Hindawi
International Journal of Intelligent Systems
Volume 2023, Article ID 8378073, 17 pages
https://doi.org/10.1155/2023/8378073

Research Article

Video Surveillance Object Forgery Detection using PDCL Network with Residual-based Steganalysis Feature

Yan-Fen Gan,[1] **Ji-Xiang Yang**,[2] **and Jun-Liu Zhong**[3]

[1]*School of Computer Science, The South China Business College, Guangdong University of Foreign Studies, Guangzhou 510545, China*
[2]*School of Electronics and Communication, Guangdong Mechanical and Electrical Polytechnic, Guangzhou 510550, China*
[3]*Department of Information and Communication Engineering, Guangzhou Maritime University, Guangzhou 510725, China*

Correspondence should be addressed to Jun-Liu Zhong; junliuzhong@foxmail.com

Received 12 February 2023; Revised 20 May 2023; Accepted 24 May 2023; Published 20 June 2023

Academic Editor: B. B. Gupta

Video surveillance has various applications in various fields and industries. However, the rapid development of video processing technology has made video surveillance information susceptible to multiple malicious attacks. At present, the state-of-the-art methods, including the latest deep learning techniques, cannot get satisfactory results when addressing video surveillance object forgery detection (VSOFD) due to the following limitations: (i) lack of VSOFD-specific features for effective processing and (ii) lack of effective deep network architecture designed explicitly for VSOFD. This paper proposes a new detection scheme to alleviate these limitations. The proposed approach first extracted VSOFD-specific features via residual-based steganalysis feature (RSF) from the spatial-temporal-frequent domain. Key clues of video frames can be more effectively learned from RSF, instead of raw frame images. Then, the RSF feature is used to form the residual-based steganography feature vector group (RSFVG), which serves as the input of our following network. Finally, a new VSOFD-specific deep network architecture called parallel-DenseNet-concatenated-LSTM (PDCL) network is designed, which includes two improved CNN and RNN modules. The improved CNN module fuses and processes the coarse-to-fine feature extraction and simultaneously preserves the frame independence in video frames. The improved RNN module learns the correlation features between the adjacent frames to identify forgery frames. Experimental results show that the proposed scheme using the PDCL network with RSF can achieve high performance in test error, *precision*, *recall*, and F_1 scores in our newly constructed dataset (SYSU-OBJFORG + newly generated forgery video clips). Compared to existing SOTA methods, our framework achieves the best F_1 score of 90.33%, which is greatly improved by nearly 8%.

1. Introduction

Video surveillance has extensive applications across various industries and fields [1–3]. For instance, they can serve as evidence in trials, providing a basis for subsequent investigations. Furthermore, video evidence can be used for news material, insurance claims, intelligence agencies, etc. However, with the continuous advancements in multimedia processing technology, the manipulation of videos has become commonplace. Therefore, forensic analysis for multimedia aims to ensure the authenticity of the content. Due to the extensive usage of surveillance videos, verifying the authenticity of such information has become a critical component of multimedia forensic analysis. Object-based forgery is a common method of video forgery since adding or removing an object from a surveillance video can significantly alter its meaning. It has seriously affected people's trust in the authenticity of court evidence, news reports, and other surveillance video evidence, which has significant negative effects on society. For this reason, video surveillance object forgery detection (VSOFD) has become urgently demanding.

Digital video forgery techniques were previously studied to address VSOFD. Existing popular digital video forgery generally consists of *frame*-based forgery and *object*-based forgery [4], as portrayed in Figure 1. Frame-based forgery

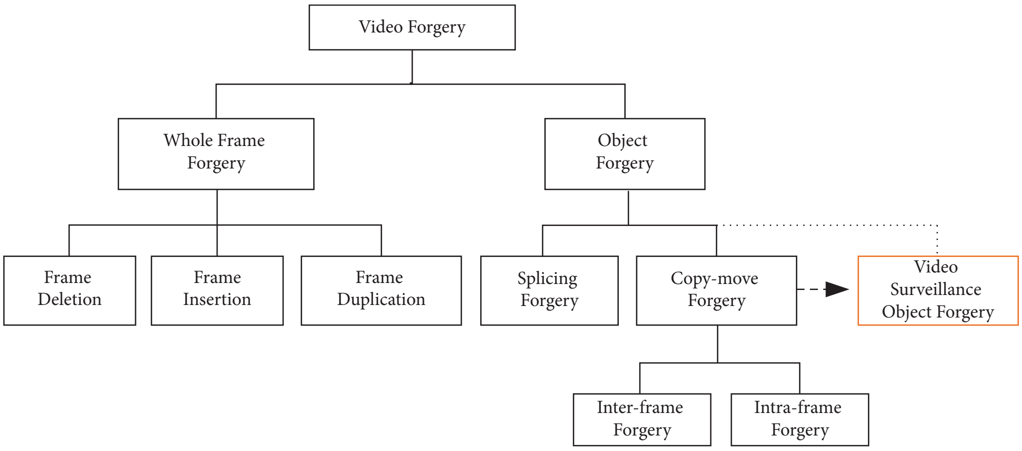


Figure 1: The mainly digital video forgery techniques.

takes the whole frame content as the operating cell, which includes frame duplication (insertion) [5] and frame deletion [6], and, respectively, aims to duplicate and delete some successive frames in the same video or different videos to highlight or conceal some critical events. Currently, many state-of-the-art (SOTA) detection methods devote to frame-based forgery and achieve satisfactory results, including frame correlation statistics difference [5], frame motion residual [6], and deep learning [7].

Unlike frame-based forgery with obvious traces (*e.g.*, sudden lighting change, temporal flickering, scene fluctuation), object-based forgery [8–14] with sophisticated techniques can achieve realistic forgery effects. Taking a particular account of the video objects reflecting the video meaning and contents is significant to object-based forgery detection. Generally, object-based forgery contains two subbranches, namely, *splicing* and *copy-move* forgeries (Figure 1). Splicing is a forgery with heterogeneous sources. The copied (splicing) source and the pasted frame (localization) originated from different video clips (Figure 2(a)). Many videos splicing forgery detection (VSFD) methods reference the image splicing detection schemes [15–17], which search for the splicing traces between the heterogeneous splicing content(s) and the spliced background of the target video. Besides, the SOTA methods also consider temporal dimension to analyze spatial-temporal block correlation [8] or use recurrent neural networks (RNN) with temporal correlation [9] to locate and distinguish forgery frames.

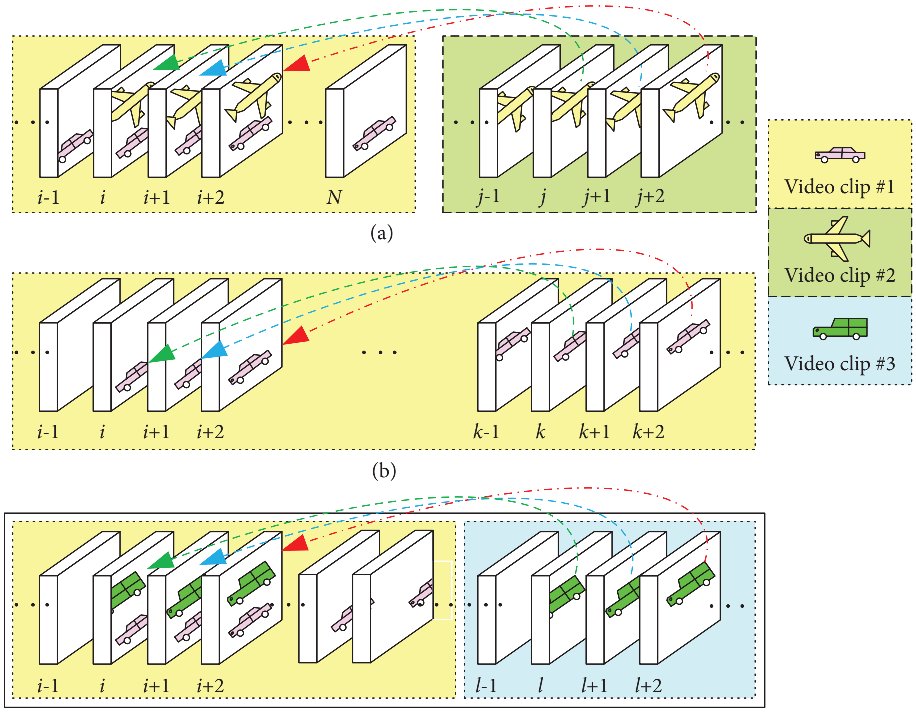
Copy-move is a forgery with a homogenous source. The copied source and its pasted frame (localization) originated from the same video clips (Figure 2(b)) [10]. This forgery technique can consist of both intra-frame and inter-frame manipulations. In intra-frame forgery, the object(s) are copied from one frame and pasted into the same frame. In inter-frame forgery, the object(s) are copied from one frame and pasted into other different frames within a short interval (e.g., 200 frames) to create a realistic forgery. These phenomena provide significant detection clues to search for matching correlations since the copied and pasted objects are in the same video clip. Recent video copy-move forgery detection (VCMFD) methods reference image copy-move forgery detection (CMFD) schemes [11] and consider temporal dimension for addressing both intra-frame [12, 13] and inter-frame forgeries [14].

As discussed previously, the relatively consistent content expression in video copy-move forgery can easily create realistic forgery effects. In practice, video surveillance object forgery is like film plots. Malicious attackers generally employ copy-move forgery variations, in which the copied object(s) and the pasted frame are separated for a relatively long time. A suspicious video clip is usually considered rather than a surveillance video of unlimited continuous length for video forensics. Therefore, this forgery can make the suspicious video clip contain only the pasted object(s) without any copied source clues. As a result, it can easily bypass most existing video surveillance object forgery because the current methods cannot find the copied and pasted pairs. This special forgery variation (as shown in Figure 2(c)) is a popular video surveillance object forgery.

The existing video splicing forgery and copy-move forgery detection methods fail to detect video surveillance object forgery because of two reasons:

(1) In video splicing forgery, the content of the forgery video is originated from two different videos and contains equipment information from different cameras. According to the characteristics of a video splicing forgery, the splicing detection methods can only identify the splicing traces between the splicing content(s) and the spliced background of the forgery video. However, the copy-move content of the video surveillance object forgery originated from the same videos (their equipment information belongs to one camera) but different video clips. Therefore, the video splicing detection methods fail to identify the copy-move forgery traces captured by the same camera in VSOFD.

