Equations 5:9 [32]. In the initial convolutional layer, 3 × 3 filters (kernels) are applied to the input frame.

$$Y[i,j] = (X * W)[i,j] = \sum_{m=0}^{h-1}\sum_{n=0}^{w-1} X[i-m, j-n] \cdot W[m,n] \quad (5)$$

where X represents the input and Y represents the output feature map of the convolutional layer. The indices i and j denote the spatial position of each element in the output feature map. With h and w standing for the height and width of the convolutional filter W, the convolution operation is performed over a specific region of the input.

To ensure stable training, batch normalization is applied to normalize the feature responses. This step is followed by the ReLU activation function. The ReLU function is mathematically expressed as:

$$f(x) = max(0, x) \quad (6)$$

where x is the output feature map from the convolutional layer.

The first pooling layer, which uses max pooling, is then applied with a 2 × 2 pool size and a stride of 2.

$$Y[i,j] = max_{m,n \in Window(i,j)}(X[i \times s + m, j \times s + n]) \quad (7)$$

Here, s stands for the pooling operation's stride. The area in the input feature map X that corresponds to the pooling window at (i,j) in the output feature map is denoted by the window (i,j).

Similar steps are taken by the next convolutional layers, which extract 32 and 64 feature maps, respectively, using 3 × 3 filters. ReLU activation functions and batch normalization are regularly used to improve feature representations. The last step involves flattening the feature maps into a vector and running them through a fully connected layer that contains two classification-related neurons.

$$y_i = \sum_{j=1}^{n} w_{ij} \cdot x_j + b_i \quad (8)$$

where:

y_i: The output of the i-th neuron in the fully connected layer.

w_{ij}: The weight connecting the j-th input neuron to the i-th output neuron.

x_j: The j-th input neuron value.

b_i: The bias for the i-th output neuron.

In the final step, the softmax activation function transforms the raw output scores into probability values, effectively distinguishing between the two classes: forged and non-forged.

$$\sigma(z_k) = e^{z_k} / \sum_{j=1}^{C} e^{z_j} \quad (9)$$

where z_k is the input to the softmax function for class k and $\sigma(z_k)$ is the probability of class k.

The output indicates how probable it is that the input frame belongs to a particular class. The final classification layer uses cross-entropy loss to compute the expected class by selecting the class with the greatest likelihood. Figure 4 illustrates the convolutional network classification procedure, while Algorithm 2 provides a comprehensive description of the technique.

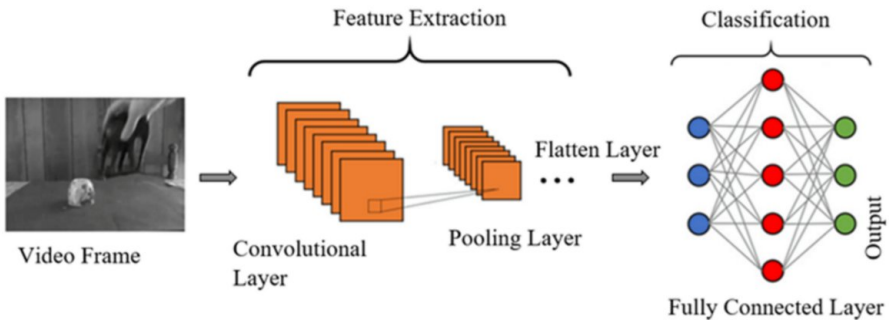

FIGURE 4: Workflow of the classification process using convolutional neural networks

Algorithm 2

Input:

Grayscale video frames (320 × 240 × 1)

Output:

