International Journal of Computers and Informatics

Journal Homepage: https://www.ijci.zu.edu.eg

Int. j. Comp. Info. Vol. 10 (2026) 139–151

Paper Type: Original Article

Video Duplication Forgery Detection Using EfficientNetB0 with Motion-Based Verification

Mona M. Ali [1,*], Hanaa M. Hamza [2,3], Neveen I. Ghali [4], and Khalid M. Hosny [2]

[1] Department of Digital Media Technology, Faculty of Computers and Information, Future University in Egypt (FUE), New Cairo, Egypt; mona.almakhton@fue.edu.eg.

[2] Department of Information Technology, Faculty of Computers and Information, Zagazig University, Zagazig 44519, Egypt; k_hosny@zu.edu.eg;

[3] Artificial Intelligence and Data Science Program, Engineering Sector, Zagazig National University, Egypt; hmkamal@fci.zu.edu.eg;

[4] Department of Information Technology, Faculty of Computers and Artificial Intelligence, Azhar University, Cairo, Egypt; Neveen.ghali@azhar.edu.eg.

Received: 16 Jan 2026 **Revised**: 16 Feb 2026 **Accepted**: 08 Mar 2026 **Published**: 10 Mar 2026

Abstract

The authenticity of digital videos has become a critical concern due to the widespread availability of advanced editing tools that enable subtle manipulations. Frame duplication is a common form of video tampering in which frames are copied and reinserted at different temporal positions to hide or alter events. Detecting such manipulations is challenging. This paper introduces a dual-stage detection framework that addresses this challenge by combining deep feature representations with motion-consistency analysis. The proposed method first employs a lightweight EfficientNetB0 model to extract discriminative features from video frames. A temporal-constrained cosine similarity module then identifies potential duplicate candidates by comparing features only beyond a minimum frame gap and reducing false positives from adjacent frames. In the second stage, a dense optical flow verification module analyzes motion patterns between candidate pairs, confirming duplications only when high visual similarity is accompanied by negligible inter-frame motion. The framework is rigorously evaluated on multiple benchmark datasets, including TDTVD, Fadl, and SULFA. Experimental results demonstrate that the method achieves state-of-the-art performance, attaining an average accuracy of 99.82% and a recall of 100% on the TDTVD dataset. Comparative analysis shows consistent superiority over existing techniques in both detection accuracy and operational efficiency. The architecture ensures computational practicality by limiting resource-intensive optical flow computation to a small subset of high-similarity frames. This work offers significant improvements in reliably identifying duplication forgeries while maintaining feasibility for real-world applications.

Keywords: Video Forgery; Frames Duplication; EfficientNetB0; Deep Learning; Motion Analysis.

1 | Introduction

The widespread availability of advanced digital editing tools and generative artificial intelligence has turned video manipulation into a significant threat that undermines the integrity of visual media [1]. This includes the creation of deepfakes intended to spread disinformation and the subtle alteration of surveillance footage.

Corresponding Author: mona.almakhton@fue.edu.eg

Such video forgeries present serious challenges to security, journalism, and the verification of legal evidence [1].

One common technique of tampering is frame duplication, in which a sequence of frames is copied and reinserted into the same video to conceal or repeat certain events. Detecting these forgeries is crucial for authenticating digital video content; however, it remains a complex task. This complexity arises from the need to differentiate malicious duplication from natural scene repetition, such as static backgrounds or periodic motion[2].

Traditional video forensic methods relied on handcrafted features and block-matching algorithms, which exhibited limited robustness to post-processing operations and struggled with temporal consistency modeling [3]. Deep learning has revolutionized the field, enabling comprehensive learning of spatio-temporal features specific to various forgeries and compression artifacts [4, 5]. Recent years have seen rapid advancement in CNN-based architectures, Siamese similarity learning frameworks, and multi-modal fusion strategies that combine appearance and motion cues. There are still significant gaps that need to be addressed, such as many methods rely on computationally intensive architectures that are not suitable for processing large-scale video datasets, existing benchmark datasets for frame duplication detection are limited in both scale and diversity, and few approaches incorporate motion verification, which is necessary to reduce false positives caused by similarities in natural scenes [6]. This work aims to overcome these limitations by providing a comprehensive literature review and proposing a novel dual-stage detection framework. This work addresses these limitations through a comprehensive literature synthesis and a novel dual-stage detection framework

The contributions of this work can be summarized as follows:

- We introduce a detection pipeline that combines deep feature similarity with optical flow verification, effectively separating malicious duplications from natural video redundancies.
- We employ a temporal constraint in the similarity matching phase, which enhances precision and dramatically lowers computational overhead compared to exhaustive pairwise frame comparison.
- We demonstrate that the EfficientNetB0 architecture provides an optimal trade-off between representational power and computational load for feature-based forensic analysis of video sequences.
- We conduct an extensive evaluation on multiple benchmark datasets, including TDTVD, Fadl, and SULFA, showing that our method achieves state-of-the-art performance, surpassing existing techniques in both detection accuracy and operational practicality.

This paper introduces a methodology for identifying duplications by integrating deep feature extraction with advanced similarity analysis and motion verification, addressing the limitations of existing approaches. The proposed method aims to enhance detection accuracy and reduce false positives by incorporating a multi-stage verification process that extends beyond simple feature matching to analyze temporal consistency and motion characteristics inherent in authentic video sequences.