(2) In video copy-move forgery, this forgery's copy-move content originates from the same video clips (their equipment information belongs to one camera). According to the characteristics of a video copy-move forgery, the copy-move detection methods



(a)

(b)

(c)

Figure 2: The video splicing, video copy-move, and video surveillance object forgery techniques. The (*l*-*i*) >> (*k*-*i*) means that the interval between frame *l* and frame *i* is far longer than frame *k* and frame *i* in the same video. (a) Video splicing forgery in different videos. (b) Video copy-move forgery in a same video clip. (c) Video surveillance object forgery in a same video but different video clips.

must find the one-to-one matching correlation between the copied and pasted objects. However, in the video surveillance object forgery, the copied objects and paste frame do not appear in the same suspicious video clip (but in other parts of the surveillance video clip). Therefore, the video copy-move forgery detection methods can easily bypass most existing video surveillance object forgery because the current methods cannot find the copied and pasted pairs in VSOFD.

Only a few methods [4, 18–20] have been proposed to address VSOFD. However, even the latest deep learning model in existing works cannot provide satisfactory detection accuracy due to two main challenges:

(i) Discriminative feature in the independent spatial, temporal, or frequent domain for existing VSOFD: Some methods based on motion residual [4, 19] for VSOFD, can only extract spatial or temporal domain features and do not consider changes in the frequency domain of the forged video. Similarly, current deep learning methods [12, 18, 21], CNN or RNN, are not specifically designed for VSOFD, which only address the feature extraction in the spatial or temporal domains, respectively. The features of the above algorithms cannot effectively discover the forgery trace. Therefore, extracting comprehensive and effective features for VSOFD is necessary.

(ii) Effective deep network architecture for VSOFD: A video is a time series of sequential and correlated frames, while each frame has its independence. CNN is powerful in handling spatial characteristics. It is likely to employ CNN for VSOFD. However, CNN only takes a frame as input. In this situation, CNN fails to keep frame coherence in the video, leading to unsatisfactory detection accuracy. Although RNN can maintain frame coherence, it does not possess the ability to handle spatial characteristics. Therefore, a new architecture having both the abilities of CNN and RNN is necessary.

To address these two defects, a novel VSOFD scheme integrated with several newly designed techniques is proposed, including spatial-temporal-frequent comprehensive and effective deep network architecture:

Effective residual-based steganalysis feature (RSF) is designed explicitly for the spatial-temporal-frequent domain. Residual-based signal (RS) is first extracted from the spatial-temporal domain. Subsequently, RSF is further extracted from the RS in the frequency domain. The RSF can effectively represent spatial-temporal-frequent perspectives with dimensionality reduction in features. Then, the RSF feature is used to form the residual-based steganography feature vector group (RSFVG), which serves as the input of our following network.

A new learning framework called *parallel-DenseNet-concatenated-LSTM* (PDCL) network is designed, which combines both the CNN (parallel-DenseNet) and the RNN

(LSTM) structures. The proposed PDCL structure simultaneously preserves frame independence (spatial domain) and captures correlation (temporal domain). Furthermore, its CNN module has a parallel cross-layers-block structure that extracts the coarse-to-fine fusion feature for processing. Noteworthy, the CNN module in PDCL is compatible with different RSF and preserves the RSF independence of each frame with a column convolutional kernel. Subsequently, the independent CNN frame features are concatenated as input for the RNN module to learn the correlation and coherence of the frame sequence.

The rest of the paper is organized as follows. The related work is presented in Section 2. The proposed RSF and parallel-DenseNet-concatenated-LSTM (PDCL) are detailed in Sections 3 and 4, respectively. The experiments and conclusions are provided in Sections 5 and 6, respectively.

2. Related Work

2.1. Related Video Surveillance Object Forgery Detection. The video surveillance object forgery can achieve realistic vision results without leaving tampering traces. Unfortunately, there are very few research works on VSOFD in the literature. In recent years, Chen et al. [4] created the SYSU-OBJFORG dataset, introduced video object forgery detection, and proposed a temporal forgery detection algorithm based on motion residuals. All video clips in the available SYSU-OBJFORG are extracted from primitive video footage of several static surveillance cameras. Moreover, a substantial part of forgery video clips in the SYSU-OBJFORG has the same properties as our study issue (video surveillance object forgery). Therefore, video object forgery detection can provide an excellent reference to video surveillance object forgery detection (VSOFD).

The inherent statistical properties of a video can be divided into two categories: the inherent intra-frame properties describing its spatial characteristics and the inherent inter-frame properties describing its temporal characteristics. Since the motion residuals contain the inherent attributes of the corresponding frames both within and between them, it becomes the primary analysis tool of VSOFD. Two hand-crafted automatic identification algorithms [4, 19] rely on motion residual (MR) as the feature of each frame and use a machine learning classifier for discriminating the forgery and genuine frames. Motion residuals can only reflect the spatial-temporal domain's correlation but ignore the frequency characteristic changes of a forged video in the features (defect (i)). Meanwhile, these classifiers are difficult to handle many hyper-parameter adjustments. It is only well-designed for specific forgery datasets (*e.g.*, SYSU-OBJFORG). They cannot provide satisfactory detection efficiency and accuracy (defect (ii)).

Yang et al. [18] propose a deep network based on a spatial rich model (SRM) and 3D convolution (C3D) to address an application similar to VSOFD. However, this general CNN structure does not include discriminative features in the spatial-temporal-frequent model (defect (i)). It needs to extract the difference and coherence between the successive frames. Furthermore, its general CNN structure (without RNN) is incompetent in addressing the VSOFD effectively (defect (ii)) and cannot process slowly moving forgery objects. Jin et al. [21] propose dual-stream networks for video object-based forgery detection. This technique is similar to fast forgery detection [12]. It uses corresponding DNN modules to replace the hand-crafted feature extraction, processing, and tracking modules, e.g., dual-stream networks for feature extraction instead of exponential Fourier moments [12]. It is well-designed for both splicing and general copy-move forgery detection. Still, features extracted from the convolutional layer lack the spatial-temporal-frequent perspective (defect (i)), leading to the failure of VSOFD detection. Moreover, these methods [12, 21] cannot detect occlusive or smoothing background forgery.

The previous studies lack comprehensive and effective features and an effective deep network architecture for VSOFD, which we have discussed in this related work. To address these two defects, a novel VSOFD scheme integrated with newly designed techniques is proposed: (i) Effective residual-based steganalysis feature (RSF) is designed which can effectively represent spatial-temporal-frequent perspectives with dimensionality reduction in features. (ii) The proposed PDCL structure simultaneously preserves frame independence (spatial domain) and captures correlation (temporal domain) to effectively improve VSOFD results.

2.2. Residual Signal and Steganalysis Feature. The motion residual is a popular feature extraction for video forensics. To address motion-compensated frame rate up-conversion (MC-FRUC), Ding et al. [22] build a residual signal to search for the forgery splicing traces. Then, the identification problem of MC-FRUC is transformed into a classification or discrimination problem of differences in residual signals between the video frames. Saddique et al. [19] also rely on motion residual (MR) as the feature of each frame for discriminating the forgery and genuine frames.

First, the feature extraction method uses an overlapping temporal window (aggregated operation) to slide in the video sequence one frame at a time. Then, the minimum, maximum, and median of motion residual inside a temporal frame window create the aggregated frames AF. For the k^{th} frame in the video,

$$\begin{aligned} AF_{x,y}^{k} &= \mathrm{Min}\ \text{ or }\ \mathrm{Max}\ \text{ or }\ \mathrm{Med}\left[F_{x,y}^{k-Ls}, F_{x,y}^{k-Ls+1}, \ldots, F_{x,y}^{k}, \ldots, F_{x,y}^{k+Ls-1}, F_{x,y}^{k+Ls}\right], \\ RS_{x,y} &= \left|F_{x,y}^{k} - AF_{x,y}^{k}\right|, \end{aligned} \tag{1}$$

where min, max, and med mean the minimum, maximum, and medium values; $2Ls+1$ is the number of the aggregated frames $F_{x,y}$, which contains the center (current) frame k in the aggregated frames; Ls represents the number of the successive previous and subsequent frames of frame k; and x and y are the pixel position in the corresponding frame. $RS_{x,y}$ is the residual-based signals (RS).

The motion residual can reflect the correlation of fine-grained pixels between the adjacent frames but lacks global vision. Some popular steganography techniques, *e.g.*, DCT [23], CCPEV [24], subtractive pixel adjacency matrix (SPAM) [25], spatial domain rich model (SRMQ1) [26], spatial and color rich model (SCRMQ1) [27], as the effective features, are extracted from frame residual to obtain the global vision. Therefore, combining motion residual and steganography effectively obtains the local and global features for discriminating video forgery frames.

2.3. CNN + RNN (DenseNet and LSTM) Structure for Feature Processing and Forgery Frame Identification. With the rapid development of convolutional neural networks (CNN), many effective CNN models, such as VGG16 [28] and ResNet [29], have shown powerful feature extraction abilities and excellent image classification in the spatial perspective. However, these classic CNN models have a feed-forward transferring structure whose current network layer only receives the processed information from the preceding layer. Subsequently, the information from the current layer is processed and transferred to the next layer sequentially. Moreover, these classic CNN models have some defects, *e.g.*, of many parameters, network layers, and widths, which may easily cause overfitting. Huang et al. [30] proposed a DenseNet instead of the VGG16 serial structure, with a concatenated structure that transfers the concatenated feature maps of all preceding layers. DenseNet gives a good reference for the proposed VSOFD DNN architecture. Furthermore, the video is a time series. Any video frame and its neighboring frames are temporal-dependent. RNN technique [31], especially LSTM, can capture this long short-term dependence between the preceding and current frames. Therefore, it is suitable for the correlation statistics between the video frames.

3. The Proposed Residual-Based Steganalysis Feature Extraction

The proposed method includes two main stages: (1) residual-based steganalysis feature extraction (Figure 3) and (2) parallel-DenseNet-concatenated-LSTM for feature learning and forgery frame detection (Figure 4).

Practically, the surveillance video clip contains the stationary scenes and the object motion parts. In the video surveillance object forgery, static scenes cover the target object and remove its motion traces. This kind of object removal belongs to the spatial domain. Nevertheless, video is a continuous time series of frames, and each video frame strongly correlates with its adjacent frames. Therefore, video surveillance object forgery is a kind of attack to change spatial-temporal coherence. In our work, a novel motion residual extraction strategy is proposed for RS with the following concerns, which is much different from the literature [4, 19]:

(i) Microscopic level in the spatial-temporal domain (Section 3.1): the fine-grained pixel difference between the detected pixel and the adjacent pixels in the spatial domain, and the fine-grained pixel difference between the current frame position and the adjacent frame positions in the temporal domain.

(ii) Macroscopic level in the frequent domain (Section 3.2): steganography features differences between the frame and adjacent frames in the energy and frequent domains.

In our work, the proposed residual-based steganalysis feature extraction is shown in Figure 3, where Figures 3(a)–3(c) present the residual-based signal (RS) extraction for concern (i) and residual-based steganography feature (RSF) extraction for concern (ii), respectively.

