electronics

MDPI

Review

A Contemporary Survey on Deepfake Detection: Datasets, Algorithms, and Challenges

Liang Yu Gong *,† **and Xue Jun Li** *,†

Department of Electrical and Electronic Engineering, Auckland University of Technology, Auckland 1010, New Zealand

* Correspondence: liangyu.gong@autuni.ac.nz (L.Y.G.); xuejun.li@aut.ac.nz (X.J.L.)

† These authors contributed equally to this work.

Abstract: Deepfakes are notorious for their unethical and malicious applications to achieve economic, political, and social reputation goals. Recent years have seen widespread facial forgery, which does not require technical skills. Since the development of generative adversarial networks (GANs) and diffusion models (DMs), deepfake generation has been moving toward better quality. Therefore, it is necessary to find an effective method to detect fake media. This contemporary survey provides a comprehensive overview of several typical facial forgery detection methods proposed from 2019 to 2023. We also analyze and group them into four categories in terms of their feature extraction methods and network architectures: traditional convolutional neural network (CNN)-based detection, CNN backbone with semi-supervised detection, transformer-based detection, and biological signal detection. Furthermore, it summarizes several representative deepfake detection datasets with their advantages and disadvantages. Finally, we evaluate the performance of these detection models with respect to different datasets by comparing their evaluating metrics. Across all experimental results on these state-of-the-art detection models, we find that the accuracy is largely degraded if we utilize cross-dataset evaluation. These results will provide a reference for further research to develop more reliable detection algorithms.

Keywords: deepfake detection; deep learning methods; transformer; semi-supervised learning; evaluating metrics; state-of-the-art models

Citation: Gong, L.Y.; Li, X.J. A Contemporary Survey on Deepfake Detection: Datasets, Algorithms, and Challenges. *Electronics* **2024**, *13*, 585. https://doi.org/10.3390/electronics13030585

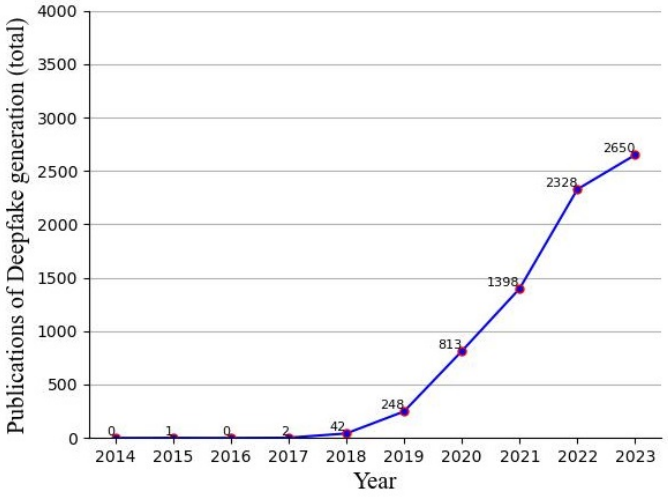
Academic Editors: Chiman Kwan and Beiwen Li

Received: 16 December 2023
Revised: 23 January 2024
Accepted: 29 January 2024
Published: 31 January 2024

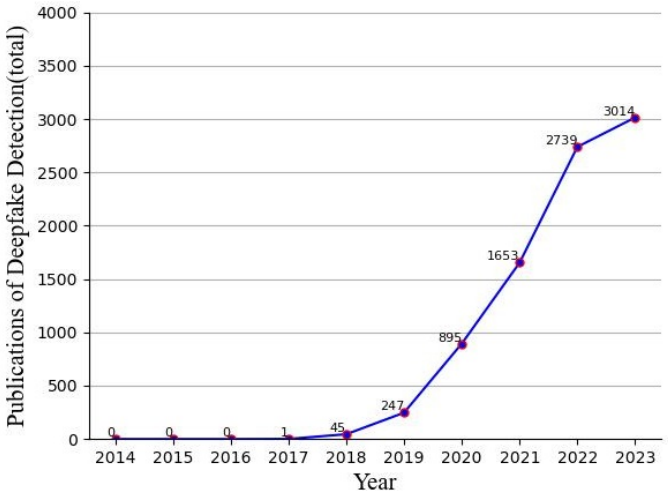
1. Introduction

Although deepfakes were initially associated with entertainment such as movie visual effects, camera filters, and digital avatars [1], they are defined as "believable generated media by Deep Neural Network" and have evolved into a mainstream tool for facial forgery. Their illegal applications now pose serious threats to social stability, national security, and personal reputation [2]. Facial manipulation technologies started with 3D landmark face swap and auto-encoders [3] to generate fake media; however, the trend of deepfake generation nowadays involves more powerful generative models such as generative and adversarial networks (GANs) [4,5] and diffusion models (DMs) [6] for creating more realistic counterfeit media. As for the illegal application of this technique, one Reddit user first released generated pornographic videos of actress Gal Gadot as the protagonist of deepfakes, which caused a huge sensation and harmed the victim's reputation at the end of 2017. In addition, Rana et al. [7] found that the top ten pornographic websites have released over 1790 deepfake videos to transfer celebrities' faces to porn stars' faces. To address these threats, deepfake detection and its performance have attracted significant consideration both in the academic and industrial fields; for example, Facebook, Microsoft, and Amazon jointly launched a Deep Fake Detection Challenge (DFDC) [8] on Kaggle from 2019 to 2020. Meanwhile, we compared the publications of deepfake generation with detection from Dimensions (See Figure 1), which is a scholarly database that goes

beyond research articles and their citations; then, we surprisingly noticed a shift in the publication trend, with deepfake detection surpassing deepfake generation in the past two years. Therefore, it is necessary to conduct in-depth research on deepfake detection methods for future investigators to study in order to prevent fraud through facial forgery or illegal information dissemination via deepfakes.



(**a**)



(**b**)

Figure 1. Relative publication data obtained from Dimensions database [9] at the end of 2023 by searching "deepfake generation" and "deepfake detection" as keywords: (**a**) number of deepfake-generation-related scholarly papers from 2014 to 2023; (**b**) number of deepfake-detection-related papers from 2014 to 2023.