Classification of frames as forged or non-forged

1. Load grayscale video frames.

2. Apply a 3 × 3 convolution with 16 filters (same padding).

3. Perform batch normalization and ReLU activation.

4. Use 2 × 2 max pooling (stride 2).

5. Apply a second 3 × 3 convolution with 32 filters.

6. Repeat batch normalization, ReLU activation, and 2 × 2 max pooling.

7. Apply a third 3 × 3 convolution with 64 filters.

8. Again, apply batch normalization, ReLU activation, and max pooling.

9. Create a vector by flattening the output feature map.

10. Pass the vector through a fully connected layer with 2 outputs.

11. Apply softmax to obtain class probabilities.

12. Classify the frame as forged or non-forged based on the higher probability.

Results

Dataset description

To implement the proposed approach, a widely accepted and standardized dataset is essential. For this study, the REWIND dataset [33] was used. This dataset consists of 20 videos, including 10 authentic videos and 10 that have been manipulated using Copy-Move forgery. Each video is in color, with a resolution of 320 × 240 pixels and a frame rate of 30 frames per second.

Table *1* provides details about the videos used for training, including the total number of frames per video and the number of forged frames in the manipulated versions. Figure *5* showcases examples of both forged and authentic video frames from the dataset.

As the dataset's ground truth, each forged sequence comes with an extra MAT file that details differences in the *Y*, *U*, and *V* components for each frame in the original and created sequences. Examples of manipulated video frames from the dataset are shown in Figures *6* and 7, together with the accompanying ground truth.

Video	Number of frames	Number of forged frames
1	210	210
2	329	48
3	313	82
4	319	63
5	583	190
6	262	56
7	412	128
8	274	33
9	554	211
10	239	65
Total	3,495	1,332

TABLE 1: Video characteristics and frame distribution in the training dataset

FIGURE 5: Examples of frames taken from REWIND dataset

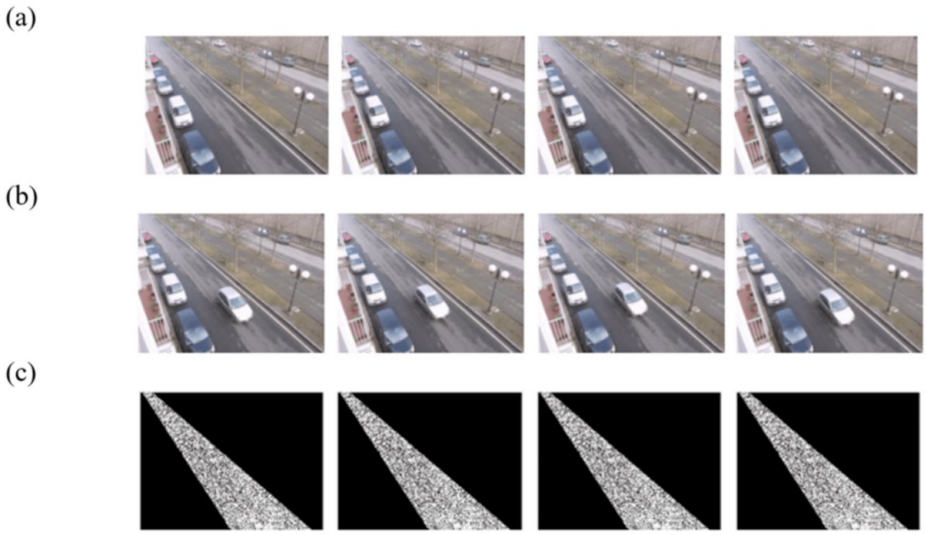

FIGURE 6: (a) Frames from original video in the dataset. (b) Frames from the forged video in the dataset. (c) Ground truth for the video