3.1. Residual-Based Signal Extraction. Given concern (i), the proposed RS extraction considers spatial-temporal coherence and difference analysis. First, it is the fact that the closer positions of two frames in a video sequence indicate a higher frame correlation. Therefore, the RS extraction is within only a short temporal frame window $Ls=\{1, 2\}$, instead of a long temporal frame window, *e.g.*, $Ls \geq 10$ in [19]. Considering that the Laplacian operator can benefit from enhancing the forgery traces and sharpening the image details, a Laplacian operator is employed to create RS instead of calculating the minimum, maximum, and median of motion residual in existing schemes [4, 19]. Since the video frame sequence is a 1-dimensional (1-D) time series of frames, the proposed RS extraction only requires calculating a two-order discrete Laplacian operator in the 1-D temporal domain (equation (2)).

$$\nabla^2 f[k] = f''[k] = \begin{cases} f[k-1] - 2f[k] + f[k+1], & \text{when } Ls = 1, \\ \dfrac{(f[k-2] + f[k-1] - 4f[k] + f[k+1] + f[k+2])}{4}, & \text{when } Ls = 2, \end{cases} \tag{2}$$

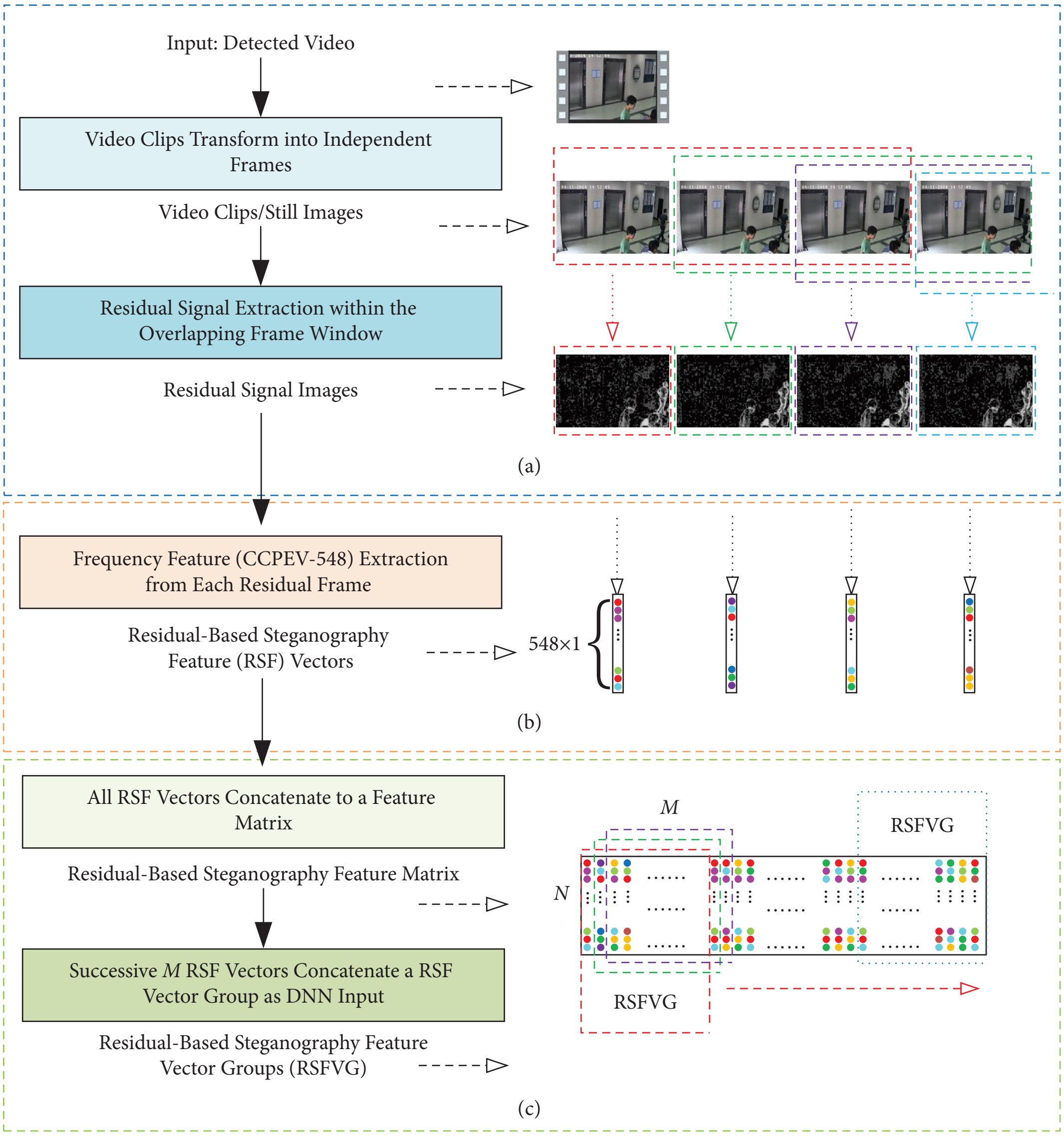

Input: Detected Video
Video Clips Transform into Independent Frames
Video Clips/Still Images
Residual Signal Extraction within the Overlapping Frame Window
Residual Signal Images
(a)
Frequency Feature (CCPEV-548) Extraction from Each Residual Frame
Residual-Based Steganography Feature (RSF) Vectors
548×1
(b)
All RSF Vectors Concatenate to a Feature Matrix
Residual-Based Steganography Feature Matrix
Successive M RSF Vectors Concatenate a RSF Vector Group as DNN Input
Residual-Based Steganography Feature Vector Groups (RSFVG)
M
N
RSFVG
RSFVG
(c)

Figure 3: The proposed residual-based steganalysis feature extraction. (a) Residual-based signal extraction. (b) Residual-based steganography feature extraction. (c) Residual-based steganography feature vector group.

where $f\,[k]$ is the gray-scale features of all pixels in the kth frame, and k is the center frame in a short temporal window. Following the literature [32], the convolution kernels of the 1-D and 2-D Laplacian operators are set to [1, −2, 1] and [1, 1, −4, 1, 1].

Second, in the spatial domain, a two-order discrete Laplacian operator in equation (3) for the 2-D image or video frame is applied to remove the subtle camera-shaking interference, i.e., Gaussian noise, to highlight the areas where pixel values change rapidly.

$$\begin{aligned}\nabla^2 p[x,y] &= D_x^2[p(x,y)] + D_y^2[p(x,y)]\\ &= p[x+1,y] + p[x-1,y] - 4p[x,y] + p[x,y+1] + p[x,y-1],\end{aligned} \tag{3}$$

where x, y is the pixel coordinate in the residual-frame map $\nabla^2 f\,[k]$, $\nabla^2 p\;[x,y]$ is the proposed RS, and the convolution kernel of the 2-D Laplacian operators are [[0, 1, 0]; [1, −4, 1]; [0, 1, 0]].

3.2. Residual-Based Steganography Feature Extraction. Steganalysis can analyze, find, and distill the hidden information of steganographic carriers at a macroscopic level. In addressing concern (ii), steganography feature selection is

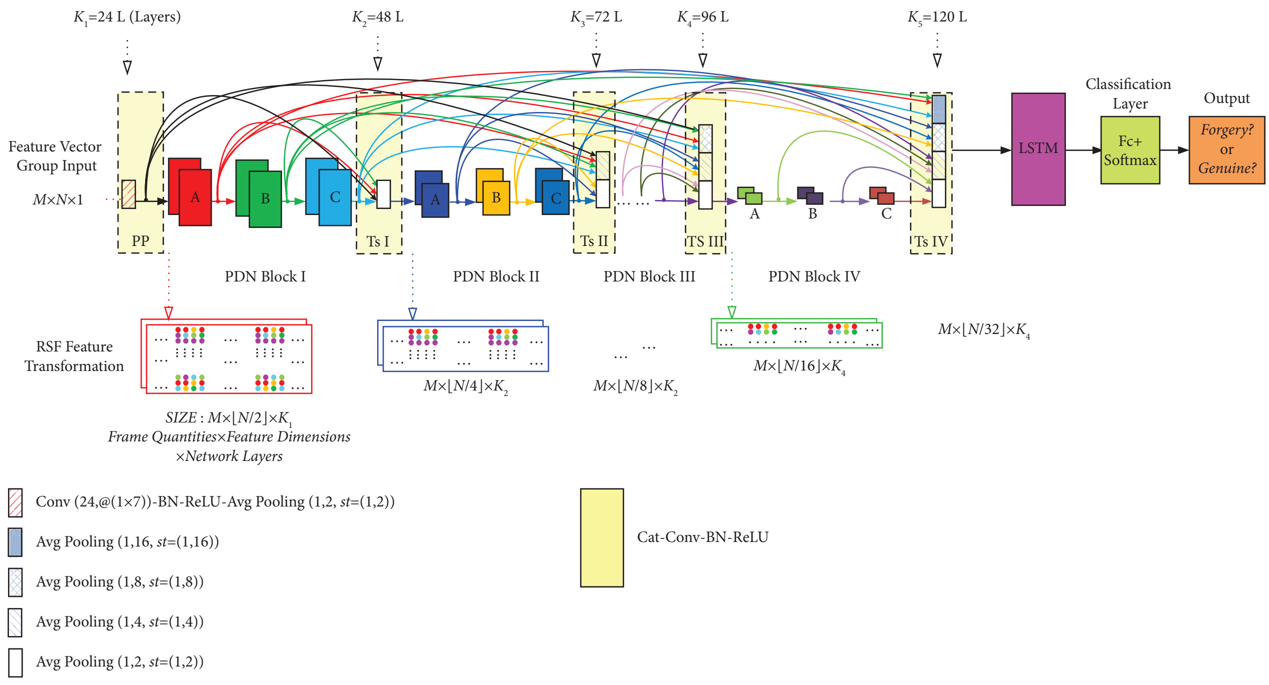


Figure 4: The proposed parallel-DenseNet-concatenated-LSTM structure for VSOFD.

crucial in feature extraction. In our work, the residual-based steganography feature (RSF) extraction further uncovers the steganography feature from the residual-based signal (RS) as the feature representation. RSF extraction transforms a spatial-temporal feature matrix (*e.g.*, RS) into a spatial-temporal-frequent feature vector that can reduce the huge amount of information (*e.g.*, from 3D to 2D).

There are two reasons to employ RSF as the VSOFD feature: (1) effectiveness: most of the SOTA steganography techniques are competent in this work, and (2) efficiency: the compact steganography feature (namely, the short-length feature or low-dimension) can relieve "the curse of length or dimensionality." The compact steganography feature is also suitable for the following deep model training. Among RSF techniques, DCT [23], CCPEV [24], and SPAM [25] are selected for steganography feature extraction. Figure 3(b) illustrates the CCPEV feature with 548-D. Each RSF vector is $N \times 1$ dimensional, where N is the feature dimension. The SRMQ1 [26] and SCRMQ1 [27] with relatively long-length (or high-dimensional) features are used for the experimental comparisons. Algorithm1 shows the RS and RSF extraction algorithms.