Thus, a comprehensive literature review of deepfake detection will be useful for researchers to study this field further in different aspects with the development of deepfake generation. This motivates us to present a deepfake detection survey in review of (1) deepfake detection databases, (2) categorized several typical deepfake detection of frame-based and video-based methods, (3) the latest trend of detection methods (biological signal based), and (4) a summary and analysis of future trend of deepfake detection. Specifically, the main contributions of this survey are twofold.

1. We summarize and categorize detection techniques such as self-consistency, consistency-based detection, detection with vision transformer, contrastive learning for detection, inconsistency detection, biological signal-based detection, and a specially designed network (Capsule Network).
2. We perform an in-depth comparison in terms of evaluating the metrics of three state-of-the-art fake detectors: (i) Capsule Network, (ii) Consistency Learning Representation (CORE), and (iii) T-Face Detection Model.

2. Deepfake Detection Datasets

Most online face forgery tools (such as DeepFaceLive [10] and Roop [11]) are open source and do not require sophisticated technical skills, so using open-source software such as Basic DeepFake maker [12] is the main method for creating deepfake datasets. Due to multiple forgery methods, deepfake data are increasing at a very high rate of approximately 300% every year [2], but the data published online have different forgery qualities. This section introduces several representative datasets and illustrates their advantages and disadvantages.

2.1. FaceForensics++

FaceForensics++ [13] is a pioneering large-scale dataset in the field of face manipulation detection. The main facial manipulations are representative, which include DeepFakes, Face2Face, FaceSwap, FaceShifter, and Neural Textures methods, and data are of random compression levels and sizes [14]. This database originates from YouTube videos with 1000 real videos and 4000 fake videos, the content of which contains 60% female videos and 40% male videos. In addition, there are three resolutions of videos: 480p (VGA), 720p (HD), and 1080p (FHD). As a pioneering dataset, it has different quality levels of data and equalized gender distributions. The deepfake algorithms include face alignment and Gauss–Newton optimization. However, this dataset suffers from low visual quality with high compression and visible boundaries of the fake mask. The main limitation of this dataset is the lack of advanced color-blending processing, resulting in some source facial colors being easily distinguishable from target facial colors. In addition, some target samples cannot effectively fit on the source faces because there exists facial landmark mismatch, which is shown in Figure 2.

Figure 2. Several FaceForensics++ samples. The manipulated methods are DeepFakes (Row 1), Face2Face (Row 2), FaceSwap (Row 3), and Neural Textures (Row 4). DeepFakes and FaceSwap methods usually create low-quality manipulated facial sequences with color, landmark, and boundary mismatch. Face2Face and Neural Textures methods can output slightly better-quality manipulated sequences but with different resolutions.

2.2. DFDC

From 2020 to 2023, Facebook, Microsoft, Amazon, and research institutions put efforts into this field and jointly launched a Deep Fake Detection Challenge (DFDC) [8] on Kaggle to solve the problem of deepfakes presenting realistic AI-generated videos of people performing illegal activities, with a strong impact on how people determine the legitimacy of online information. The DFDC dataset is currently the largest public facial forgery dataset, which contains 119,197 video clips of 10 s duration filmed by real actors. The manipulation data (See Figure 3) are generated by deepfake, GAN-based, and non-learned techniques with resolutions ranging from 320 × 240 to 3840 × 2160 and frame rates from 15 fps to 30 fps. Compared with FaceForensics++, this database has a large-enough sample amount, different poses, and a rich diversity of human races. In addition, the original videos are from 66 paid actors instead of YouTube videos, and fake videos are generated with similar attributes to original real videos. However, the main drawback is that the quality level of data is different due to several deepfake generative abilities. Therefore, some samples have the problem of boundary mismatch, and source faces and target faces have different resolutions.

Figure 3. DFDC samples. We manually utilized InsightFace facial detection model to extract human faces from the DFDC. Although some of the samples are without color blending and with obvious facial boundaries, the average quality is a little higher than the first-generation deepfake datasets.

2.3. Celeb-DF V2

Celeb-DF V2 is derived from 590 original YouTube celebrity videos and 5639 manipulated videos generated through FaceSwap [15] and DFaker as mainstream techniques. It consists of multiple age, race, and sex distributions with many visual improvements, making fake videos almost indistinguishable to the human eye [16]. The dataset exhibits a large variation of face sizes, orientations, and backgrounds. In addition, some post-processing work is added by increasing the high resolution of facial regions, applying color transfer algorithms and inaccurate face masks. However, the main limitation of this dataset is the low data amount with less sample diversity because all original samples are downloaded from YouTube celebrity videos, and there is small ethnic diversity, especially for Asian faces.Here, we present a few samples of Celeb-DF V2 (see Figure 4).

Figure 4. Celeb-DF V2 crop manipulated facial frames. Except for transgender and transracial fake samples (Row 3), it is hard to distinguish real and fake images with the human eye.

There are other higher-quality deepfake datasets created by extensive application of the GAN-based method; for example, DFFD [17], which was published in 2020, created an entire synthesis of faces by StyleGAN [18]. Comparing datasets published after 2020 with previous datasets, it can be observed the data amount is much larger with multiple forgery methods such as GAN and forgery tools. In addition, the original data sources are not limited to online videos such as YouTube and also consist of videos shot by real actors. Thus, we predict the trend of future DeepFakes datasets to be larger scale with various forgery methods, multiple shooting scenarios, and different human races. We summarize the advantages and disadvantages of several commonly used datasets in Table 1.

Table 1. The typical and commonly used datasets of facial forgery detection.

Datasets	Real/Fake	Data Source	Methods	Advantages	Limitations
UADF (2018) [19]	49/49	YouTube	FakeApp	Early release	Low data amount
FaceForensics++ (2019)	1000/5000	YouTube	FS, F2F, NT, DeepFakes, and FS	Multiple methods	Visible manipulated artifacts
DeepFake-Detection (2019)	363/3068	Actors	DeepFakes	Relatively good effects	Low data amount
Celeb-DF (2019)	590/5639	YouTube	Improved DeepFakes	Realistic manipulation	Less forgery methods
DFDC (2020)	19,197/1,000,000	Actors	DeepFakes and GAN	Various techniques	Different quality levels
DeeperForensics-1.0 (2020)	50,000/10,000	Actors	DeepFakes	Large-scale with different attributes	Less forgery methods
iFakeFaceDB (2020)	494,414/33,000	Previous dataset	GAN	Multi-scenarios	Unknown
DFFD (2020)	58,703/240,336	YouTube and previous datasets	GAN, DeepFakes, and FakeAPP	Large-scale and multi-techniques	Different quality levels
FFIW10K (2021)	12,000/10,000	YouTube	DeepFaceLab, FSGAN, and FS	Multi-face and scenarios	Unknown

FS: FaceSwap; F2F: Face2Face; NT: Neural Textures; FS: FaceShifter.