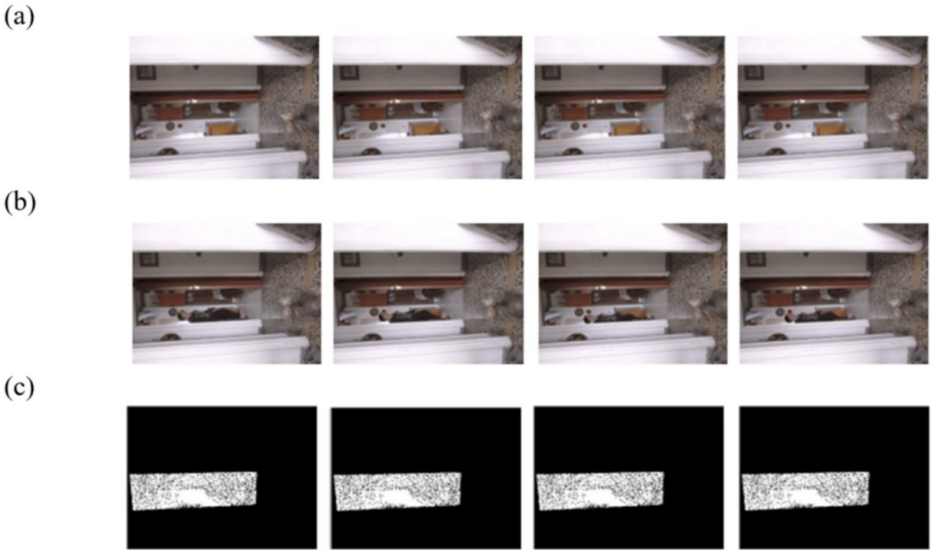

FIGURE 7: (a) Frames from original video in the dataset. (b) Frames from the forged video in the dataset. (c) Ground truth for the video

This study aims to detect counterfeit frames in the REWIND dataset by supplying a CNN with both generated and authentic frames. To improve training efficiency and accuracy, the fake section of each frame is cut off before training. After frames are extracted from the cropped area, the network is given the original dataset segment together with the tampered piece. Figure *8* displays the frames from one of the films in the database that indicate the location of the fake's start.

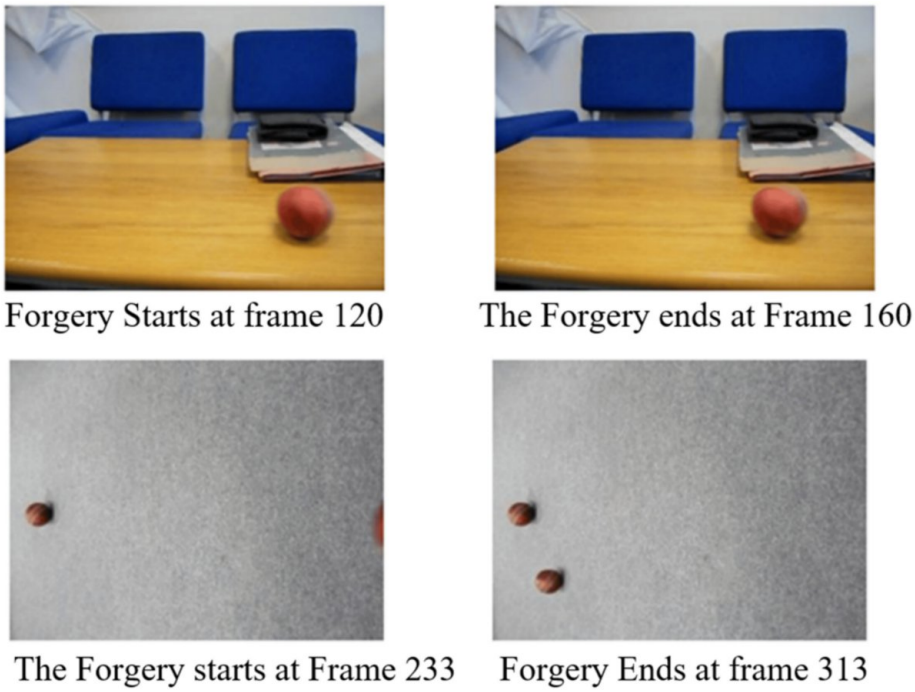

FIGURE 8: The period of forgery in some videos from the dataset

Both authentic and fake video frames were taken out of the previously mentioned dataset and fed into the model during the training phase. These frames were used in the training process. Ninety percent of the data was set aside for training, while the remaining 10% was set aside for testing in order to maximize training efficiency.