The remainder of this paper is structured as follows: Section 2 reviews related work in video forgery detection. Section 3 details the architecture and components of the proposed framework. Section 4 presents the experimental setup, datasets, and a comprehensive analysis of results, including comparisons with contemporary methods. Finally, Section 5 concludes the paper and suggests directions for future research.

2 | Literature Review

Recent studies have identified three dominant architectural families for video frame duplication detection: CNN-based spatiotemporal models, Siamese similarity-learning frameworks, and dual-stage verification systems. This section provides a detailed methodological analysis of representative works from each family, highlighting their strengths and limitations in relation to the goals of our work.

CNNs play a central role in video forensics, primarily due to their effectiveness in extracting spatial features and their adaptability for temporal analysis. A prominent approach involves fusing spatial and temporal information. For instance, Fadl et al. [7] combined 2D CNNs for spatial artifact detection with 3D CNNs (C3D) for capturing short-term temporal inconsistencies, creating an efficient system effective for forgeries like frame insertion and deletion in surveillance footage. Expanding on this, Akhtar et al. [3] developed the DEEP-STA framework, which integrates CNN-based spatial feature extraction with dedicated modules for temporal consistency analysis and precise localization of various inter-frame tampering types. While 3D CNNs themselves offer a natural architecture for processing video volumes and detecting subtle temporal anomalies, they typically demand greater computational resources and larger datasets.

CNN-based methods are valued for their computational efficiency and strong performance in identifying local, frame-level tampering artifacts. Their primary limitation lies in the constrained temporal receptive field of convolutional kernels, which can hinder the detection of long-range manipulations. 3D CNNs are well-suited for video analysis but require significant computational resources and large datasets. This makes them impractical for real-world forensic applications where efficiency is crucial. To address this, we propose a method that uses a lightweight 2D CNN for spatial feature extraction. This is combined with a separate, efficient motion verification stage to capture temporal inconsistencies, avoiding the high computational cost of 3D convolutions.

Siamese networks with shared weights model frame similarity for duplicate detection and anomaly localization. Munawar and Noreen's I3D-Siamese RNN uses I3D features, Euclidean distance, and RNNs, achieving 86.6% and 93% accuracy [8]. This interpretable method requires careful parameter tuning, limiting generalization. Siamese frameworks, optimized for matching, depend on hyperparameters and sampling, needing dataset-specific tuning [8]. Separating natural repetition from malicious duplication may require motion verification. Siamese frameworks offer a strong foundation for duplication detection. However, their effectiveness rely on careful hyperparameter optimization and sampling strategy selection, frequently necessitating dataset-specific adjustments [8]. Furthermore, relying solely on learned similarity to differentiate between genuine repetition and malicious duplication may prove insufficient. Explicit motion verification is crucial, as visually similar frames can stem from static scenes or cyclical movements. The verification process is directly integrated in our methodology, decreasing the reliance on the similarity model to understand these complex scenarios.

Dual-stage frameworks that combine appearance-based screening with motion-based verification are effective at reducing false positives in video forgery detection while maintaining high accuracy. In these architectures, an initial coarse stage quickly identifies candidate segments using texture-based features such as Motion Residual LBP (MR-LBP) [6]. The verification stage applies optical flow analysis to assess motion consistency, thereby filtering out false alarms caused by legitimate scene repetition. Optical flow residuals and spatiotemporal consistency effectively detect insertions, deletions, and duplications, but increase pipeline complexity and require dataset-specific parameter tuning. Inspired by these effective two-stage approaches, our framework uses EfficientNetB0 for efficient feature extraction. It employs temporally-constrained similarity search to reduce candidate set size and applies dense optical flow verification to this smaller set, improving computational efficiency compared to prior dual-stage methods relying on handcrafted features or heavier backbones.

The research conducts a comprehensive examination of the scaling of convolutional neural networks by analyzing the implications of augmenting network depth, breadth, and input resolution for both accuracy and computational efficiency. The authors [9] present a compound scaling strategy that uniformly adjusts a model's depth, width, and resolution using a single compound coefficient while maintaining a balanced structure. A new baseline architecture, EfficientNet-B0, was developed via neural architecture search to optimize accuracy and computational cost. Efficient Net models (B1-B7) were created by scaling a basic network across multiple dimensions. This family achieves higher accuracy than existing convolutional networks while requiring significantly fewer parameters and floating-point operations. EfficientNet models

demonstrate strong transfer-learning performance across various vision datasets, often achieving new state-of-the-art results with considerably reduced model sizes.

Methods that utilize optical flow residuals and spatiotemporal consistency checks have shown strong performance in detecting insertions, deletions, and duplications. While this design significantly enhances localization precision, it also introduces greater complexity to the pipeline and often requires careful tuning of dataset-specific parameters. The demonstrated effectiveness of this two-stage approach motivates the proposed dual-stage framework, which employs EfficientNetB0 and dense optical flow.

3 | The Proposed Method

We propose a video duplication forgery detection framework combining EfficientNetB0-based deep feature extraction with cosine similarity and dense optical flow verification. Our comprehensive framework is constructed around three principal components: (1) a deep feature extraction module that employs the EfficientNetB0 architecture to systematically analyze and extract relevant features from the input data, (2) a highly efficient similarity-based duplicate detection module that incorporates temporal constraints to enhance the precision and reliability of the detection process over time, and (3) an advanced optical flow verification module specifically designed to facilitate thorough motion analysis by tracking the movement of objects across successive frames. The overall architecture is represented in Figure 1.