3. Deepfake Detection

Different survey articles have categorized deepfake detection methods from different perspectives. For example, Rana et al. [7] group detection methods into four categories: deep-learning-based techniques, classical machine-learning-based techniques, statistical

techniques, and blockchain-based techniques. Nguyen et al. [20] directly categorized based on data type, i.e., video detection and image detection. These categorizations are relatively general and not beneficial for a deeper understanding of the latest deepfake detection algorithms and model architectures. In this section, we start from the perspective of data augmentation, feature extraction, and loss function and illustrate their unique model architectures to provide a new classification of deepfake detection methods. This review is a summary of some of the innovative algorithms at the current stage. The aim of this review paper is to study how a model can learn more manipulated information features. Thus, it analyzes more data augmentation methods, feature extraction backbones, and loss function designs, which will hopefully inspire researchers to study this field further.

3.1. Traditional CNN-Based Detection Methods

Traditional CNN-based methods [21] are widely applied to detect tampered images by frequency-domain features and statistical features. The CNN architecture can increase the learning capacity of a model by constantly stacking convolutional blocks and shortcut connections. Previous research has already proven that several models, such as Xception-Net and ResNet, can effectively extract fake features caused by compression and device fingerprints. Thus, we can find that most newly published deepfake detection methods choose these networks as the backbone for feature extraction. With the development of forgery techniques, manipulated media always come with advanced post-processing, which shows that deepfakes can be more realistic, and a detailed forgery clue is almost hard to distinguish from real ones. Thus, the similarity distances between extracted deepfake and real features are shorter than previously, and it is harder for a model to distinguish these samples by binary classification algorithms such as the support vector machine (SVM) because the extracted features are too concentrated on a specific area of the hyperplane. This is the biggest limitation of traditional CNN-based models for frame-based deepfake detection because the inconsistency between the frames is not considered at this stage. Usually, the unseen data come with different manipulation qualities, and the traditional CNN-based method cannot overperform on unlearned features. Thus, this category cannot provide good results in the latest manipulation and can easily overfit. In this section, we only focus on Capsule Network, which is one specially designed network for deepfake detection.

Capsule Network

Capsule Network [22] itself was not a new term in the field of deep learning, and it was first proposed to solve the problem that traditional convolutional neural networks (CNNs) have limited applicability in "inverse graphics". For classification work, traditional CNNs aimed to stack convolutional layers to extract multi-scale features corresponding to different receptive fields, where there is less consideration of the relationship between different feature information. However, Capsule Network (See Figure 5) has one typical strength: they are able to learn 3D spatial information about objects with their relationship and then model them explicitly. Moreover, Capsule Network [23] can use fewer parameters and data to perform similarly to CNN.

Specifically, Nguyen et al. [23] proposed a VGG-19 [24] feature extraction backbone, which is from the first layer to the third layer as the backbone and pre-trained in ILSVRC datasets [25], able to reduce overfitting and transfer learning. The input images are set to 300×300. The output features are passed through the backbone and sent to 10 primary Capsule Networks and output 4×1 vector capsules by a dynamic routing algorithm. Each primary Capsule Network [26] consists of two 2D CBL modules (convolution layer + batch normalization + ReLU), a statistical pooling layer and two 1D CB modules (1D convolution layer + batch normalization), and the overall CapsuleNet includes the parallel connection of ten primary Capsule Nets (see Figure 6). The parameters of convolutional blocks are shown in Figure 6 as well. Another difference between CapsuleNet and CNN is the output. Dynamic routing is designed to calculate how the output of lower-level capsules is allocated

to higher-level capsules, which is achieved by the squash activation function shown in Equation (1):

$$squash(u) = \frac{\|u\|^2}{1 + \|u\|^2} \frac{u}{\|u\|} \quad (1)$$

where the left side is the scale factor obtained by the extracted features, and the right side is the unit vector.

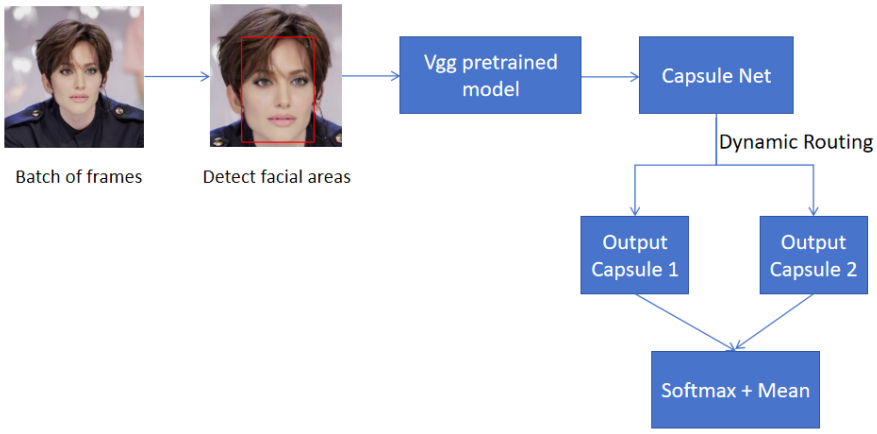


Figure 5. The architecture of CapsuleNet faces forensics detection. This method utilizes the CNN backbone to extract features and Capsule Network to output vectors for prediction.

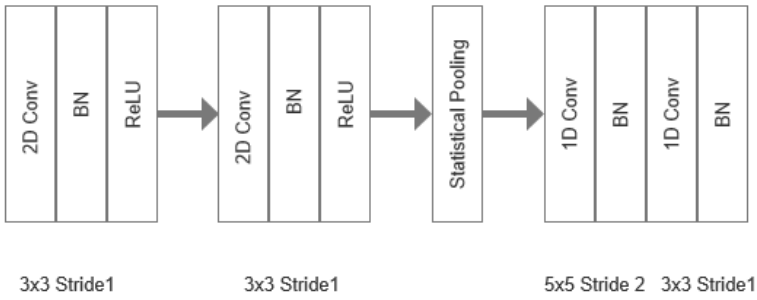


Figure 6. Details of the primary Capsule Network structure with relative parameters. Each Capsule Network includes the parallel connection of 10 primary capsules.