3.3. Samples Processing Using Residual-Based Steganography Feature Vector Group (RSFVG). In this sample processing algorithm, vision persistence is considered. In the human psychophysical system (*i.e.*, vision persistence), an object in a video clip appears at least 0.1–0.4 seconds, *i.e.*, 3–10 frames. For this reason, a certain number of successive frames (called a frame group) can better represent the temporal characteristics of deep model learning. In Section 3.2, the RSF vectors of all frames are concatenated into a $W \times N \times 1$ matrix, where W is the total number of frames in the detected video clip, and N is the feature dimensions of RSF. We apply a local temporal vector sliding window in an RSF matrix, with one vector at a time, to obtain different combinations of successive RSF vectors in the corresponding frame group (Figure 3(c)), namely, the RSF vector group (RSFVG). RSFVG helps to provide sufficient samples for effective learning. The relations of RS, RSF, and RSFVG are illustrated in the right part of Figure 3. Based on the requirement of vision persistence, each RSFVG has $M = 2L_M + 1$ frames (RSF vectors), where L_M is no less than 3 and $L_M > Ls$.

The ground-truth $\mathrm{GT_F}$ for each RSF vector (frame) is labeled in a binary code. The binary of 0 and 1 represents a genuine frame and a forgery frame, respectively. The $\mathrm{GT_F}$ of each RSFVG is a GT vector with the size of $M = (2L_M + 1) \times 1 \times 1$. Algorithm 2 shows the sample processing algorithm.

4. Parallel-DenseNet-Concatenated-LSTM (PDCL) Architecture

Similar to RSF extraction, it is also required to consider the spatial-temporal perspectives in deep models. For this reason, a novel architecture called parallel-DenseNet-concatenated-LSTM (PDCL), combining CNN and RNN, is proposed. The overview of PDCL is illustrated in Figure 4.

4.1. Parallel-DenseNet Framework. DenseNet [30] is a kind of CNN with excellent performance in pattern recognition and image/video classification tasks. However, DenseNet is a feed-forward serial structure that can only process the concatenated features (in the same feature size). For this reason, DenseNet cannot simultaneously handle the coarse

```
Input: The video frames $V_F = \{V_{F1}, V_{F2}, \ldots, V_{FW}\}$.
#$W$ is the number of frames in a detected video; RS is extracted from the number of frames $2Ls + 1$ including the center (current) frame $k$, successive previous frames $Ls$, and subsequent frames $Ls$ of frame $k$; $X_M$ and $Y_N$ are the width and height of the frame.
For $k = Ls + 1$ to $W - Ls$ do
    Calculate two-order discrete Laplacian operator $\nabla^2 f(k)$ in a short temporal frame window $\{k\text{-}Ls, \ldots, k, \ldots, k + Ls\}$; # equation (2)
  For $x = 2$: $X_M - 1$ do
    For $y = 2$: $Y_N - 1$ do
      Calculate a two-order discrete Laplacian operator $\nabla^2 p_k(x, y) = \mathrm{RS}_k$ in a local neighboring region $\{(x - 1, y - 1), \ldots, (x + 1, y + 1)\}$; # equation (3)
    End
  End
End
For $k = Ls + 1$ to $W - Ls$ do
  Extract $\mathrm{RSF}_k$ vector from $\mathrm{RS}_k$.
End
Output: RSF vector of each frame.
```

Algorithm 1:Residual-based signal (RS) and residual-based steganography feature (RSF) extraction algorithm.

```
Input: The RSF matrix $V_M$ with the corresponding residual-based steganography feature vectors $\{RSF_1, RSF_2, \ldots, RSF_W\}$.
#$W$ is the total number of frames in a detected video or all RSF vectors in $V_M$; RS is extracted from the number of frames $2Ls + 1$; RSF is extracted from the corresponding RS; the $2L_M + 1$ RSF vectors form a group (RSFVG); GT means the binary ground-truth of each frame or vector (Genuine = 0, Forgery = 1).
For $j = L_M + 1$ to $W - L_M$ do
  Concatenate RSF vectors $\{RSF_{j\text{-}LM}, RSF_{j\text{-}LM+1}, \ldots, RSF_{j+LM}\}$ in RSF $V_M$ to create an $\mathrm{RSFVG}_j$ with the size of $M \times N \times 1$;
  Label the GT of all frames in the $\mathrm{RSFVG}_i$ matrix to create GT group $\mathrm{GT}_{Fj} = [\mathrm{GT}_{j\text{-}LM}, \mathrm{GT}_{j\text{-}LM+1}, \ldots, \mathrm{GT}_{j+LM}]$ with the size of $M \times 1 \times 1$;
End
Output: The corresponding RSFVG data for DNN learning.
```

Algorithm 2: Sample processing relying on residual-based steganography feature vector group (RSVFG) for DNN input.

and fine features (in different sizes). From this perspective, our parallel-DenseNet (PDN) is proposed to address this issue by concatenating the serial and parallel features (*i.e.*, cross-layer and cross-block features) from the preceding layers and blocks. In this way, the coarse-to-fine features can be simultaneously learned.

Section 3.3 and Figure 4 mention that the input feature map is a 3-D tensor $X \in \mathbb{R}^{M \times N \times K}$. First, each box and its arrowed line with different colors represent the output of the corresponding layer of the PDN block. Each PDN block is a serial structure consisting of three layers of A, B, and C. Different PDN blocks and Ts layers are parallel structures. The color dots in the bottom figure illustrate an RSFVG. M is the number of concatenated RSF vectors or frame quantity in an RSFVG. Note that M (width of the color dot boxes at the bottom of Figure 4) is fixed to preserve their feature independence in the PDN structure, and the feature dimension (height of the color dot boxes at the bottom of Figure 4) is cut in half in the whole PDCL processing. This design benefits the following LSTM module to learn the temporal correlation between the frames in RSFVG. The PDCL architecture with adjustable capability is compatible with various RSF dimensions. Figure 3(b) takes the CCPEV steganalysis feature as an input RSF sample, *e.g.*, $N = 548$, to simplify the analysis. The initial channel K of the RSFVG feature map is one layer. Therefore, the feature map size of the PDCL input is $M \times 548 \times 1$.

The PDN framework consists of the preprocessing (PP) layer, PDN block, and transition (Ts) layer. The PP layer is a Conv-BN-ReLU-AvgPool block, similar to the Ts layer. Noteworthy, Convolution (Conv) uses a column convolutional kernel of $(1, c = 7)$ instead of a conventional square kernel of (c, c) for processing. This way can keep M unchanged and each RSF vector independent. Conv also extends the feature map from 1 channel to 24 channels. Batch normalization (BN) can then remove the internal covariate shift and retain the same distribution of each layer. Rectified linear unit (ReLU) is a nonlinear transformation function that can decrease gradient vanishing and speed up the network training. Finally, an AvgPool operation implements the feature map filtering. The stride (1, 2) of AvgPool also aims to keep the M-independent RSF vectors in an RSFVG and compresses the feature dimension from $N = 548$ to $N/2 = 274$. Finally, the PP layer outputs a rough PDN feature map with a size of $M \times \lfloor N = 548/2 \rfloor \times 24$.

Then, the PDN blocks and the subsequent Ts layer constitute the backbone of PDN. Figure 4 shows the 4 PDN blocks (I, II, III, IV) with 3 PDN layers. The 4 PDN blocks sequentially extract the coarse-to-fine and multiscale dense features. Each PDN layer is a Conv-BN-ReLU structure (referring to the DeneseNet layer architecture [30]), as shown in Figures5(b), 5(d), 5(f), and 5(h). It is well-known that more network layers improve the network performance but with more serious effects of gradient vanishing, redundant information, and higher computation cost. Therefore, two improvements are required in the DenseNet block and Ts layer.

4.1.1. Serial Structure. Each PDN block has the same output layer depth with a modest 24 channels.

In a PDN block, each layer has the same series (Conv-BN-ReLU structure, Figure 6). As shown in Figure 5(b), the column convolutional kernel of (1, $c = 7$) is used similar to the Conv in the preprocessing layer. Since a frame RSF vector with a limited feature length is much simpler than the image content with 3-D rich details, each PDN layer outputs only 8 channels. In the PDN block (Figure 6), the three layers, A, B, and C, all output 8 channels, namely, the growth rate $\mathscr{g} = 0$. Each PDN layer outputs 3×8 channels, more than 24 channels than the preceding layer. The modest depth of the Conv layer can reduce the network parameter size and avoid gradient vanishing. Finally, the PDN block output is a feature map y_j with 24 output channels in the following equation:

$$y_j = \mathrm{Cat}\left(\left[y_{j,A}, y_{j,B}, y_{j,C}\right]\right), \tag{4}$$

where Cat represents the concatenation, and j is the j^{th} block.

4.1.2. Parallel Structure. The DenseNet transition layer receives the processed information from the last layer and all preceding blocks.

The parallel-DenseNet transition layer, which is similar to the DenseNet transition layer, is a transition connection between two PDN blocks. The DenseNet transition layer only receives the preceding layer features. Feature transmission is a serial way. Each DenseNet block and its following transition layer only address its features with a specific size and dimension. The PDN transition layer is not entirely like the DenseNet transition layer, which only compresses the feature depths and sizes. A close feedforward and parallel structure can concatenate the cross-layer features from all preceding blocks or layers for fusing learning (Figure 5). This series-parallel structure can also process multidimensional and multiscale dense features that fuse in each transition layer. The depths (*i.e.*, number of channels) of feature maps of the PDN transition layer are given in the following equation:

$$T_j = \mathrm{Conv}\left(\mathrm{Cat}\left(\left[\mathrm{Avg}_j\left(y_{j,A}, y_{j,B}, y_{j,C}\right), \mathrm{Avg}_{j-1}\left(y_{j-1,A}, y_{j-1,B}, y_{j-1,C}\right), \ldots, \mathrm{Avg}_1\left(y_{1,A}, y_{1,B}, y_{1,C}\right)\right]\right)\right), \tag{5}$$

where T_j is the j^{th} Transition layer output, $j = 1, 2, 3, 4$, Cat ($[j]$) represents the j^{th} PDN block output within it, and Avg represents the average pooling.