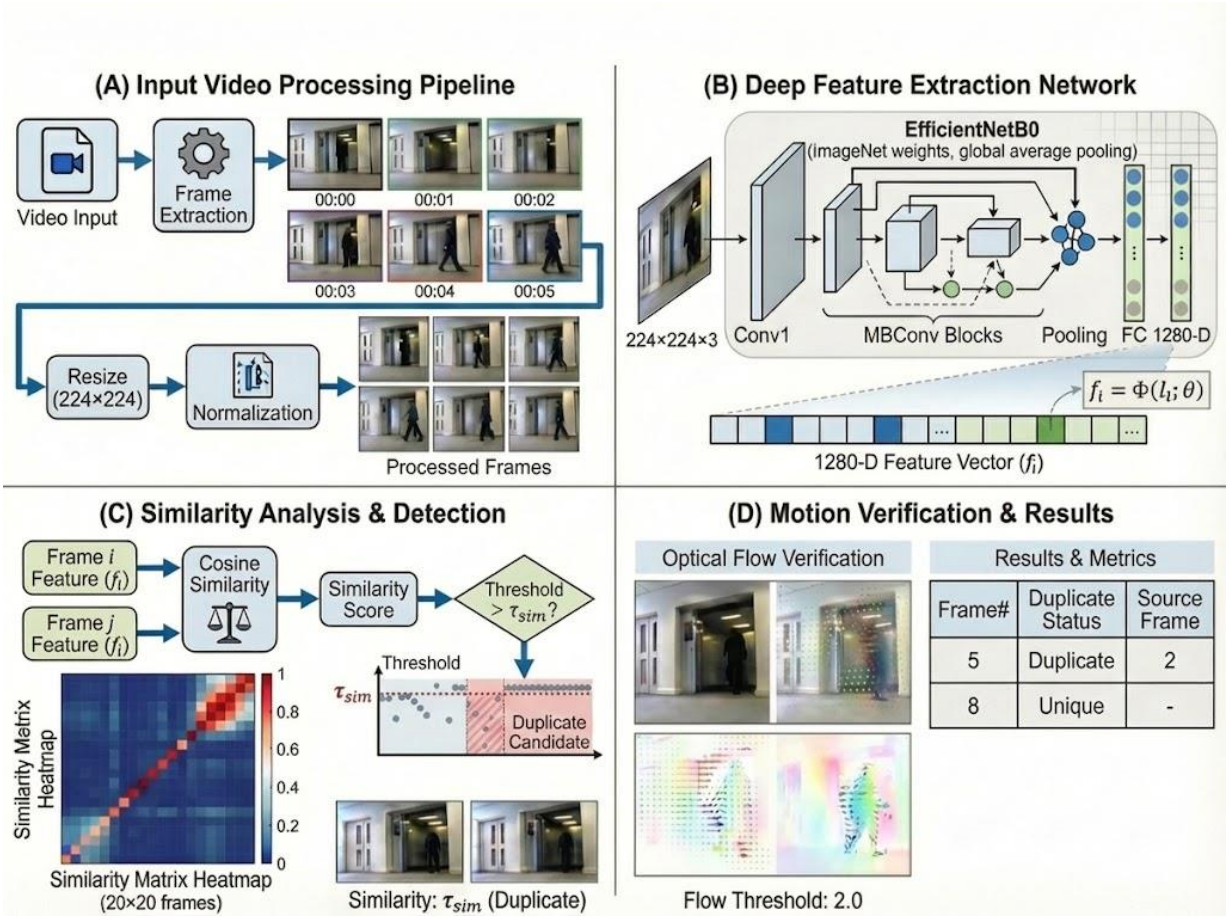

Figure 1. The proposed framework for detecting duplicate frames in video: (A) Input video processing pipeline extracts frames for analysis. (B) Deep feature extraction using EfficientNetB0 generates discriminative representations. (C) Similarity analysis identifies potential duplicates using cosine distance. (D) Optical flow verification refines detection by analyzing inter-frame motion, with final classification results

3.1 | Preprocessing

Input videos are decomposed into frames using the OpenCV library, with frame indices rigorously recorded to preserve temporal order, which is crucial for enforcing temporal gap constraints and supporting subsequent motion analysis. Each extracted frame is preprocessed to be aligned to the input specifications of EfficientNetB0. Frames are resized to 224 × 224 pixels using bilinear interpolation, in accordance with the ImageNet training configuration for EfficientNetB0 [9]. The color representation is preserved in the RGB color space with three channels, and pixel intensities are normalized to the [0, 1] interval prior to standardization utilizing the ImageNet mean and standard deviation, where μ = [0.485, 0.456, 0.406] and σ = [0.229, 0.224, 0.225] as shown in Figure 1A. This standardized preprocessing pipeline ensures compatibility with the pretrained EfficientNetB0 weights and enhances transfer learning by aligning the input data distribution with that of the original training dataset [1, 6]. The selected resolution balances the preservation of spatial detail with computational efficiency.

3.2 | Feature Extraction

EfficientNetB0 is adopted as the feature extraction backbone due to its balanced computational cost and accuracy, as it contains 5.3 million parameters and requires only 0.39 billion FLOPs per image, offering one of the strongest accuracy-to-computation ratios among contemporary convolutional architectures[9]. Its compound scaling strategy, which jointly adjusts depth, width, and input resolution, along with the inclusion of squeeze and excitation blocks, enhances its ability to capture fine-grained texture irregularities that often signal visual manipulation [10]. Previous research has shown that larger variants within the same family, such as EfficientNetB4 through B7, achieve notable performance in detecting video forgeries and facial manipulations, reinforcing the relevance of this architecture to forensic analysis tasks. The compact design of EfficientNetB0 supports efficient processing of extended video sequences and large datasets while maintaining feasible memory and time requirements[9].