In summary, Capsule Network can study 3D spatial information relationships, which is better than the traditional CNN model, and it utilizes fewer parameters. However, its drawback is the weak generalization capacity that Capsule Network cannot perfectly predict unseen data. A detailed explanation is provided in Section 3.5.2.

3.2. CNN Backbone with Semi-Supervised Learning

The semi-supervised method is a model learning method that is between supervised learning and unsupervised learning, which means that the final decision is not only determined by the projection of features and labels. Compared with supervised learning methods, it requires certain data augmentations to enrich current datasets to data pairs and further similarity calculation on data. Most semi-supervised learning methods are achieved by three approaches, namely, context-based learning, temporal-based learning, and contrastive learning, and these approaches are widely applied in frame-based detection and video-based detection. In this section, we will mainly investigate four representative detection methods: temporal detection, consistency learning, inconsistency learning detection, and contrastive learning detection.

3.2.1. Consistency Representation Learning

COnsistency REpresentation Learning of Forgery Detection (CORE) [27] is an effective network that can capture different representations between two data augmentations and regularize feature similarity by the cosine distance to enhance consistency so that it can achieve relatively good results in both in-dataset and cross-dataset evaluation. We acknowledge that traditional classifiers always follow the steps of data augmentation, extracting and encoding features, and fully connected layers to achieve classification tasks. However, CORE presents a new architecture (see Figure 7) focusing on the consistency of sample pairs generated by different data augmentation strategies such as random resized crop and random erasing (RE) [28], for example. RE augmentation utilizes a scale factor from 0.02 to 0.2 and an aspect ratio from 0.5 to 2 to cut out a region in the face. For each input image, there is only a 0.33 probability without data augmentation, which ensures the input images are various. CORE proposes a shared parameter Xception encoder [29] that separately extracts features of data pairs, maps them into two-dimensional representation vectors in the latent space, and then calculates their pixel-wise consistency loss. In addition, cosine consistency loss can avoid forcing two representations to be totally the same. Finally, two representations obtained from the backbone are fed to two classifiers trained by cross-entropy loss as well and penalize the distances between two features adopted by cosine loss. The total loss is guided by the linear combination of cosine loss and cross-entropy loss, which is calculated by

$$L_{cos} = (1 - f_1^n \cdot f_2^n)^2 \tag{2}$$

$$L_c = \sum_{1}^{N} L_{cos} \tag{3}$$

$$L_{loss} = L_{CE} + \alpha L_c \tag{4}$$

where f_1^n and f_2^n represent the normalization of Feature 1 and Feature 2, respectively; L_{CE} represents the cross-entropy loss; α represents the balance weight for cross-entropy loss and consistency loss; and N represents the pairs of data augmentation.

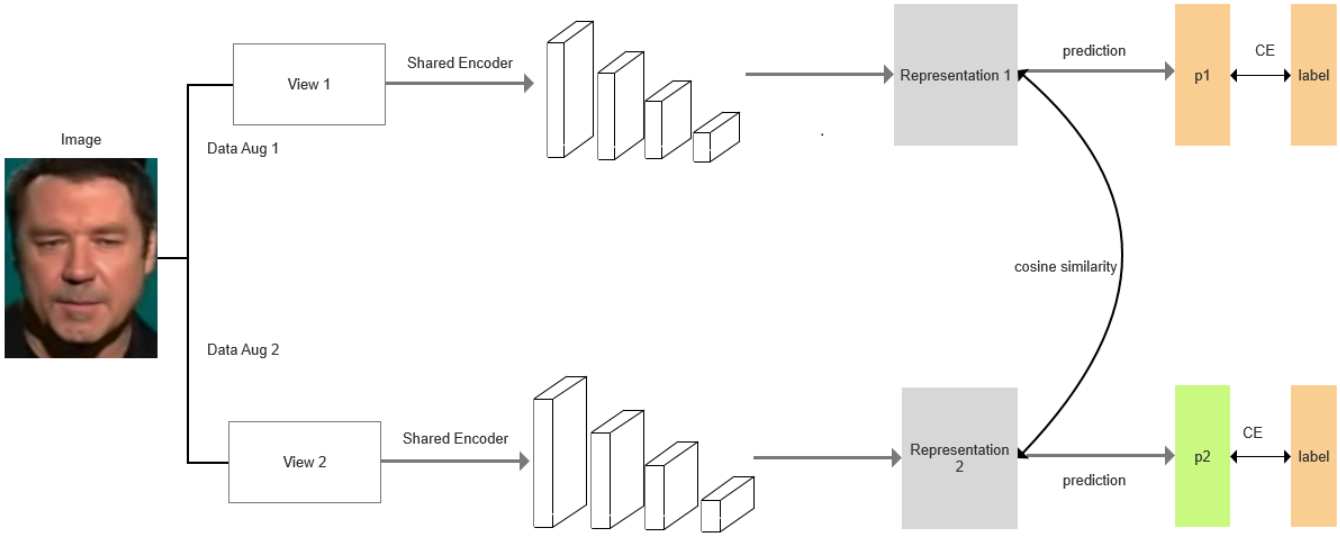


Figure 7. The architecture of CORE. One method extracts pairs of representations by data augmentation and calculates consistency loss to guide final loss function.

Thus, CORE is used to observe the similarity between two views' representations obtained from data augmentations and feature extraction because we regard that each type of data should be consistent even though applying different data augmentations. The main strength of this method is to utilize a new loss function that the classification work is determined by both cross-entropy loss and representation similarity after different data

augmentations. It largely increases the generalization ability of previous detection methods because it shows better evaluating metrics on cross-dataset experiments. However, the performance of this method largely relies on the data augmentation stage; thus, different augmentations will influence the final evaluating metrics.