Each transition layer$_j$ output is more than 24 channels to the preceding transition layer$_{j\text{-}1}$, namely propagated rate ρ of 24. With the column convolutional kernel, the preprocessing layer and transition layers I, II, III, and IV are with outputs $(M \times \lfloor N/2 \rfloor \times 24)$, $(M \times \lfloor N/4 \rfloor \times 48)$, $(M \times \lfloor N/8 \rfloor \times 72)$, $(M \times \lfloor N/16 \rfloor \times 96)$, and $(M \times \lfloor N/32 \rfloor \times 120)$. If $N = 548$, $\lfloor N/32 \rfloor = 17$.

4.2. Concatenated-LSTM Framework. In this subsection, the focus of our work shifts to the feature correlations between the current frame and its adjacent frames. Unlike DenseNet, PDCL adds the LSTM layer before the linear layer. The LSTM processes RSFVG (the concatenated RSF vectors) by reshaping the 3-D feature map ($\mathbb{R}^{M \times N \times K}$) to 2-D ($\mathbb{R}^{M \times (N \times K)}$), namely, from $M \times \lfloor N/32 \rfloor \times 120$ to $M \times (\lfloor N/32 \rfloor \times 120)$. LSTM outputs an $M \times 120$-D feature map. Then, the following linear classification layer contains 4096 fully connected (FC) and attached SoftMax functions. Finally, PDCL outputs an $M \times 2$ matrix corresponding to the M vectors (frames) in the local temporal frame-group window to identify the genuine or forgery frame.

4.3. Loss Function. The loss function in equation (6) for training the model considers a sum of a concatenation of all frames in an RSFVG,

$$\mathrm{Loss} = \frac{\sum_{j=1}^{M} L\left(\mathrm{O}_j, \mathrm{GT}_j\right)}{M}, \tag{6}$$

where $L(*)$ function is a binary cross-entropy, O_j represents the final classification output of frame j or RSF vector j in an RSFVG, GT_j represents the ground-truth label for the vector j, $M = 2L_M + 1$ is the frame quantity in an RSFVG, and Loss is the loss function of total PDCL.

5. Experimental Results

This section first presents the existing SYSU-OBJFORG dataset,the proposed extension dataset and training strategies. Section 5.2 offers the evaluation metrics. Based on the different RSF dimensions, PDCL with different PDN block and transition layer quantities generates different PDCL derivative architectures. Section 5.3 presents the performance comparisons among the PDCL, derivative structures, and other schemes. Finally, Section 5.4 presents the experimental analysis and discussion.

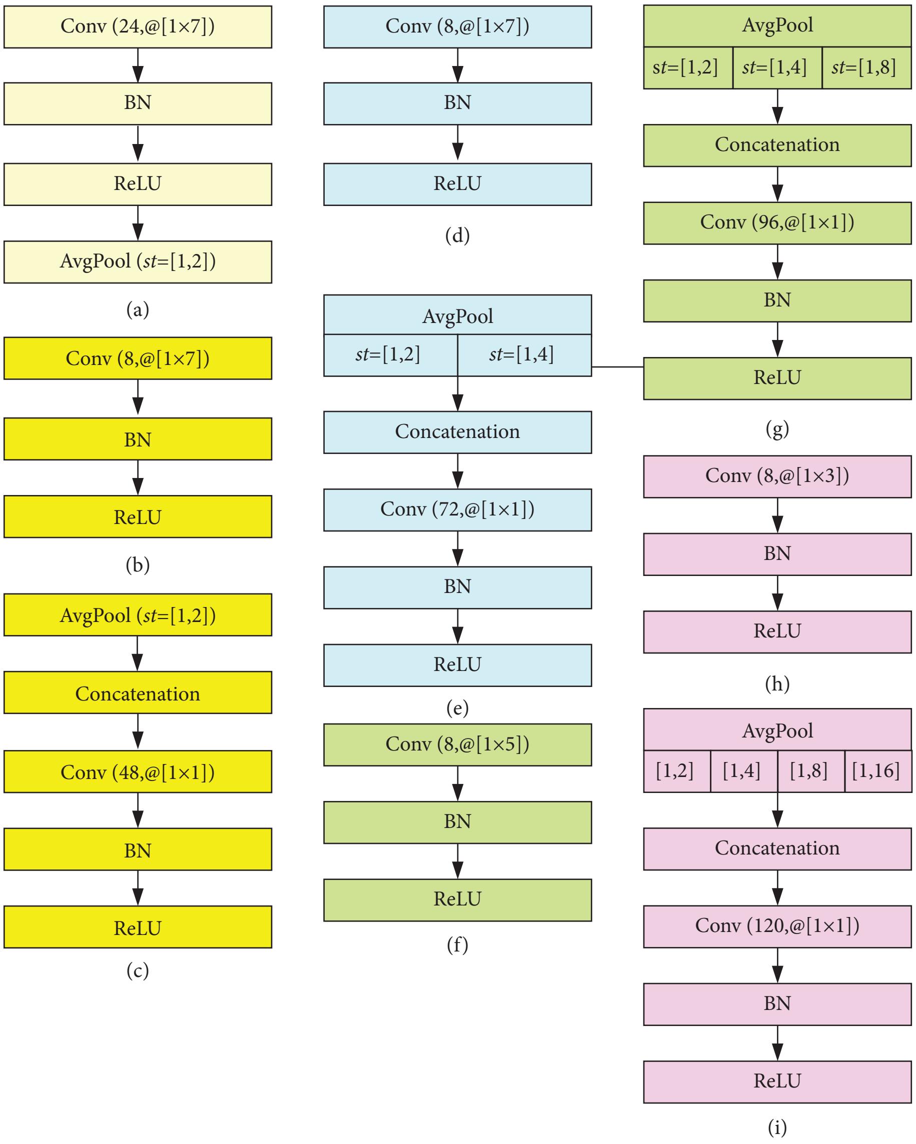


Figure 5: The detailed frame of preprocessing, PDN, and transition layers. (a) pre-processing layer, (b) Parallel-DenseNet layer I-A, (c) transition layer I, (d) Parallel-DenseNet layer II-A, (e) transition layer II, (f) Parallel-DenseNet layer III-A, (g) transition layer III, (h) Parallel-DenseNet layer IV-A, and (i) transition layer IV.

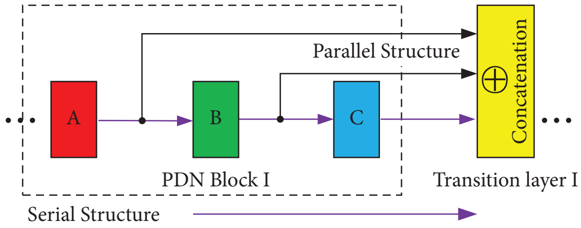


Figure 6: Serial and parallel structure in a PDN block and transition layer.

5.1. A New Training Dataset (Complete Dataset). VSOFD is a significantly challenging task in the visual surveillance field. Nevertheless, very few public datasets are designed specifically for VSOFD due to the complicated and tedious fabrication processing for such a huge amount of video information. SYSU-OBJFORG 1.0 is the largest publicly available VSOFD dataset containing 100 genuine and 100 forgery video clips. These forgery clips have been manipulated from genuine video footage shot by a static surveillance camera. The forgery video sources are encoded with MPEG-4/H.264 with 1280 × 720 frame size, 3 Mbits/s, 25 frames/s, and a forgery frame duration of 1–5 s. Based on the 100 genuine video clips, we have manipulated the other 100 forgery clips from the corresponding 100 genuine clips to create an SYSU-OBJFORG 2.0 dataset. The SYSU-OBJFORG 2.0 contains 100 genuine clips and 200 forgery clips. However, the average duration of detected forgery video clips is only about 11 s. Although the dataset contains 200 forgery video clips with a total of 58354 frames, it still does not provide sufficient forgery training samples for CNN + RNN.

To provide sufficient forgery samples, we have taken 400 genuine scenes with MPEG-4/H.264/AVI coding. The 400 genuine surveillance videos contain indoor and outdoor static scenes. A total of 108,400 frames from these 400 video clips are then manipulated to create the UM-OBJFORG 1.0 dataset with 400 forgery video clips containing object removal and object addition. The forgery video clips in UM-OBJFORG 1.0 are 1920 × 1080 frame size, 1.0 Mbits/s, 25 frames/s, and a forgery frame duration of 2–5 s. Figure 7 shows the samples of the UM-OBJFORG 1.0 dataset.

The complete dataset used for evaluation includes SYSU-OBJFORG 2.0 and UM-OBJFORG 1.0 datasets. As discussed in Section 3.3, each RSFVG has $M = 2L_M + 1$ frames, and the total number of frames in a detected video is W. The number of effective RSFVGs in every video clip is $W\text{-}(2L_M)$. The number of effective RSFVGs in SYSU-OBJFORG 2.0 and UM-OBJFORG 1.0 datasets (with 200 and 400 video clips, and 58354 and 108,400 frames, respectively) is $(58354 + 108{,}400) - [(200 + 400) \times (2L_M)]$, where $L_M \geq 3$. Table 1 shows the effective RSFVG and the corresponding frames in the complete dataset for $L_M = 3, 5, 7, 10$. Finally, the frames of the two datasets are split with a ratio of 8 : 1 : 1 for the training, validation, and testing stages.

5.2. Evaluation Criteria. The primary network classifier evaluation criterion is frame accuracy (*FACC*) in the following equation:

$$\text{Error} = (1 - \text{FACC}) = \frac{\text{incorrectly detected frames}}{\text{all the frames}}. \quad (7)$$

Lower error indicates better classifier performance. Furthermore, *precision*, *recall*, and F_1 [12] are the common criteria for the forgery forensics field.

$$\text{Precision} = \frac{\text{correctly detected forgery frames}}{\text{all detected forgery frames}},$$

$$\text{Recall} = \frac{\text{correctly detected forgery frames}}{\text{all the forgery frames}}, \quad (8)$$

$$F_1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}},$$

where F_1 is a comprehensive evaluation criterion, which is also the balance of *precision* and *recall*. Higher *precision*, *recall*, and F_1 indicate good performances. In summary, *error*, *precision*, *recall*, and F_1 together provide complete performance evaluations from various perspectives.

5.3. Performance Comparisons among the PDCL Derivative Structures and Other Schemes. Derivative structures are generated by considering different combinations of parameters and network structures with different model performances.

5.3.1. Comparisons of PDCL Derivate Based on Different Combinations of Ls and L_M. There are two parameters $Ls = 1$ or 2 and $L_M = 3, 5, 7$, or 10 for RS and RSFVG, respectively. Different combinations of $(Ls,\ L_M) \in (1,\ 2) \times (3,\ 5,\ 7,\ 10)$ determine the effectiveness of RSF extraction (*i.e.*, the CCPEV of 548-D features) and hence affect the network learning effectiveness and convergence speed. This work determines the optimal combination of (Ls, L_M) based on validation errors of the proposed PDCL network under different combinations of (Ls, L_M). From Figure 8, PDCL with $Ls = 1$ and all combinations of L_M (except $L_M = 3$) achieve the best validation $error < 3\%$ in the complete dataset.