3.3 | Similarity

Figure 1C shows that for each incoming frame F_i Cosine similarity is computed between its feature vector f_i and the features of previously selected key frames stored in a dictionary D Cosine similarity is defined as

$$\text{sim}(f_i, f_j) = \frac{f_i \cdot f_j}{\|f_i\| \, \|f_j\|} \qquad (1)$$

where $f_j \in D$, and measures angular similarity between feature representations. To reduce false detections caused by natural temporal continuity, only frame pairs separated by a minimum temporal gap $\tau_{\text{gap}}$are compared, such that $| i - j | \geq \tau_{\text{gap}}$. The value of $\tau_{\text{gap}}$is selected based on the video frame rate. A frame is considered a duplicate candidate if its maximum similarity with any eligible key frame exceeds a threshold τ_{sim} and selected in the range [0.90, 0.98] based on empirical evaluation. If no such match is found, the frame is added to the dictionary D, enabling incremental representation of unique scene content. This strategy reduces computational cost relative to exhaustive pairwise comparisons while enabling the detection of duplicate frames across temporal locations.

The same similarity threshold $\tau_{\text{sim}} = 0.95$ is applied uniformly across all datasets after optimization on the validation set. This approach is justified by the robustness and generalizability of the EfficientNetB0 feature space, and empirical validation across all three datasets confirms consistent performance with this fixed threshold.

3.4 | Motion-Based Verification

High visual similarity between video frames may arise either from intentional duplication or from natural scene repetition, such as static surveillance views or periodic motion. Motion analysis is therefore introduced

to distinguish between these cases, since duplicated frames typically exhibit negligible motion, whereas naturally similar frames usually contain measurable motion caused by camera movement, object dynamics, or noise[11].

For each frame pair (F_i, F_j)identified as a candidate in the first stage, dense optical flow is computed using the Farnebäck algorithm[12]. This process estimates a dense motion field $\Phi_{ij} \in \mathbb{R}^{H\times W\times 2}$, where each vector $[u(x,y), v(x,y)]$represents the horizontal and vertical displacement at pixel location (x, y). Standard Farnebäck parameters are adopted to ensure stable motion estimation, including a three-level image pyramid, a 15 × 15 pixel window size, three iterations per pyramid level, and a 5 × 5 pixel neighborhood for polynomial approximation [12].

Motion consistency between frames is quantified using the average optical flow magnitude, computed as the mean pixel displacement over the entire frame. A candidate pair is confirmed as a duplication if the average motion magnitude falls below a predefined threshold, tau_flow. Typical values of τ_{flow} range between 1.0 to 3.0 pixels, corresponding to minimal motion levels that can be attributed to sensor noise or compression artifacts rather than genuine scene dynamics. The computational cost of this stage is proportional to the number of candidate pairs and the frame resolution. By restricting optical flow computation to similarity-flagged candidates, the method avoids the quadratic cost associated with exhaustive motion analysis across all frame pairs [11].

The average optical flow magnitude was selected as the verification metric for its computational efficiency and discriminative power in duplicate frame detection. In binary classification, duplicated frames exhibit near-zero motion which marked by sensor noise and compression artifacts and naturally similar frames show measurable motion. The similarity stage generates a reduced candidate set, modeled as a function $(Fi)\rightarrow Cij$. The motion verification stage then refines these candidates using a conditional operator applied to Cij. Error propagation is addressed, noting that Stage 1 false negatives are irrecoverable, while Stage 1 false positives are filtered in Stage 2. This formalization emphasizes high recall in Stage 1 and high precision refinement in Stage 2.The pseudo-code in Figure 2 shows a two-stage method based on motion constraints in natural videos that combines high appearance similarity with low motion magnitude. The algorithm reliably identifies malicious duplication and reduces false positives from natural scene repetition.

```
Procedure MAIN()
    Initialize EfficientNetB0 model with pre-trained ImageNet weights (top excluded, pooling='avg')
    Configure GPU memory for efficient processing (set memory growth)
    Set parameters: VIDEO_FOLDER, GROUND_TRUTH_PATH, OUTPUT_FOLDER, MIN_GAP, SIM_THRESHOLD
    Create output directories if not exist
    For each video file in VIDEO_FOLDER:
        Results ← PROCESS_VIDEO(video_path, MIN_GAP, SIM_THRESHOLD)
        Save Results to Excel sheet
        If GROUND_TRUTH_PATH exists:
            Metrics ← EVALUATE_RESULTS(Results, GROUND_TRUTH_PATH)
            Save Metrics to Excel sheet
        End If
    End For
End Procedure
Function EXTRACT_FEATURES(frame)
    Resize frame to 224x224 and apply EfficientNet preprocessing (preprocess_input)
    Perform forward pass through EfficientNetB0 (without top layers, using average pooling)
    Return flattened feature vector
End Function
Function COSINE_SIMILARITY(feat1, feat2)
    Compute cosine distance and return similarity = 1-cosine(feat1, feat2)
End Function
Function OPTICAL_FLOW_MAGNITUDE(frame_a, frame_b)
    Convert frames to grayscale
    Compute Farneback optical flow between the two frames
    Return mean magnitude of flow vectors
End Function
Procedure PROCESS_VIDEO(video_path, min_gap, sim_threshold)
    Extract all frames from the input video using OpenCV
    Let frame_count = number of frames
    Initialize an array frame_features of size frame_count
    For i = 0 to frame_count-1:
        frame_features[i] ← EXTRACT_FEATURES(frame_i)
    End For
    Initialize key_frames as empty dictionary   // stores {index: feature_vector}
    duplicates_found ← 0
    For i = 0 to frame_count-1:
        current_feat ← frame_features[i]
        is_duplicate ← False
        For each (idx, key_feat) in key_frames:
            If |i-idx| < min_gap Then
                Continue   // frames too close, skip comparison
            End If
            similarity ← COSINE_SIMILARITY(current_feat, key_feat)
            If similarity > sim_threshold Then
                If |i-idx| < CLOSE_FRAME_WINDOW (e.g., 100) Then
                    motion ← OPTICAL_FLOW_MAGNITUDE(frames[idx], frames[i])
                    If motion < MOTION_THRESHOLD (e.g., 2.0) Then
                        Mark frame i as duplicate of frame idx
                        duplicates_found++
                        is_duplicate ← True
                        Break
                    End If
                Else   // frames are far apart
                    Mark frame i as duplicate of frame idx
                    duplicates_found++
                    is_duplicate ← True
                    Break
                End If
            End If
        End For
        If not is_duplicate Then
            Add i to key_frames with its feature vector
        End If
    End For
    Return duplicate annotations (DataFrame) and total processing time
End Procedure
```