3.2.2. Self-Consistency Method

Most defense methods focus on detecting suspicious artifacts of fake media such as eye blinking, blending boundaries [30], and face warping [31]. However, the manipulated clues can be observed in the feature patches as well. Zhao et al. [32] proposed a method for finding out the inconsistency of content-independent and spatially local information within manipulated images by designing a multi-task learning architecture, as shown in Figure 8, including the consistency branch and classification branch. More specifically, it regards the manipulated images to contain different source features, which come from the imaging pipeline (photo-response non-uniformity [33]) and encoding approaches (JPEG compression pattern [34]) at different locations. Thus, this is a detection method mainly based on calculating the similarity between different source feature patterns by pairwise self-consistency learning similarity score in the consistency branch. The backbone selects ResNet-34 [35] initialized by pre-trained weights on ImageNet for extracting pre-processed video frames' features. Then, the feature map ($H \times W \times C$) is divided into several patches, and each patch is compared against the rest of the patches to compute dot product similarity (consistency score):

$$s(f_i, f_j) = \delta\left(\frac{\theta(f_i)\theta(f_j)}{\sqrt{C}}\right) \tag{5}$$

where f_i and f_j represent patches of the feature, δ represents the sigmoid function, C represent channel size, and θ represents the embedding function.

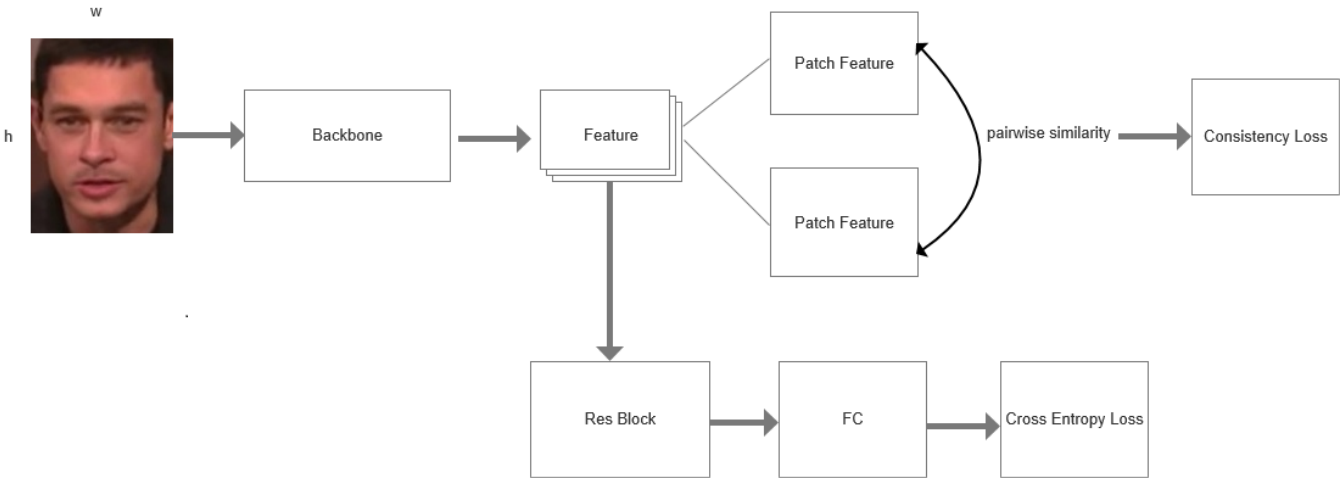


Figure 8. The architecture of the consistency branch and classification branch [32], One method calculates the patch similarity and classification loss to guide the final loss.

By iteration of this process, finally, a 4D consistency volume will be obtained, and ground-truth 4D volume is created by bi-linear downsampling and computation of element-wise differences, where 1 denotes patches are consistent and 0 otherwise. In addition, the inconsistency image generator utilizes elastic deformation to improve various masks so it can eliminate spurious correlations. Then, it randomly selects the color-blending method to improve feature robustness as well as data augmentation such as JPEG compression, Gaussian noise, and color jittering. The pairwise consistency learning (PCL) loss uses binary cross-entropy loss (BCE Loss) to supervise:

$$L_{PCL} = \frac{1}{N}\sum BCE(V_{pred}, V_{GT}) \tag{6}$$

In short, this method also utilizes two factors to determine the final loss function. However, the difference is to find the patches' similarity and not the consistency between different representations because it is based on the notion that the forgery clues of fake video generation will project to different feature patches. In addition, it requires performing ablation experiments on four datasets, including DFR, CD2, DFDC, and DFDC-P, and the average area under the curve (AUC) reached 82% above, which is a significant development in forgery detection. However, there are two limitations: this method cannot detect entire facial synthesis created by GAN or diffusion models, and it can be further improved on low-quality data.

3.2.3. Dual-Contrastive Learning Detection (T-Face)

Recently, Youtu Lab [36] proposed a dual-contrastive learning architecture that is aimed at distinguishing the authenticity of faces by different data augmentation. We found out that traditional detection methods like CNN-based methods are not suitable for detecting and learning generalized feature representations. Dual-contrastive learning (DCL) includes inter-instance contrastive learning and intra-instance contrastive learning for narrowing the distance between embeddings of the same class and widening the distances between non-homogeneous embeddings. Unlike traditional data augmentation, T-face consists of a data views generation (DVG) module with four data enhancements to eliminate task-irrelevant contextual factors—random patch, high-frequency enhancement [37], frame shift, and corresponding mix-up, which are selected with certain probability—and randomly combine with traditional data augmentation to generate two data views, V_1 and V_2. The details of four data augmentation factors are shown in Table 2.

Table 2. Details of four data augmentation factors.

Data Augmentation	Details	Purposes
Random patch	Divide facial area into $k \times k$ patches and randomly shuffle and reassemble them	Destroy facial structures
High-frequency enhancement	Features through SRM model and combine with original image	Boost generalization ability
Frame shift	Select multiple frames from one video	Reduce the influences of motions and expressions
Corresponding mix-up	Associate fake images with corresponding real images	Eliminate some obvious forgery clues

The architecture of DCL (see Figure 9) includes two branches of feature extraction encoders (f_q and fk); then, the batch images are fed to encoders to obtain extracted features by applying 1×1 convolution operation to squeeze the channel to obtain the query q and the key k separately. The encoders utilize the strategy of the exponential moving average. The parameters of the key encoder are updated from query encoder parameters. It promotes generalized feature learning by cross-entropy loss, InfoNCE loss [38], and intra-instance contrastive loss together. For the performance of T-face, Sun et al. [36] conducted a cross-dataset experiment on FF++ (trained), DFD, DFDC, Wild DeepFake, and Celeb-DF. The AUC metric and equal error rate (EER) presented relatively good results on Celeb-DF and DFD, which are shown in Table 3.