5.3.2. Comparisons of PDCL Derivative Structures Based on Different RSFs. Different PDCL derivative structures ($\text{PDCL}_{\text{3-DCT}}$, $\text{PDCL}_{\text{8-SRMQ1}}$, $\text{PDCL}_{\text{9-SCRMQ1}}$, $\text{PDCL}_{\text{4-SPAM}}$) can be generated by employing different RSFs, such as DCT, SRMQ1, SCRMQ1, and SPAM, as illustrated in the following.

(1) $\text{PDCL}_{\text{3-DCT}}$: The steganalysis of DCT [23] has a low-dimensional feature (216-D). PDCL structure, as shown in Figures 4 and 6, the number of PDCL blocks, and transition layers are reduced from 4 to 3. As shown in Figure 5(g), it means that transition layers III output a feature of size $= M \times \lfloor N/16 \rfloor \times 96$ to the subsequent LSTM module.

(2) $\text{PDCL}_{\text{8-SRMQ1}}$ and $\text{PDCL}_{\text{9-SCRMQ1}}$: SRMQ1 [26] and SCRMQ1 [27] have high-dimensional features, *i.e.*, 12753-D and 18157-D, respectively. Under SRMQ1 and SCRMQ1, the number of PDCL blocks and transition layers is increased from 4 to 8 and 9, respectively.

(3) $\text{PDCL}_{\text{4-SPAM}}$: SPAM [25] (686-D features) has a similar dimension to the CCPEV of 548-D. Under SPAM, the number of $\text{PDCL}_{\text{4-SPAM}}$ blocks, and transition layers are 4.

The definite number of PDCL_B blocks follows the metrics $32 > \lfloor N/2^{B+1} \rfloor \geq 16$, where B is the number of PDCL blocks of the transition layer, and N is the RSF dimension. This definite number keeps the proximate feature dimension as the input of parallel-LSTM for performance comparison. Besides, the popular CNN (*e.g.*, Vgg16 [28], ResNet [29], and DenseNet [30]) replaces the proposed PDN to create different models (VGG16 + LSTM, ResNet + LSTM, and DenseNet + LSTM) for comparison. In Figure 9, the validation errors under $(Ls, L_M) = [(1, 5), (1, 7), (1, 10)]$ are better than that of $(Ls, L_M) = (1, 3)$. Therefore, the PDCL derivatives use the feature extraction framework $(Ls, L_M) = (1, 5)$ for the validation errors. As shown in Figure 9(a), the proposed $\text{PDCL}_{\text{4-CCPEV}}$ and $\text{PDCL}_{\text{9-SCRMQ1}}$ (PDCL baseline) achieve the best validation error among the derivatives. Figure 9(b) shows that the proposed PDCL structures are better than CNN + RNN structures. It means that the PDCL structure is much more suitable for VSOFD.

5.4. Experimental Analysis and Discussion. The performance between PDCL, its various derivative structures, and other SOTA schemes is evaluated in the complete dataset. The PDCL derivatives are based on different RSFs, including DCT, SPAM, CCPEV, SRMQ1, and SCRMQ1. All the PDCL derivatives use a combination of a short temporal frame window

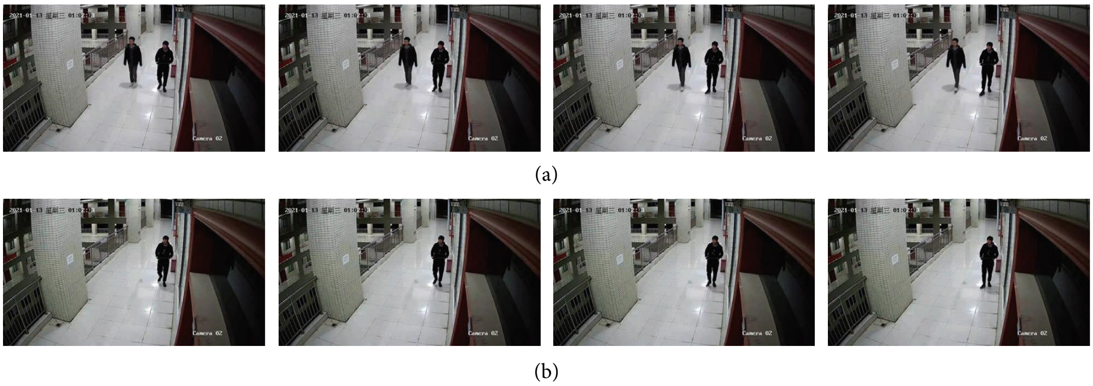


Figure 7: The genuine and forgery samples of the UM-OBJFORG 1.0 dataset: (a) video genuine frame samples and (b) video surveillance object forgery samples.

Table 1: The numbers of effective RSFVGS and the corresponding frame quantities in the complete dataset.

Complete dataset	$L_M = 3$	$L_M = 5$	$L_M = 7$	$L_M = 10$
No. RSFVGs and frames	163,154 × 7	160,754 × 11	158,354 × 15	154,754 × 21

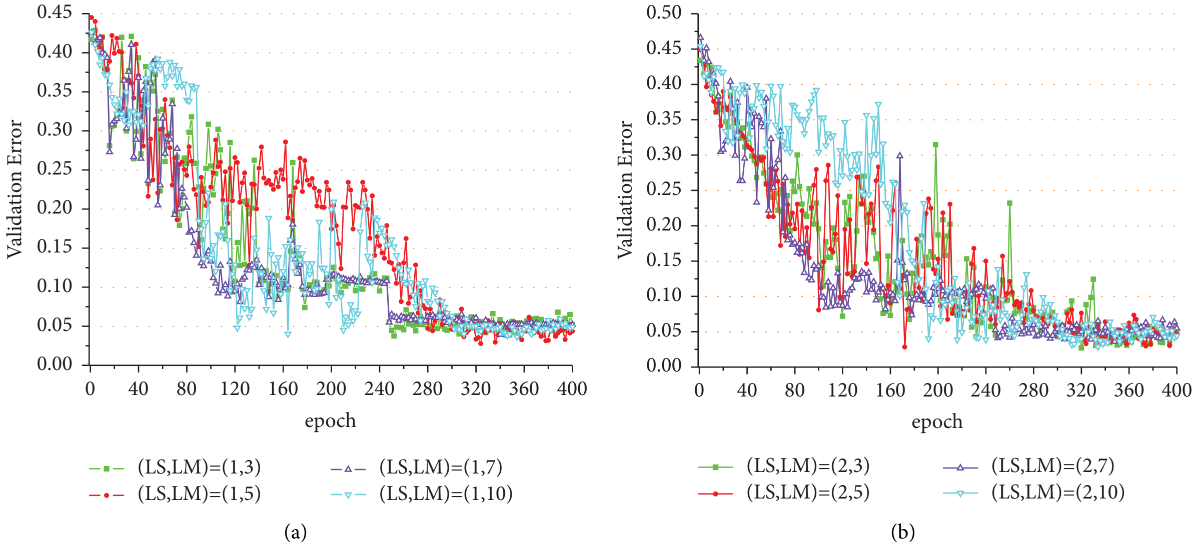
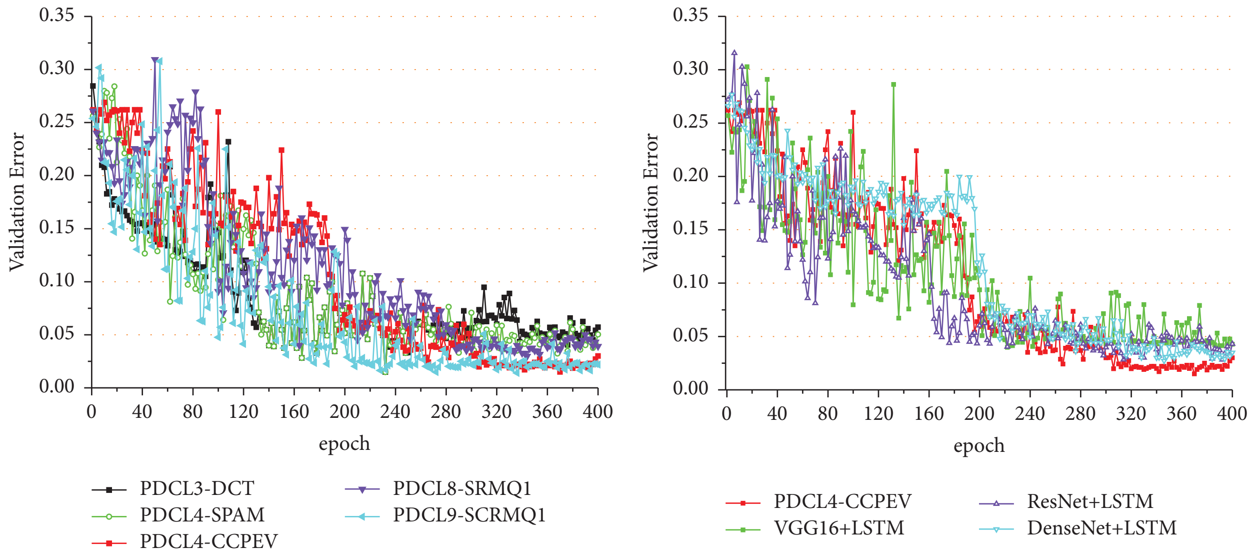


Figure 8: Validation errors of the PDCL derivate structures under different (Ls, L_M). (a) (Ls, L_M) = [(1,3), (1,5), (1,7), (1,10)]. (b) (Ls, L_M) = [(2,3), (2,5), (2,7), (2,10)]

$Ls = 1$ and the RSFVG window $2L_M + 1$ for various RSF extractions, contributing to the comprehensive performance comparisons. Nevertheless, there are few existing VSOFD methods and models. Therefore, some related video forensics methods are used to compare with the proposed PDCL, including hand-crafted techniques and the DNN model, *e.g.*, automatic identification and forged segment localization algorithm with CCPEV feature ($\text{AIFSL}_{\text{CCPEV}}$) [4], fast forgery detection with exponential Fourier moments (FFD_{EFMs}) [12], dense moment feature index, and best match algorithm with Radial-Harmonic-Fourier moments ($\text{DMFIBM}_{\text{RHFMs}}$) [13], Patch-Match with Polar Cosine Transformation (PM_{PCT}) [14], and Motion Residual and Parasitic Layers (MRPL) [19]. These methods can be used in video copy-move forgery detection [12–14], video splicing detection [19], and other forensics fields. Besides, Spatiotemporal Trident Networks (STNs) [18] are also compared in our work. Table 2 details the performance comparisons in the complete datasets. Figure 10 shows the visualization results of some VSOFD samples.