Figure 2. Pseudo-code of the Proposed Dual-Stage Video Duplication Detection Method.

4 | Experiments and Results

The proposed method was evaluated on Google Colab using a Python 3 environment with GPU acceleration, with 15.0 GB of GPU memory, 12.7 GB of RAM, and 112 GB of local storage. The implementation was developed using TensorFlow Keras with TensorFlow version 2.19.0 and supported by standard scientific libraries, including NumPy version 2.0.2. This setup enabled efficient dataset processing, reliable performance evaluation against ground truth annotations, and assessment of scalability and real-world applicability.

4.1 | Performance parameters

The performance of the proposed video forgery detection method was evaluated using ground truth annotations and standard classification metrics. Precision, recall, F1-score, and detection accuracy were calculated from true positives, false positives, true negatives, and false negatives using equations (2) to (5). Precision measures the reliability of detected forgeries; recall reflects the ability to identify duplicated frames; the F1-score balances these two metrics; and detection accuracy indicates overall classification correctness.

$$P = \frac{TP}{TP+FP} \quad (2)$$

$$R = \frac{TP}{TP+FN} \quad (3)$$

$$F1 = 2x\frac{P\ x\ R}{P+R} \quad (4)$$

$$DA = \frac{TP+FP}{TP+FP+FN+TN} \quad (5)$$

Evaluation was performed through video level comparison with manually annotated data. The results demonstrate high precision, recall, and F1-score, confirming the effectiveness and robustness of the proposed method in detecting duplicate content, with fewer false alarms than baseline approaches.

4.2 | Datasets

The development of the Temporal Domain Tampered Video Dataset (TDTVD)[13], which aims to address the lack of publicly available datasets for video tampering detection. It comprises 210 videos, categorized into Event/Object/Person (EOP) tampering and Smart Tampering (ST). The EOP category consists of 120 videos, with 40 videos for each tampering type: frame deletion, frame insertion, and frame duplication. The remaining 90 videos in the ST category employ multiple tampering techniques applied across multiple locations within the same video. The dataset is derived from 16 original videos from SULFA[14] and 24 from the YouTube VTD dataset[15], ensuring a diverse range of activities and scenarios for researchers to test their algorithms.

Detailed ground-truth information is provided for each tampered video, including the type of tampering, the number of frames affected, and their locations, which is crucial for verifying tampering-detection algorithms. The dataset has been validated against two video tampering detection methods, demonstrating its effectiveness in supporting research in video authentication. The TDTVD dataset is publicly accessible, providing a valuable resource for researchers working on video tampering detection and algorithm development [13].

Fadl et al.[16] introduced a benchmark dataset for inter-frame forgery detection designed to evaluate the performance of detection algorithms under realistic conditions. The dataset is constructed from videos collected from SULFA [14], LASIESTA[17], and IVY LAB [18] covering multiple resolutions and frame rates commonly encountered in surveillance footage. It includes four common types of video tampering: frame deletion, frame insertion, frame duplication, and frame shuffling, which frequently occur in scenarios with stationary scenes and fixed cameras. The dataset comprises 60 videos, including 28 original and 32 forged sequences, with durations ranging from 10 to 27 seconds. By incorporating diverse types of video forgery that alter semantic content, this dataset provides a valuable and realistic benchmark for developing and evaluating inter-frame forgery detection methods in multimedia forensics.

4.3 | Results

This section presents a comprehensive evaluation of the proposed video duplication detection framework across two benchmark datasets, TDTVD [13] . Results are compared with state-of-the-art hashing and deep learning-based forgery detection methods. Runtime analysis is also conducted to assess the computational efficiency of the framework.