Table 3. Experiments of DCL [36] in different datasets.

Datasets	AUC	EER
FF++	99.30%	3.26%
DFD	91.66%	16.63%
DFDC	76.71%	31.97%
Wild Deepfake	71.14%	36.17%
Celeb-DF	82.30%	26.53%

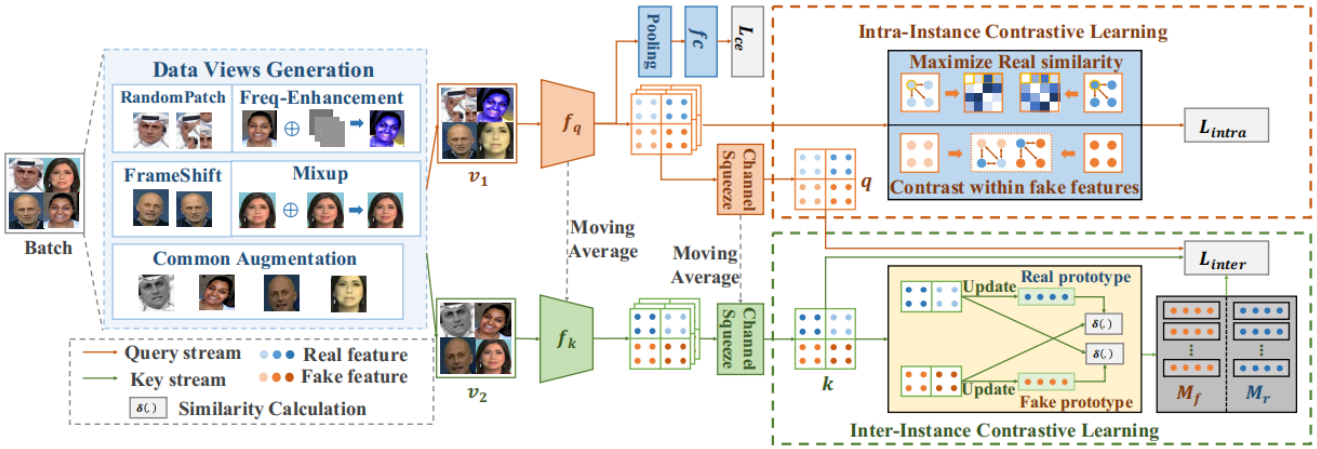


Figure 9. An overview of DCL architecture. Reprinted with permission from Ref. [36]. 2024, K.Sun. Four random data augmentation factors were utilized in this contrastive learning method to guide inter-class and intra-class distances separately.

3.2.4. Spatio-Temporal Inconsistency Learning

Video-based detection such as LSTM [39] cannot achieve better results than the above frame-based detection methods in the current stage, but learning temporal artifacts has already been proven to increase detection robustness. Thus, Gu et al. presented a detection work [40] based on learning fake videos' spatial and temporal inconsistency. To our knowledge, it is quite difficult to distinguish one frame as real or fake because of the developed manipulation techniques; however, if we combine frames and observe their temporal inconsistency such as face position jittering, detection will be easier. A novel STIL block (see Figure 10) works in a two-stream manner and contains a temporal inconsistency module (TIM), a spatial inconsistency module (SIM), and an information supplement module (ISM), which are proposed to obtain a representation with the spatial and temporal information.

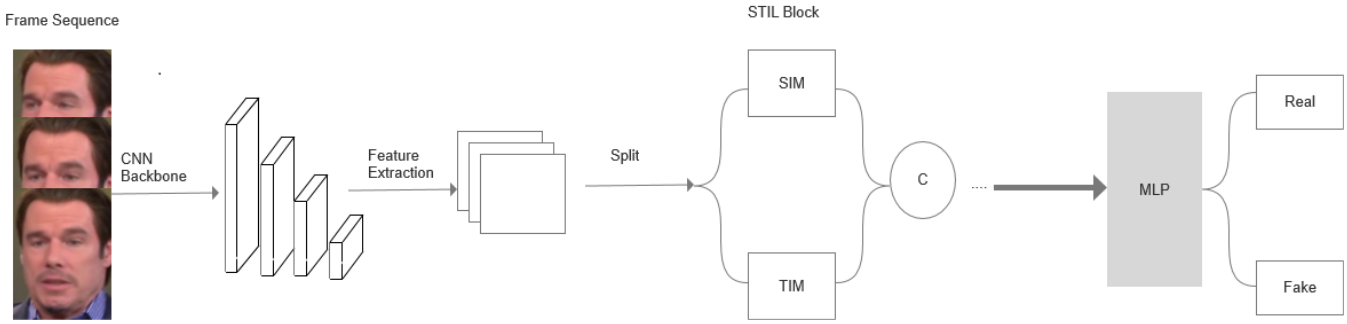


Figure 10. The structure of STIL [40]. Each STIL block contains SIM, TIM, and ISM modules.

Firstly, the input sequence $[T, C, H, W]$ was split along the channel dimension into two portions, where each portion is a feature with a size of $[T, \frac{C}{2}, H, W]$. Then, two portions were separately fed as the inputs of TIM and SIM to acquire two inconsistency representations from spatial and temporal perspectives. Specifically, SIM and TIM were designed as three branch modules, as shown in Tables 4 and 5, which aimed to find pixel-wise boundary mismatch and temporal inconsistency. Based on the ablation experiments, STIL proposed that the best performance occurs when we fuse SIM modules into TIM modules.

TIM modules utilize temporal difference calculation, which is the subtraction of adjacent frame features.

$$s_t^h = Conv1(x_{t+1}^h - x_t^h) \tag{7}$$

where $Conv1$ is 3×1 convolutional layer, and x_{t+1}^h and x_t^h are features of $t+1$ times of frame and t times of frame, respectively.

Table 4. SIM branches.

Branches	Operations
Upper branch	ResNet block for shortcut connection
Middle branch	Downsampling, utilize 1×3 and 3×1 convolutional layer to obtain vertical and horizontal features and upsampling
Confidence calculation	Fuse the upper and middle features to obtain confidence by sigmoid
Bottom branch	3×3 convolutional layer and multiply with confidence

Table 5. TIM branches.