5.4.1. *Effect of RSF (Spatial-Temporal-Frequent Feature).*

The residual-based steganography feature (RSF) vector effectively extracts the implicit and unique features for classifying forgery video frames. From the results of Table 2, all the methods based on RSF achieve relatively good performance in *precision*, *recall*, and F_1. For example, the $\text{PDCL}_{9\text{-SCRMQ1}}$ and $\text{PDCL}_{4\text{-CCPEV}}$ models achieve the best performance scores of 90% in F_1, followed by the SRMQ1 with 88.42% in F_1. Among all the RSF, even the worst performing



(a)

(b)

FIGURE 9: The validation error comparisons: (a) among RSF features and (b) among PDCL$_{4\text{-CCPEV}}$ and other CNNs in the complete dataset.

TABLE 2: The performance comparisons on the dataset.

Method category	Individual systems with $Ls_{M1,5}$	Evaluation criteria (%)			
		Test error	Precision	Recall	F_1
PDCL derivates	PDCL$_{3\text{-DCT}}$	11.42	84.50	74.10	78.96
	PDCL$_{4\text{-SPAM}}$	11.06	80.43	81.12	80.77
	PDCL$_{4\text{-CCPEV}}$	5.90	88.56	**91.64**	90.08
	PDCL$_{8\text{-SRMQ1}}$	6.67	88.04	88.79	88.42
	PDCL$_{9\text{-SCRMQ1}}$	**5.49**	**91.46**	89.23	**90.33**
CNN	DenseNet$_{\text{CCPEV}}$	39.95	<10	<10	<10
CNN + RNN	VGG16 + LSTM$_{\text{CCPEV}}$	9.13	82.01	86.87	84.37
	ResNet + LSTM$_{\text{CCPEV}}$	8.50	84.58	87.96	86.24
	DenseNet + LSTM$_{\text{CCPEV}}$	7.94	86.98	88.85	87.91
	$Ls_{\ M1,7}$				
PDCL derivates	PDCL$_{3\text{-DCT}}$	11.42	84.42	74.01	78.96
	PDCL$_{4\text{-SPAM}}$	11.29	80.37	80.86	80.62
	PDCL$_{4\text{-CCPEV}}$	5.85	88.40	**91.94**	90.21
	PDCL$_{8\text{-SRMQ1}}$	6.55	88.12	89.53	88.88
	PDCL$_{9\text{-SCRMQ1}}$	**5.56**	**91.45**	89.22	**90.32**
CNN + RNN	VGG16 + LSTM$_{\text{CCPEV}}$	9.15	81.98	86.82	84.33
	ResNet + LSTM$_{\text{CCPEV}}$	8.51	84.55	88.00	86.24
	DenseNet + LSTM$_{\text{CCPEV}}$	7.97	86.90	88.82	87.85
	$Ls_{\ \text{M1,10}}$				
PDCL derivates	PDCL$_{3\text{-DCT}}$	11.85	84.42	73.78	78.69
	PDCL$_{4\text{-SPAM}}$	11.50	80.57	81.04	80.70
	PDCL$_{4\text{-CCPEV}}$	6.00	88.39	**92.10**	90.21
	PDCL$_{8\text{-SRMQ1}}$	6.77	87.91	89.51	88.70
	PDCL$_{9\text{-SCRMQ1}}$	**5.69**	**91.43**	89.18	**90.29**
CNN + RNN	VGG16 + LSTM$_{\text{CCPEV}}$	9.18	82.00	86.77	84.32
	ResNet + LSTM$_{\text{CCPEV}}$	8.52	84.56	87.90	86.20
	DenseNet + LSTM$_{\text{CCPEV}}$	7.92	87.00	88.89	87.93
SOTA methods with/without RSF	AIFSL$_{\text{CCPEV}}$ [4]	15.65	79.45	63.8	70.76
	FFD$_{\text{EFMs}}$ [12]	34.39	<10	<10	<10
	DMFIBM$_{\text{RHFMs}}$ [13]	30.47	<10	<10	<10
	PM$_{\text{PCT}}$ [14]	31.97	<10	<10	<10
	STNs [18]	10.23	84.02	80.93	82.44
	MRPL [19]	15.22	78.21	67.07	72.21

The bold values in the test error column are used to highlight minimum values. The bold values in precision, recall, and F1 column are used to highlight maximum values.

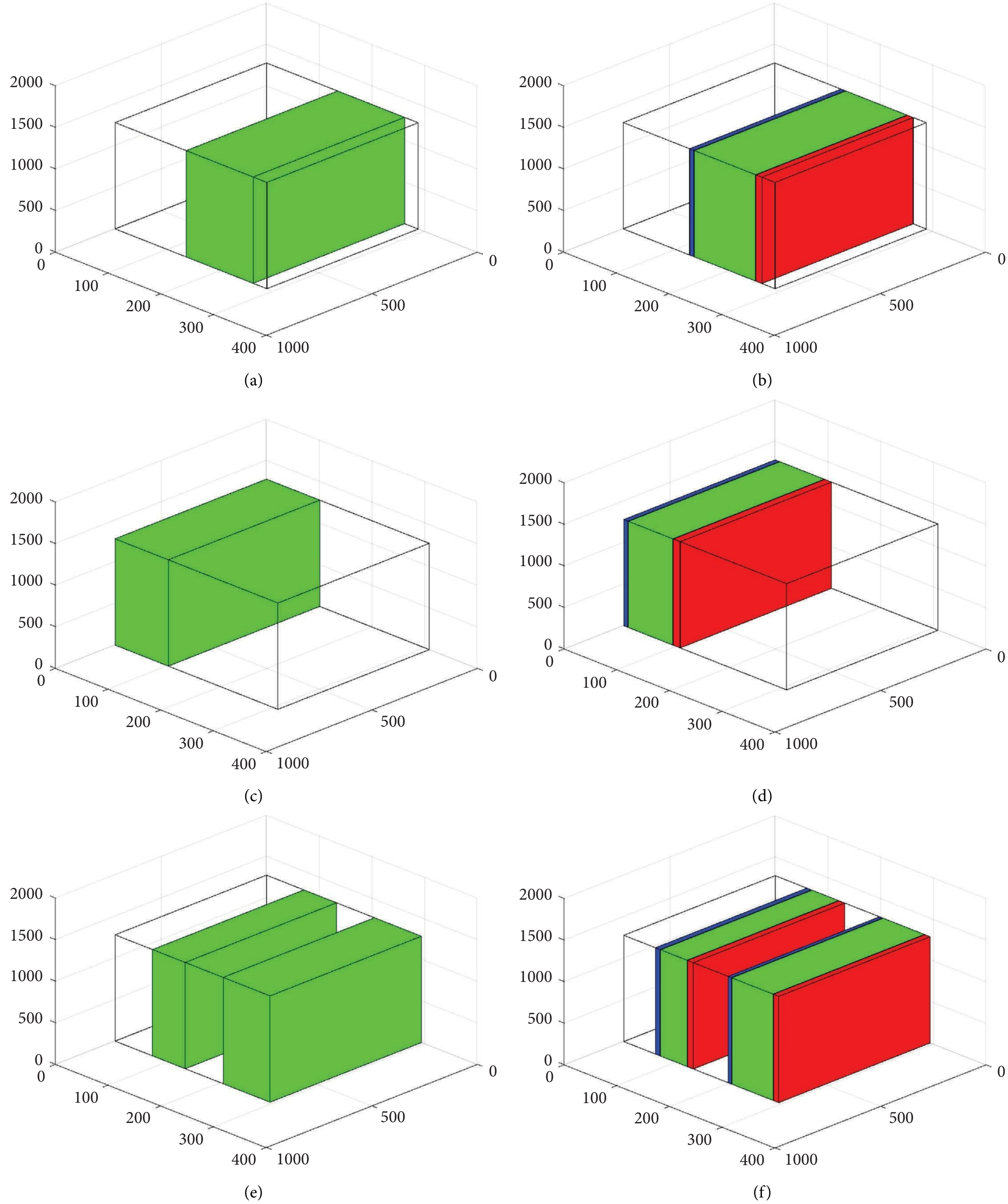


Figure 10: The visualization results of some VSOFD samples. In ground-truth (a, c, e) the forged frames are set to green. In the detection result (b, d, f), the forged frames correctly detected are set to green, the pristine frames detected as the forged frames are set to blue, and the forged frames detected as the pristine frames are set to red. (a) Groundtruth of forged video #1. (b) Detected result of forged video #1. (c) Groundtruth of forged video #16. (d) Detected result of forged video #16. (e) Groundtruth of forged video #85. (f) Detected result of forged video #85.

model PDCL$_{3\text{-DCT}}$ (*i.e.*, with DCT steganalysis feature) can obtain nearly 78.69% in F_1, which is already better than the MRPL method only with the residual signal as a processing feature. Besides, the CNN models (VGG16, ResNet, DenseNet) using the RSF also perform better than STNs without RSF. It can be explained that the RSF fusing spatial-temporal-frequent feature as video processing features and RSFVG structure can highly reflect the spatial-temporal-frequent coherence and the difference between frame contents. STNs are suitable for addressing the splicing forgery. However, STNs apply only 3 frames as a group to identify the forgery group/frame. The short temporal frame group is incompetent in detecting the forgery object of slow motion. For example, the $\mathrm{PDCL}_{1,3}$ in Figure 8(a) achieves a weaker performance than $\mathrm{PDCL}_{1,5}$ and $\mathrm{PDCL}_{1,7}$.

5.4.2. Effective Deep Network PDCL Architecture for VSOFD

(1) Effect of Novel Parallel-DenseNet (PDN) Structure. From the results of Table 2, the proposed PDCL with PDN structure achieves the overall best performances, including the lowest test error and the highest precision, recall, and F_1 scores compared to VGG16, ResNet, and DenseNet. DenseNet [30] is a kind of CNN with excellent performance in pattern recognition and image/video classification tasks. However, DenseNet is a feed-forward serial structure that can only process the concatenated features (in the same feature size). For this reason, DenseNet cannot simultaneously handle the coarse and fine features (in different sizes). From this perspective, our parallel-DenseNet (PDN) is proposed to address this issue by concatenating the serial and parallel features (i.e., cross-layer and cross-block features) from the preceding layers and blocks. In this way, the coarse-to-fine features can be simultaneously learned. The PDN structure (CNN module) in PDCL is compatible with different RSF and preserves the RSF independence of each frame with a column convolutional kernel. It is suitable for the following LSTM to process each frame's coherence and difference to learn the sequence's correlation and coherence. Therefore, the proposed PDCL derivates perform better than the $\mathrm{VGG16} + \mathrm{LSTM}_{\mathrm{CCPEV}}$, $\mathrm{ResNet} + \mathrm{LSTM}_{\mathrm{CCPEV}}$, and $\mathrm{DenseNet} + \mathrm{LSTM}_{\mathrm{CCPEV}}$.