4.3.1 | TDTVD Dataset Results

The proposed dual-stage framework was evaluated on the frame duplication subset of the TDTVD [13] dataset. As summarized in Table 1, the method achieved an average accuracy of 99.82%, recall of 100%, precision of 98.59%, and F1-score of 99.25% across duplication lengths of 10, 20, and 30 frames. The consistent 100% recall indicates the method's robustness in identifying all duplicated frames in the ground truth. Precision exhibited a positive correlation with the number of duplicated frames, increasing from 98.33% for 10-frame duplications to 98.96% for 30-frame duplications. This indicates that the method becomes increasingly reliable at rejecting false positives as the temporal scope of the forgery expands. The corresponding rise in F1-score to 99.46% for the longest duplication length further reflects this enhancement in balanced performance, as shown in Figure 3.

A comparison with existing approaches [19-22], is summarized in Table 2. The performance of the proposed framework is strong. The method achieved the highest reported accuracy of 99.82% and a perfect recall rate of 100% among all evaluated techniques, while also preserving a high precision of 98.59% and the top F1-score of 99.25%. This well-balanced performance indicates an enhanced ability to reliably detect forged content while minimizing false positives.

Table 1. Average Performance Rates for the proposed method using the TDTVD Dataset.

Number of frames duplication	Accuracy	Recall	Precision	F1 Score
10	99.87%	100%	98.33%	99.09%
20	99.79%	100%	98.48%	99.20%
30	99.80%	100%	98.96%	99.46%
Average	99.82%	100%	98.59%	99.25%

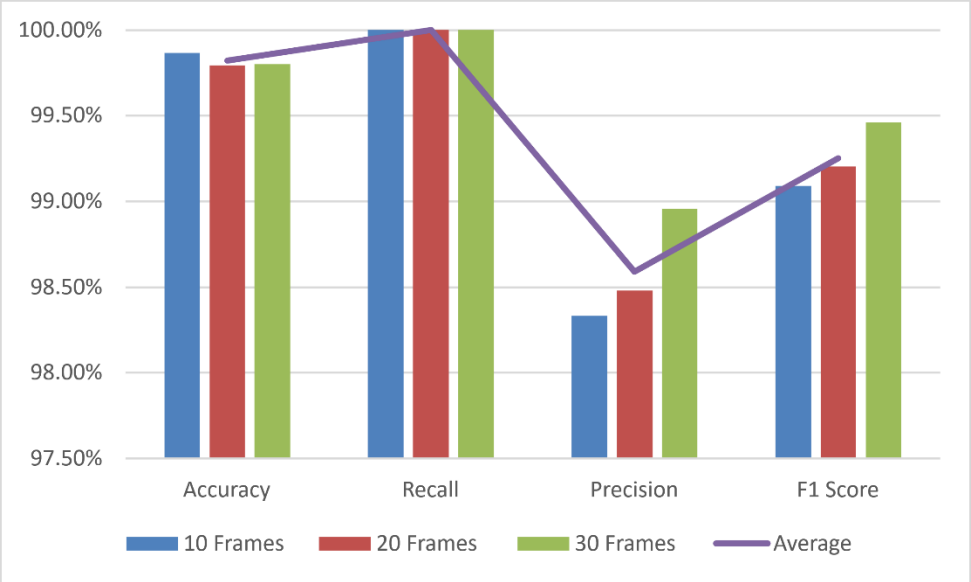

Figure 3. Performance assessment of the proposed method for detecting forgery videos in the TDTVD dataset.

Table 2. Accuracy comparison of the proposed method with other methods using the TDTVD dataset.

Method	Accuracy	Recall	Precision	F1 Score
Loukhaoukha [19]	99.43%	98.58%	97.52%	98.03%
Akhtar et al.[20]	99.57%	99.95%	99.61%	-
Shekar et al.[21]	96.67%	-	-	-
Abraham et al.[22]	97.5%	-	-	-
Proposed Method	99.82%	100%	98.59%	99.25%

4.3.2 | Results for Fadl and SULFA Datasets

To assess its generalization, the proposed framework was further evaluated on additional benchmark datasets. As reported in Table 3 for the Fadl dataset [16]. The method showed superior performance across all available evaluation metrics, achieving an accuracy of 99.58%, a precision of 99.02%, a recall of 99.47%, and an F1-score of 99.23%. These results demonstrate a clear performance advantage over previously reported approaches.

The performance comparison on the SULFA dataset, as detailed in Table 4, demonstrates the competitive efficacy of the proposed method. It maintains a balanced performance profile, with a recall of 99.02%, precision of 99.47%, and an F1-score of 99.23%. Notably, while the method by Singh et al. [23] reports a marginally lower accuracy of 99.50%, it achieves perfect precision 100%, and a slightly higher F1-score 99.40%. The proposed method provides a more comprehensive evaluation across all four-standard metrics. In comparison, the results for Raskar and Shah [24] is 97.00% accuracy, and Girish and Nandini [25] is 98.13% accuracy. This consistent performance across accuracy, precision, recall, and F1-score highlights the method's robustness and reliability in detecting video duplications within different benchmark datasets.