The experimental results demonstrate a significant improvement in the performance of both proposed methods compared to previous techniques for detecting forgery in videos. The findings reveal a notable advantage of the deep CNN approach in forgery detection, underscoring the effectiveness of neural networks in this domain. In addition to providing superior accuracy, the CNN method offers reduced computational complexity and minimizes the need for manual feature engineering. Both accuracy and F1-score in equation 10 were used to demonstrate the superiority of CNN over previous methods in general and the proposed method using SVM in particular. This combination of low complexity and high accuracy highlights the CNN approach as a highly efficient solution compared to traditional methods, marking a significant advancement in video forgery detection capabilities. The results are clearly presented in Table *2* and Table *3*, which highlight the comparative performance of both methods, while Figure *9* visually emphasizes the superiority of the CNN-based approach, especially when compared with previous research such as Wang and Farid [11], Su and Li [24], Subramanyam and Emmanuel [25], and Pun et al. [26]. Additionally, Figure *10* illustrates the forged regions detected in the manipulated frames, providing visual confirmation of the effectiveness of the proposed method.

$$F1 - score = 2TP/(2TP + FP + FN) \quad (10)$$

where TP (True Positive) refers to correctly classifying a frame as “forged,” FP (False Positive) refers to incorrectly classifying an “original” frame as “forged,” and FN (False Negative) refers to incorrectly classifying a “forged” frame as “original.”

Algorithm	Wang and Farid [11]	Proposed SVM	Subramanyam and Emmanuel [25]	Pun et al. [26]	Su and Li [24]	Proposed CNN
Detection accuracy	70.0	87.2	89.7	90.8	92.6	93.0

TABLE 2: Detection accuracy on the REWIND dataset

CNN, Convolutional Neural Network; SVM, Support Vector Machine

And F1-score for both methods is as follows in Table 3.

Algorithm	Proposed SVM	Proposed CNN
F1-score	0.65	0.95
Precision	0.61	0.96
Recall	0.7	0.94

TABLE 3: F1-score for proposed methods

CNN, Convolutional Neural Network; SVM, Support Vector Machine

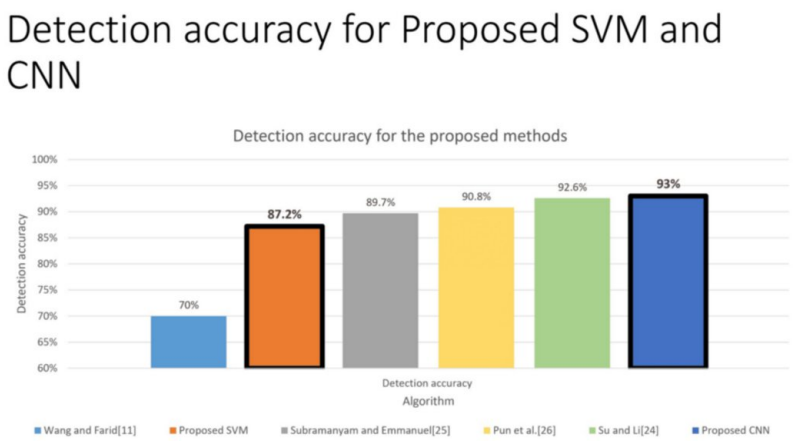

FIGURE 9: Detection accuracy for proposed SVM and CNN

CNN, Convolutional Neural Network; SVM, Support Vector Machine

FIGURE 10: Detecting forged regions in forged video frames

Although it is not the best or most efficient method for video forgery detection, the findings clearly show that the suggested method using SVM performs better than some previous methods, nearing the accuracy of others. On the other hand, the suggested approach that makes use of a CNN is obviously better than the others, attaining greater accuracy in forgery detection when compared to all other approaches. This demonstrates the CNN approach's resilience since it continuously outperforms alternative methods in terms of accuracy, confirming its efficacy for accurate and trustworthy video forgery identification.

Discussion

The performance of two video forgery detection techniques was evaluated using the REWIND dataset. Both techniques were tested on the same dataset to ensure a consistent basis for comparison.