Branches	Operations
Upper branch	Convolution and reshape, temporal difference calculation, and vertical temporal inconsistency enhancement
Middle branch	Convolution and reshape, temporal difference calculation, and horizontal temporal inconsistency enhancement
Bottom branch	ResNet block for shortcut connection

3.3. Transformer-Based Detection

Although the transformer network was first designed for learning long contextual sequence information and solving natural language tasks such as machine translation, Dosovitskiy et al. [41] showed that it is not necessary to highly rely on CNN to perform image classification tasks, and a pure transformer applied to image patch embeddings can perform better on classification as well. Vision transformer (VIT) divides an image into several flattened patches and embeds them into patch sequences first; then, it applies an attention mechanism to obtain each patch's attention weight and extract features. This means that the transformer framework on classification is gradually replacing traditional CNN classification even though there is limited exposure to deepfake detection. For example, there is a video transformer with the incremental learning approach [42] published in 2021, which utilizes Xception [29] as the backbone and 12 transformer blocks to learn feature correlations. In addition, most studies are still using CNN as a feature extractor and transformer block to classify deepfake images such as the convolutional vision transformer [43]. In this section, we introduce two transformer-based methods, which are the end-to-end transformer architecture and the video transformer.

3.3.1. End-to-End Transformer Detection

Most transformer-based deepfake detection only utilizes transformer blocks in downstream tasks, such as studying the correlation relationship after extracting features by the CNN-based backbone. However, the community of vision transformer classification lacks an end-to-end transformer detection framework for deepfake detection. DFDT [44] proposes an end-to-end deepfake detection framework (see Figure 11) using vision transformers that can basically solve the problem that traditional CNN cannot obtain the correlation between spatial patches and the information loss caused by the receptive field. It comprises four basic components: patch embedding [45], multi-stream transformer block, attention-based patch selection, and multi-scale classifier. After the pre-processing of videos, each video selects 20% video frames as the input to focus on facial regions only. In addition, the two-stream transformer block is designed to detect forgery clues in different facial regions by dividing input images into different patch sizes. The low-level patch branch processes smaller facial regions, such as lips, and the high-level patch branch is for extracting larger facial manipulated features, such as boundary mismatch. Then, the patch embedding is different from vision transformer's embedding because it realizes that non-overlapping feature extraction, where the sliding windows' stride equals the patch size, will harm the neighboring local structures. In other words, this method chooses overlapping patch extraction where two adjacent patches share a specific area, and it will preserve neighboring information better. Similarly, with a class token added, the patches

flatten the positional embedding and project it into a latent space. A patch selection mechanism based on attention weight is applied in the second module, which can pay more attention to sensitive information and dismiss the less useful information in the training phase. Once the two-stream transformer blocks output the feature with the patch attention weight, they will both make initial predictions. The average of all the predicted results in the final decision.

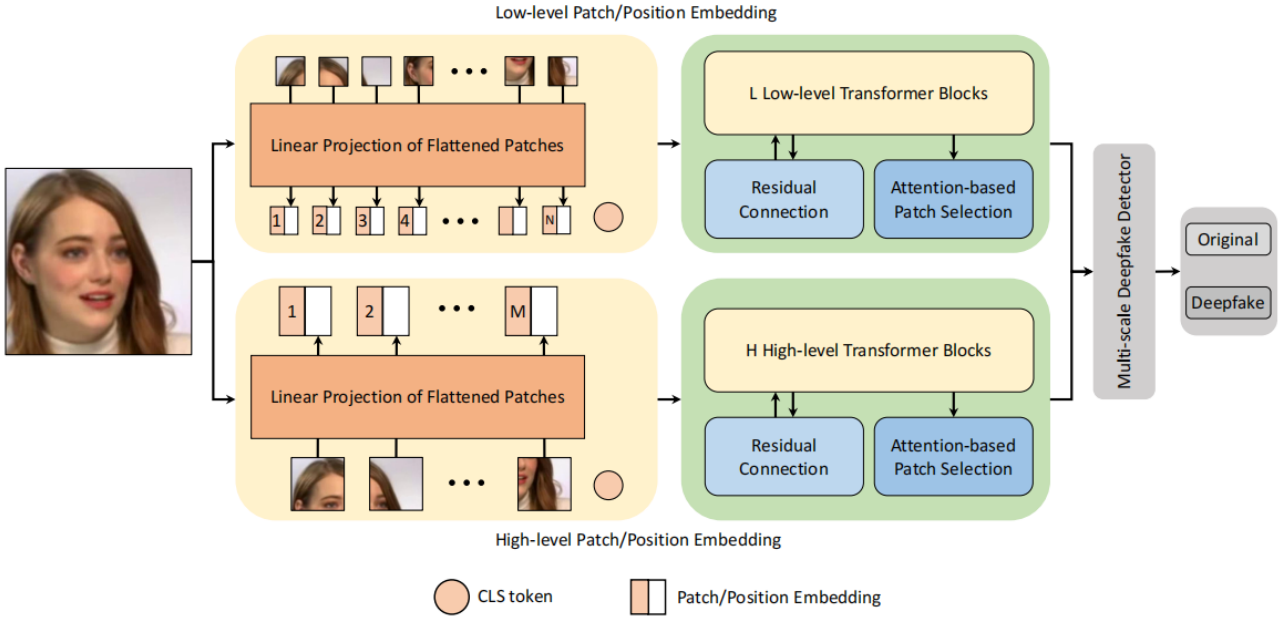


Figure 11. An overview of DFDT framework for deepfake detection [44]. It includes overlapping patch embedding, patch selection mechanism, multi-scale transformer block, and classifier.

In summary, this detection method proposes an end-to-end vision transformer framework instead of utilizing CNN to extract features and choosing VIT to perform downstream classification. DFDT can capture different scales of alterations and achieve SOTA results both in in-dataset experiments and cross-dataset experiments, which means it has better generalization ability and strong effectiveness, and the transformer can be a better feature extractor than most CNN backbones.