The characteristics of the three datasets under evaluation vary significantly, offering complementary perspectives on framework performance. The highest accuracy 99.82%, and perfect recall are explained by TDTVD, which includes controlled tampering scenarios with clean duplications and comprehensive ground truth. Fadl includes more realistic surveillance footage with varying resolutions from 320×240 to 1920×1080 and frame rates 15-30 fps, while maintaining high performance 99.58% accuracy, demonstrating robustness to resolution and frame rate variations. SULFA contains the most challenging scenarios with compressed surveillance footage and natural scene repetition, resulting in slightly lower recall 99.02% but maintaining high precision 99.47%. The consistent high performance across these diverse datasets demonstrates that the framework generalizes well across different video characteristics. The slight performance variation reflects the inherent difficulty of each dataset: TDTVD's controlled scenarios are easier to detect, while SULFA's compressed surveillance footage presents more challenging conditions. precision remains consistently high across all datasets, indicating that the motion verification stage effectively reduces false positives regardless of video characteristics.The proposed framework is optimized for static-camera scenarios, where duplicated frames exhibit minimal motion. In moving-camera scenarios such as handheld footage, drone videos, or action cameras, global camera motion introduces non-zero optical flow even in duplicated frames, potentially causing false negatives if the motion magnitude exceeds the threshold.

Table 3. Average Performance Rates for the proposed method compared with other methods using the Fadl dataset [16].

Method	Accuracy	Precision	Recall	F1-Score
Fadl et al. [16]	-	96.00%	94.00%	-
Bakas et al. [26]	-	67.00%	69.00%	-
Zhao et al. [27]	-	61.00%	65.00%	-
Liu and Huang [28]	-	51.00%	63.00%	-
Shelke et al.[29]	94.87%	95.66%	96.48%	96.07%
Fadl et al.[7]	98.50%	-	-	-
Proposed Method	99.58%	99.02%	99.47%	99.23%

Table 4. Average Performance Rates for the proposed method compared with other methods using SULFA [14].

Method	Accuracy	Recall	Precision	F1 Score
Singh et al.[23]	99.50%	99.00%	100%	99.40%
Raskar and Shah [24]	97.00%	-	-	-
Girish and Nandini [25]	98.13%	-	-	-
Proposed Method	99.58%	99.02%	99.47%	99.23%

5 | Conclusion

This paper presented a dual-stage framework for detecting duplication forgeries in video sequences. The method integrates a deep feature extraction module based on EfficientNetB0, a similarity detection stage with temporal constraints, and a verification stage using dense optical flow analysis. This combination effectively distinguishes malicious frame duplication from naturally similar. This pipeline is designed for an optimal balance of accuracy and computational efficiency, distinguishing our approach from previous dual-stage methods that often depend on heavier backbones or alternative verification techniques. The experimental results demonstrate the framework's high efficacy and robustness. The framework processed standard benchmark videos efficiently, demonstrating its feasibility for practical application on readily available hardware. While average optical flow magnitude may not capture complex motion patterns such as localized motion in specific regions or rotational motion. it achieved an average accuracy of 99.82% and a perfect recall of 100% on TDTVD dataset, indicating exceptional forgery localization. Its superior and balanced performance was further validated on the Fadl and SULFA datasets. The proposed framework ensures computational efficiency, making the approach scalable to real-world scenarios, and offers a reliable, accurate, and practical solution for detecting video duplication forgery. Future work will aim to robustly detect complex temporal forgeries under challenging conditions, while optimizing the pipeline for real-time embedded performance. Specifically, the framework will be extended to address more manipulations such as frame insertion and deletion. Efforts will also focus on improving resilience to significant camera motion and compression artifacts that currently degrade optical flow reliability.

Acknowledgments

The authors are grateful to the editorial team and the anonymous reviewers.

Author Contributions

All authors contributed equally to this work.

Funding

This research has no funding source.

Data Availability

The datasets generated during and/or analyzed during the current study are not publicly available due to the privacy-preserving nature of the data, but are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare no conflicts of interest in this research.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors.

References

[1] W. El-Shafai, M. A. Fouda, E.-S. M. El-Rabaie, and N. A. El-Salam, "A comprehensive taxonomy on multimedia video forgery detection techniques: challenges and novel trends," *Multimedia Tools and Applications,* vol. 83, no. 2, pp. 4241-4307, 2024/01/01 2024, doi: 10.1007/s11042-023-15609-1.

[2] M. M. Ali, N. I. Ghali, H. M. Hamza, K. M. Hosny, E. Vrochidou, and G. A. Papakostas, "Interframe Forgery Video Detection: Datasets, Methods, Challenges, and Search Directions," *Electronics,* vol. 14, no. 13, p. 2680, 2025.

[3] N. Akhtar, M. Hussain, and Z. Habib, "DEEP-STA: Deep Learning-Based Detection and Localization of Various Types of Inter-Frame Video Tampering Using Spatiotemporal Analysis," *Mathematics,* vol. 12, no. 12, p. 1778, 2024. [Online]. Available: https://www.mdpi.com/2227-7390/12/12/1778.

[4] M. R. Oraibi and A. M. Radhi, "Enhancement Digital Forensic Approach for Inter-Frame Video Forgery Detection Using a Deep Learning Technique," *Iraqi Journal of Science,* vol. 63, no. 6, pp. 2686-2701, 06/30 2022, doi: 10.24996/ijs.2022.63.6.34.

[5] C. Long, A. Basharat, and A. Hoogs, "Video Frame Deletion and Duplication," in *Multimedia Forensics*, H. T. Sencar, L. Verdoliva, and N. Memon Eds. Singapore: Springer Singapore, 2022, pp. 333-362.

[6] G. Singh and K. Singh, "Copy-Move Video Forgery Detection Techniques: A Systematic Survey with Comparisons, Challenges and Future Directions," *Wireless Personal Communications,* vol. 134, no. 3, pp. 1863-1913, 2024/02/01 2024, doi: 10.1007/s11277-024-10996-6.