In the first method, features were extracted from grayscale frames using HOG, followed by classification with SVM. This approach yielded an accuracy of 87.2% and an F1-score of 0.65. The lower F1-score indicates limited sensitivity to subtle forgeries, likely due to the handcrafted nature of HOG features and the linear separation capabilities of SVM.

The second method employed a CNN, which was trained directly on the pixel values of the video frames. The network architecture included three convolutional layers with batch normalization, ReLU activations, pooling layers, and a fully connected layer for binary classification. This method achieved an accuracy of 93% and an F1-score of 0.95, outperforming the HOG+SVM combination by a significant margin. The CNN's ability to automatically learn complex spatial patterns contributed to its enhanced performance and generalization across various forgery scenarios.

The notable improvement in both accuracy and F1-score demonstrates the CNN’s effectiveness in identifying manipulated content while maintaining a strong balance between precision and recall. These results confirm that learning-based models like CNN are better suited for video forgery detection tasks compared to traditional feature-based methods.

To further evaluate the robustness of the proposed methods, experiments were conducted on forged videos subjected to common post-processing attacks such as scaling, rotation, contrast adjustment, and brightness changes. The outcomes showed that the CNN-based method was highly resilient to several changes and continued to achieve high detection accuracy even after these changes. Compared to CNN, the SVM-based approach with HOG features worked rather well as well, although it was more obvious how sensitive it was to drastic brightness shifts and high scaling. These results demonstrate that even in difficult post-processing scenarios, the suggested methods in particular, the deep learning approach-are successful in identifying Copy-Move forgeries.

CNNs outperform SVMs in video forgery detection due to their ability to automatically learn spatial features from raw data. Unlike SVMs, which depend on handcrafted features, CNNs capture both low- and high-level patterns through layered processing, making them better at identifying subtle tampering.

CNNs also support end-to-end learning, adapting to various forgery types without manual tuning. This results in higher accuracy, lower false positive/negative rates, and stronger generalization across datasets. Additionally, CNNs benefit from GPU acceleration, allowing faster and more efficient processing-especially valuable for large-scale or real-time forensic applications. These advantages make CNNs a more robust and scalable solution compared to SVMs.

Conclusions

This study conclusively demonstrates the superior performance of CNNs over traditional SVM approaches in detecting video forgeries. Both models were rigorously evaluated using the REWIND dataset. The proposed CNN architecture achieved an accuracy of 93% and an F1-score of 0.95, substantially outperforming the SVM baseline, which recorded an accuracy of 87.2% and an F1-score of 0.65. This

performance gap - 5.8% in accuracy and 0.30 in F1-score - highlights the CNN's capability to autonomously learn hierarchical spatiotemporal features directly from raw pixel data, eliminating the need for handcrafted feature extraction required by SVM-based methods. To facilitate reproducibility and future research, the proposed system was implemented using a three-layer CNN architecture with ReLU activations, max-pooling layers, and a final fully connected layer with softmax activation for classification. All input frames were preprocessed into grayscale and resized to 320 × 240 pixels before being fed into the model. The SVM model was trained using HOG features extracted from the same frame set.

Future work should focus on improving the robustness and precision of forgery detection, particularly under challenging conditions such as noise-enhanced forgeries, scaling, rotation, contrast adjustment, and other post-processing attacks. One promising direction is the development of hybrid architectures - such as combining CNNs with attention mechanisms or transformers - which may enhance the model's ability to localize and detect subtle forgeries. Furthermore, incorporating unsupervised and semi-supervised learning approaches could reduce reliance on large annotated datasets. For real-world applicability, future research should also aim to optimize these models for real-time performance and deployment on resource-constrained platforms like mobile forensic tools, social media content verification systems, and surveillance applications.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Noha M. El-Badry, Reda Elbarougy, Gamal M. Behery

Drafting of the manuscript: Noha M. El-Badry, Reda Elbarougy, Gamal M. Behery, Osama Abdelfatah