(2) Effect of Concatenated-LSTM. A video is a time series, and its frames have temporal coherence, and each frame has its independence. CNN is powerful at handling spatial features but only accepts single-frame input. In this case, CNN cannot retain the coherence features of video frames, resulting in unsatisfactory detection accuracy. Although RNN can maintain frame correlation, it cannot handle spatial features. Therefore, a new architecture with both CNN and RNN capabilities is necessary. The PDN processes the spatial content of each frame while keeping frame invariance and independence. This way, the subsequent LSTM with long-short-term dependence can better focus on the temporal correlation among the adjacent frames. In Table 2, $\mathrm{DenseNet}_{\mathrm{CCPEV}}$ is much weaker than $\mathrm{DenseNet} + \mathrm{LSTM}_{\mathrm{CCPEV}}$ for addressing video surveillance object forgery. Similarly, the proposed PDCL derivates and $\mathrm{VGG16}+\mathrm{LSTM}_{\mathrm{CCPEV}}$, $\mathrm{ResNet} + \mathrm{LSTM}_{\mathrm{CCPEV}}$, and $\mathrm{DenseNet} + \mathrm{LSTM}_{\mathrm{CCPEV}}$ all achieve better performance than the common CNN-based MRPL. This verifies the effectiveness of LSTM in VSOFD.

In conclusion, the PDCL derivates with $(\mathrm{CNN} + \mathrm{RNN})_{\mathrm{RSF}}$ structures get much better scores than the CNN only, $\mathrm{CNN}_{\mathrm{RSF}}$ ($\mathrm{DenseNet}_{\mathrm{CCPEV}}$), $\mathrm{CNN}_{\mathrm{RS}}$ (MRPL), hand-crafted methods ($\mathrm{FFD}_{\mathrm{EFMs}}$, $\mathrm{DMFIBM}_{\mathrm{RHFMs}}$, $\mathrm{PM}_{\mathrm{PCT}}$). PDCL derivates also get better scores than other $\mathrm{CNN} + \mathrm{RNN}_{\mathrm{RSF}}$ structures using the same RSF. Among the RSF, PDCL with SCRMQ1 and CCPEV in $L_{sM1,5}$, getting the best F_1 scores of 90.33%, are the PDCL baseline in our dataset.

6. Conclusion

This paper proposes a new detection scheme for VSOFD with a novel spatial-temporal-frequent feature representation called RSFVG and a newly designed PDCL network, which aims to address the following critical issues:

(1) Through RSF, spatial-temporal-frequent features can be effectively represented with dimension reduction.

(2) Through the PDCL network, highly discriminative information in each frame (through CNN) and temporal correlation features between adjacent frames (through LSTM) can be learned simultaneously while maintaining frame independence. This is a critical property or requirement for identifying forgery frames in a video clip.

From the experimental results, the proposed scheme using the PDCL network with RSF can achieve high performance in test error, *precision*, *recall*, and F_1 scores. Among them, $\mathrm{PDCL}_{9\text{-}\mathrm{SCRMQ1}}$ achieves the best F_1 scores of 90.33% in the complete dataset, which is greatly improved by nearly 8% compared to the existing SOTA methods.

Data Availability

Due to privacy issues, the database of this paper is not published publicly. The data supporting the current study are available from the corresponding author upon request.

Disclosure

Yan-Fen Gan and Ji-Xiang Yang are considered co-first authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Yan-Fen Gan and Ji-Xiang Yang contributed equally to this work.

Acknowledgments

This work was supported under the Special Project in Key Fields of Guangdong University (grant number: 2022ZDZX1048); Guangdong University Engineering Technology Center (grant number: 2022GCZX013); Guangdong Basic and Applied Basic Research Foundation (grant number: 2020A151501783); Innovation Team of Scientific Research Platform of Universities in Guangdong Province (grant number: 2021KCXTD053); Guangdong Scientific Research Capacity Improvement Project of Key Construction Disciplines (grant number: 2021ZDJS058); the Innovative Young Talents Foundation in Higher Education of Guangdong Province (grant number: 2021KQNCX164).

References

[1] J. Zhang, C. Xu, Z. Gao, J. J. Rodrigues, and V. H. C. de Albuquerque, "Industrial pervasive edge computing-based intelligence IoT for surveillance saliency detection," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 5012–5020, 2021.

[2] Z. Gao, C. Xu, H. Zhang, S. Li, and V. H. C. de Albuquerque, "Trustful internet of surveillance things based on deeply represented visual co-saliency detection," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4092–4100, 2020.

[3] S. Jha, C. Seo, E. Yang, and G. P. Joshi, "Real time object detection and tracking system for video surveillance system," *Multimedia Tools and Applications*, vol. 80, no. 3, pp. 3981–3996, 2021.

[4] S. D. Chen, S. Q. Tan, B. Li, and J. W. Huang, "Automatic detection of object-based forgery in advanced video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 11, pp. 2138–2151, 2016.

[5] B. Üstübioğlu, G. Ulutaş, V. V. Nabiyev, M. Ulutaş, and A. Üstübioğlu, "A fast detection method for frame duplication forgery based on correlation," in *Proceedings of the 2017 25th Signal Processing and Communications Applications Conference*, pp. 1–4, Antalya, Turkey, May 2017.

[6] C. H. Feng, Z. Q. Xu, S. Jia, W. T. Zhang, and Y. Y. Xu, "Motion-Adaptive frame deletion detection for digital video forensics," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 12, pp. 2543–2554, 2017.

[7] S. Verde, L. Bondi, P. Bestagini, S. Milani, G. Calvagno, and S. Tubaro, "Video codec forensics based on convolutional neural networks," in *Proceedings of the 2018 25th IEEE International Conference on Image Processing*, pp. 530–534, IEEE, Athens, Greece, October 2018.

[8] C. S. Lin and J. J. Tsay, "A passive approach for effective detection and localization of region-level video forgery with spatio-temporal coherence analysis," *Digital Investigation*, vol. 11, no. 2, pp. 120–140, 2014.

[9] D. Davino, D. Cozzolino, G. Poggi, and L. Verdoliva, "Autoencoder with recurrent neural networks for video forgery detection," *Electronic Imaging*, vol. 29, no. 7, pp. 92–99, 2017.

[10] J. L. Zhong, Y. F. Gan, C. M. Vong, J. X. Yang, J. H. Zhao, and J. H. Luo, "Effective and efficient pixel-level detection for diverse video copy-move forgery types," *Pattern Recognition*, vol. 122, Article ID 108286, 2022.

[11] Y. Zhu, C. F. Chen, G. Yan, Y. C. Guo, and Y. F. Dong, "AR-net: adaptive attention and residual refinement network for copy-move forgery detection," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6714–6723, 2020.

[12] L. C. Su, C. H. Li, Y. C. Lai, and J. M. Yang, "A fast forgery detection algorithm based on exponential-fourier moments for video region duplication," *IEEE Transactions on Multimedia*, vol. 20, no. 4, pp. 825–840, 2018.

[13] J. L. Zhong, C. M. Pun, and Y. F. Gan, "Dense moment feature index and best match algorithms for video copy-move forgery detection," *Information Sciences*, vol. 537, pp. 184–202, Oct 2020.

[14] L. D'Amiano, D. Cozzolino, G. Poggi, and L. Verdoliva, "A patchmatch-based dense-field algorithm for video copy-move detection and localization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 3, pp. 669–682, 2019.

[15] Y. Rao, J. Ni, and H. Zhao, "Deep learning local descriptor for image splicing detection and localization," *IEEE Access*, vol. 8, Article ID 25611, 2020.

[16] J. H. Bappy, C. Simons, L. Nataraj, B. Manjunath, and A. K. Roy-Chowdhury, "Hybrid LSTM and encoder–decoder architecture for detection of image forgeries," *IEEE Transactions on Image Processing*, vol. 28, no. 7, pp. 3286–3300, 2019.

[17] V. Vinolin and M. Sucharitha, "Taylor-RNet: an approach for image forgery detection using Taylor-adaptive rag-bull rider-based deep convolutional neural network," *International Journal of Intelligent Systems*, vol. 36, no. 11, pp. 6503–6530, 2021.

[18] Q. X. Yang, D. J. Yu, Z. X. Zhang, Y. Yao, and L. Q. Chen, "Spatiotemporal trident networks: detection and localization of object removal tampering in video passive forensics," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 10, pp. 4131–4144, 2021.

[19] M. Saddique, K. Asghar, U. I. Bajwa, M. Hussain, H. A. Aboalsamh, and Z. Habib, "Classification of authentic and tampered video using motion residual and parasitic layers," *IEEE Access*, vol. 8, Article ID 56782, 2020.

[20] M. Aloraini, M. Sharifzadeh, and D. Schonfeld, "Sequential and patch analyses for object removal video forgery detection and localization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 3, pp. 917–930, 2021.

[21] X. Jin, Z. He, J. Xu, Y. W. Wang, and Y. T. Su, "Object-based video forgery detection via dual-stream networks," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 1–6, IEEE, Shenzhen, China, July 2021.

[22] X. L. Ding, G. B. Yang, R. Li, L. B. Zhang, Y. Li, and X. M. Sun, "Identification of motion-compensated frame rate up-conversion based on residual signals," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 7, pp. 1497–1512, 2018.

[23] Q. Z. Liu, "Steganalysis of DCT-embedding based adaptive steganography and YASS," in *Proceedings of the 13th ACM multimedia workshop on Multimedia and security*, pp. 77–86, Buffalo, NY, USA, September 2011.

[24] T. Pevny and J. Fridrich, "Merging Markov and DCT features for multi-class JPEG steganalysis," in *Proceedings of the SPIE, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, Article ID 650503, San Jose, CA, USA, March 2007.

[25] T. Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 2, pp. 215–224, 2010.

[26] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, 2012.

[27] M. Goljan, J. Fridrich, and R. Cogranne, "Rich model for Steganalysis of color images," in *Proceedings of the 2014 IEEE International Workshop on Information Forensics and Security (WIFS)*, Atlanta, GA, USA, December 2015.

[28] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, https://arxiv.org/abs/1409.1556.

[29] K. M. He, X. Y. Zhang, S. Q. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, Las Vegas, NV, USA, June 2016.

[30] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, IEEE, Los Alamitos, CA, USA, January 2017.

[31] A. Karpathy, "The unreasonable effectiveness of recurrent neural networks," *Andrej Karpathy blog*, vol. 21, p. 23, 2015.

[32] M. Nixon and A. Aguado, "Low-level feature extraction (including edge detection)," *Feature Extraction Mage Processing*, pp. 115–179, Elsevier, Linacre House/Jordan Hill/Oxford, 3rd edition, 2008.