[7] S. Fadl, Q. Han, and Q. Li, "CNN spatiotemporal features and fusion for surveillance video forgery detection," *Signal Processing: Image Communication,* vol. 90, p. 116066, 2021/01/01/ 2021, doi: https://doi.org/10.1016/j.image.2020.116066.

[8] M. Munawar and I. Noreen, "Duplicate Frame Video Forgery Detection Using Siamese-based RNN," *Intelligent Automation \& Soft Computing,* vol. 29, no. 3, pp. 927--937, 2021. [Online]. Available: http://www.techscience.com/iasc/v29n3/43054.

[9] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*, 2019: PMLR, pp. 6105-6114.

[10] C. Liu, J. Li, J. Duan, and H. Huang, "Video Forgery Detection Using Spatio-Temporal Dual Transformer," presented at the Proceedings of the 2022 11th International Conference on Computing and Pattern Recognition, Beijing, China, 2023. [Online]. Available: https://doi.org/10.1145/3581807.3581847.

[11] Z. Ma, T. Wang, S. Xu, X. Mu, Q. Wang, and Q. Guo, "Moving object Detection Based on Farneback Optical Flow," in *2023 42nd Chinese Control Conference (CCC)*, 2023: IEEE, pp. 7350-7355.

[12] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Scandinavian conference on Image analysis*, 2003: Springer, pp. 363-370.

[13] H. D. Panchal and H. B. Shah, "Video tampering dataset development in temporal domain for video forgery authentication," *Multimedia Tools and Applications,* vol. 79, no. 33, pp. 24553-24577, 2020.

[14] Sulfa dataset." http://sulfa.cs.surrey.ac.uk/forged.php (accessed 2020-07-08).

[15] O. I. Al-Sanjary, A. A. Ahmed, and G. Sulong, "Development of a video tampering dataset for forensic investigation," *Forensic science international,* vol. 266, pp. 565-572, 2016.

[16] S. Fadl, Q. Han, and L. Qiong, "Exposing video inter-frame forgery via histogram of oriented gradients and motion energy image," *Multidimensional Systems and Signal Processing,* vol. 31, no. 4, pp. 1365-1384, 2020.

[17] C. Cuevas, E. M. Yáñez, and N. García, "Labeled dataset for integral evaluation of moving object detection algorithms: LASIESTA," *Computer Vision and Image Understanding,* vol. 152, pp. 103-117, 2016.

[18] H. Sohn, W. De Neve, and Y. M. Ro, "Privacy protection in video surveillance systems: Analysis of subband-adaptive scrambling in JPEG XR," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 21, no. 2, pp. 170-177, 2011.

[19] K. Loukhaoukha, "Frame duplication forgery detection and localization based on QR decomposition and Minkowski distance," *Journal of Forensic Sciences,* 2025.

[20] N. Akhtar, M. Hussain, and Z. Habib, "Two–Stage Detection and Localization of Inter–Frame Tampering in Surveillance Videos Using Texture and Optical Flow," *Mathematics,* vol. 12, no. 22, p. 3482, 2024.

[21] B. Shekar, W. Abraham, and B. Pilar, "A Simple Difference Based Inter Frame Video Forgery Detection and Localization," in *International Conference on Soft Computing and its Engineering Applications*, 2023: Springer, pp. 3-15.

[22] W. Abraham, B. H. Shekar, and B. Pilar, "Inter-frame Video Forgery Detection—A PCA-Based Approach," presented at the Fifth International Conference on Computing and Network Communications, Singapore, 2025.

[23] G. Singh and K. Singh, "Video frame and region duplication forgery detection based on correlation coefficient and coefficient of variation," *Multimedia Tools and Applications,* vol. 78, no. 9, pp. 11527-11562, 2019/05/01 2019, doi: 10.1007/s11042-018-6585-1.

[24] P. S. Raskar and S. K. Shah, "VFDHSOG: Copy-Move Video Forgery Detection Using Histogram of Second Order Gradients," *Wireless Personal Communications,* vol. 122, no. 2, pp. 1617-1654, 2022/01/01 2022, doi: 10.1007/s11277-021-08964-5.

[25] N. Girish and C. Nandini, "Inter-frame video forgery detection using UFS-MSRC algorithm and LSTM network," *International Journal of Modeling, Simulation, and Scientific Computing,* vol. 14, no. 01, p. 2341013, 2023, doi: 10.1142/s1793962323410131.

[26] J. Bakas, R. Naskar, and R. Dixit, "Detection and localization of inter-frame video forgeries based on inconsistency in correlation distribution between Haralick coded frames," *Multimedia Tools and Applications,* vol. 78, pp. 4905-4935, 2019.

[27] D.-N. Zhao, R.-K. Wang, and Z.-M. Lu, "Inter-frame passive-blind forgery detection for video shot based on similarity analysis," *Multimedia Tools and Applications,* vol. 77, pp. 25389-25408, 2018.

[28] H. Li, W. Luo, X. Qiu, and J. Huang, "Image forgery localization via integrating tampering possibility maps," *IEEE Transactions on Information Forensics and Security,* vol. 12, no. 5, pp. 1240-1252, 2017.

[29] N. A. Shelke and S. S. Kasana, "Multiple forgeries identification in digital video based on correlation consistency between entropy coded frames," *Multimedia Systems,* vol. 28, no. 1, pp. 267-280, 2022/02/01 2022, doi: 10.1007/s00530-021-00837-y.