Critical review of the manuscript for important intellectual content: Noha M. El-Badry, Reda Elbarougy, Gamal M. Behery

Supervision: Noha M. El-Badry, Reda Elbarougy, Gamal M. Behery

Acquisition, analysis, or interpretation of data: Osama Abdelfatah

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

References

1. Laura B, Christophe N, William PA: A very fast copy-move forgery detection method for 4K ultra HD images. Frontiers in Signal Processing. 2022, 2:906304. 10.3389/frsip.2022.906304
2. Zhong JL, Gan YF, Vong CM, Yang JX, Zhao JH, Luo JH: Effective and efficient pixel-level detection for diverse video copy-move forgery types. Pattern Recognition. 2022, 122:108286. 10.1016/j.patcog.2021.108286
3. Pandey RC, Singh SK, Shukla KK: Passive copy-move forgery detection in videos. 2014 International Conference on Computer and Communication Technology (ICCCT), Allahabad, India. 2014, 301-306. 10.1109/ICCCT.2014.7001509
4. Su L, Li C, Lai Y, Yang J: A fast forgery detection algorithm based on exponential-Fourier moments for video region duplication. IEEE Transactions on Multimedia. 2018, 20:825-840. 10.1109/tmm.2017.2760098
5. Singh RD, Aggarwal N: Detection of upscale-crop and splicing for digital video authentication. Digital Investigation. 2017, 21:31-52. 10.1016/j.diin.2017.01.001
6. Fatima B, Ghafoor A, Ali SS, Riaz MM: FAST, BRIEF and SIFT based image copy-move forgery detection technique. Multimedia Tools and Applications. 2022, 81:43805-43819. 10.1007/s11042-022-12915-y
7. Vázquez-Padín D, Fontani M, Shullani D, Pérez-González F, Piva A, Barni M: Video integrity verification and GOP size estimation via generalized variation of prediction footprint. IEEE Transactions on Information Forensics and Security. 2020, 15:1815-1830. 10.1109/TIFS.2019.2951313
8. Wu Y, Abd-Almageed W, Natarajan P: Image copy-move forgery detection via an end-to-end deep neural network. 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA. 2018, 1907-1915. 10.1109/WACV.2018.00211
9. Ou X, Wang H, Li W, Zhang G, Chen S: A scene segmentation algorithm combining the body and the edge of

the object. Information Processing & Management. 2022, 59:102840. 10.1016/j.ipm.2021.102840
10. Zhou S, Sun T, Xia X, Zhang N, Huang B, Xian G, Chai X: Library on-shelf book segmentation and recognition based on deep visual features. Information Processing & Management. 2022, 59:103101. 10.1016/j.ipm.2022.103101
11. Wang W, Farid H: Exposing digital forgeries in video by detecting duplication. MM&Sec '07: Proceedings of the 9th Workshop on Multimedia & Security. 2007, 35-42. 10.1145/1288869.1288876
12. Zhu Y, Chen C, Yan G, Guo Y, Dong Y: AR-Net: Adaptive attention and residual refinement network for copy-move forgery detection. IEEE Transactions on Industrial Informatics. 2020, 16:6714-6723. 10.1109/tii.2020.2982705
13. Chen B, Tan W, Coatrieux G, Zheng Y, Shi YQ: A serial image copy move forgery localization scheme with source/target distinguishment. IEEE Transactions on Multimedia. 2021, 23:3506-3517. 10.1109/tmm.2020.3026868
14. Liu Y, Xia C, Zhu X, Xu S: Two-stage copy-move forgery detection with self deep matching and proposal SuperGlue. IEEE Transactions on Image Processing. 2022, 31:541-555. 10.1109/TIP.2021.3132828
15. Zhang Y, Zhu G, Wang X, Luo X, Zhou Y, Zhan H: CNN-transformer based generative adversarial network for copy-move source/target distinguishment. IEEE Transactions on Circuits and Systems for Video Technology. 2023, 33:2019-2032. 10.1109/TCSVT.2022.3220630
16. Koshy L, Ajay S, Paul A, Hariharan V, Basheer A: Video forgery detection using CNN. 2021 Smart Technologies, Communication and Robotics (STCR), Sathyamangalam, India. 2021, 1-6. 10.1109/STCR51658.2021.9588860
17. Rodriguez-Ortega Y, Ballesteros DM, Renza D: Copy-move forgery detection (CMFD) using deep learning for image and video forensics. Journal of Imaging. 2021, 7:59. 10.3390/jimaging7030059
18. Aria M, Hashemzadeh M, Farajzadeh N: QDL-CMFD: A quality-independent and deep learning-based copy-move image forgery detection method. Neurocomputing. 2022, 511:213-236. 10.1016/j.neucom.2022.09.017
19. Munawar M, Noreen I, Alharthi RS, Sarwar N: Forged video detection using deep learning: A SLR. Applied Computational Intelligence and Soft Computing. 2023, 6661192:1-21. 10.1155/2023/6661192
20. Raskar PS, Shah SK: Real time object-based video forgery detection using YOLO (V2). Forensic Science International. 2012, 327:110979. 10.1016/j.forsciint.2021.110979
21. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ: Densely connected convolutional networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA. 2017, 2261-2269. 10.1109/CVPR.2017.243
22. Vaishali S, Neetu S: Enhanced copy-move forgery detection using deep convolutional neural network (DCNN) employing the ResNet-101 transfer learning model. Multimedia Tools and Applications. 2023, 83:10839-10863. 10.1007/s11042-023-15724-z
23. Al Azrak F, Sedik A, Dessowky M, et al.: An efficient method for image forgery detection based on trigonometric transforms and deep learning. Multimedia Tools and Applications. 2020, 79:18221-18243. 10.1007/s11042-019-08162-3
24. Su L, Li C: A novel passive forgery detection algorithm for video region duplication. Multidimensional Systems and Signal Processing. 2018, 29:1173-1190. 10.1007/s11045-017-0496-6
25. Subramanyam AV, Emmanuel S: Video forgery detection using HOG features and compression properties. 2012 IEEE 14th International Workshop on Multimedia Signal Processing (MMSP), Banff, AB, Canada. 2012, 89-94. 10.1109/MMSP.2012.6343421
26. Pun C-M, Yuan X-C, Bi X-L: Image forgery detection using adaptive oversegmentation and feature point matching. IEEE Transactions on Information Forensics and Security. 2015, 10:1705-1716. 10.1109/tifs.2015.2423261
27. Elbarougy R, Abdelfatah O, Behery GM, El-Badry NM: Deep learning-based method for detection of copy-move forgery in videos. Neural Computing and Applications. 2025, 37: 8451-8464. 10.1007/s00521-025-11009-8
28. Kaur M, Walia S: Forgery detection using noise estimation and HOG feature extraction. International Journal of Multimedia and Ubiquitous Engineering. 2016, 11:37-48. 10.14257/ijmue.2016.11.4.05
29. Mustafa Abdullah D, Mohsin Abdulazeez A: Machine learning applications based on SVM classification: A review. Qubahan Academic Journal. 2021, 1:81-90. 10.48161/qaj.v1n2a50
30. Isaac MM, Wilscy M: A key point based copy-move forgery detection using HOG features. 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), Nagercoil, India. 2016, 1-6. 10.1109/ICCPCT.2016.7530253
31. Ryu SJ, Lee HY, Cho IW, Lee HK: Document forgery detection with SVM classifier and image quality measures. Advances in Multimedia Information Processing - PCM 2008. Huang YMR, Xu C, Cheng KS, Yang JFK, Swamy MNS, Li S, Ding JW (ed): Springer, Berlin, Heidelberg; 2008. 5353:486-495. 10.1007/978-3-540-89796-5_50
32. Kim P: MATLAB Deep Learning. Apress, Berkeley, CA; 2017. 10.1007/978-1-4842-2845-6
33. https://sites.google.com/site/rewindpolimi/downloads/datasets/video-copy-move.