3.3.2. ISTVT: A Video Transformer for Deepfake Detection

Although vision transformers have made significant progress in several vision classification tasks, research on video transformers is scarce. In detail, the video transformer is a video-based detection method, which means it processes multiple video frames at the same time and applies the self-attention mechanism on different token dimensions. Following this inspiration, Zhao et al. [46] considered a video transformer that can jointly study spatial and temporal information in fake videos and has better performance and generalization in the deepfake detection field. ISTVT (see Figure 12) consists of a spatial–temporal self-attention module and a self-subtract module to capture spatial features and temporal inconsistency in videos. Specifically, the proposed video transformer first utilizes Xception as a backbone to extract features in video sequences; then, it splits these features by flattening them to obtain spatial, temporal, and class tokens, which can be fed to corresponding transformer blocks as input. The tokens require additional pre-processing named the self-subtract mechanism [46] to reduce the adjacent tokens in temporal dimensions. It is used for generating feature residuals to ignore redundant information and focus on temporal inconsistency. Similar to the vision transformer, the tokens are projected to Q, K, and V by a linear projection layer with shape $(T+1) \times (HW+1) \times C$, and a self-attention mechanism is applied both in the spatial dimension and the temporal dimension. A temporal attention block is inserted after each spatial attention block, and the relative position

embedding [47] applied on each temporal attention block is used to distinguish the order of frames. Finally, the MLP block is selected for the final classification task.

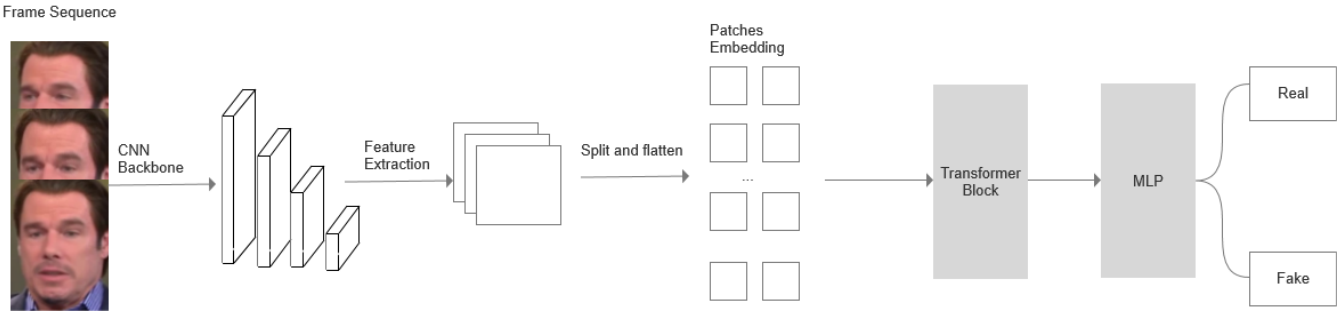


Figure 12. An architecture of ISTVT [46]. It consists of four basic components: backbone feature extraction, token embedding, self-attention blocks, and MLP.

The performance of cross-dataset experiments was tested, and the accuracy rates of Celeb-DF, DFDC, and FaceShift were 84.1%, 74.2%, and 99.3%, respectively. With the increased accuracy results, it is proved that the transformer with spatial–temporal inconsistency detection reached higher generalization in unseen data than previous video-based detection methods.

3.4. Biological Signal Detection

Although the above detection methods are relatively sophisticated and commonly applied in the deepfake detection field, analyzing forgery artifacts in biological signals has become a new trend that has attracted much attention. These methods focus on capturing fake information by combining biological signals with vision artifacts that can achieve better results or multi-modality detection. This section will only briefly introduce two methods based on PPG cells and physiological signals. These methods are still developing with good potential but always require specific matching biological maps such as heart rate.

Umur et al. [48] proposed an approach to extract PPG cells from both real and deepfake videos and designed a classification network to distinguish authenticity and related forgery methods. Specifically, this method proposed to first extract raw signals from real videos and fake videos from different facial locations by windows, and then encode these signals along with spectral density into a spatio-temporal block to create PPG cells. Thus, the model input is no longer a batch of video frames instead of PPG cells, which includes the projection of biological signals. Comparing the data pre-processing method with other video-based detection methods, it is reasonable to believe that the residuals learned by CNN with PPG biological information will obtain a better benchmark result, and the spatial and temporal patterns of biological signals can be conceived and projected to residuals [48]. The experimental accuracy finally reached 97.29% in detecting fake videos and 93.39% in their related generative models by setting a window length of 64. Moreover, remote photoplethysmography (rPPG) can extract heartbeat signals from recorded videos. Wu et al. [49] realized that facial manipulation progress is inevitable with sudden facial color changes in some periods, and rPPG could become an indicator for deepfake detection. Therefore, it is necessary to obtain a multi-scale PPG map for classification, and the accuracy of this method can reach more than 99% in the FF++ in-dataset experiment, which is a large increase compared to Capsule Net and other CNN-based deepfake detectors.

In summary, the biggest contribution of this method is that it leads deepfake detection toward source detection by adding biological signals to classify the residuals of generative models. It also provides comparison experiments to illustrate that the accuracy will increase by 47% by adding PPG cells with the same VGG-19 backbone, even outperforming the Xception Network by more than 10%. These significant experimental results prove that there is potential to add PPG signals to enhance the diversity of deepfake features with better classification abilities. However, the selection of window size will influence the

stability and representative power of PPG cells because a small window will miss PPG frequencies, and a too-large window will include more noise.

Physiological signals can be utilized to classify deepfake videos, and they can also combine multi-modal information; for example, Stefanov et al. [50] proposed a method (See Figure 13) to extract physiological signals and utilized the graph convolutional network (GCN) [51] to fuse video and physiological signal and detect the dissimilarity between audio and video modality. In particular, this method proposes an intriguing algorithm to obtain visual physiological signals according to the following steps: first, facial areas are detected with alignment, and background areas are removed; then, they are passed through MTTS-CAN [52] and a square occlusion patch to estimate heart rate and respiration rate, respectively. Finally, the difference between the estimated signal with occlusion and without occlusion is considered as the relative contribution of the mask-out region for the two physiological signals. In addition, one graph-based model is designed to fuse the facial information with a visual physiological map. This information is used as nodes with previously extracted features by ResNet18. Each feature is connected with the physiological map and calculates the cosine similarity. By training the GCN model and ResNet18, the model can combine two modalities and generate visual representations with physiological signals.

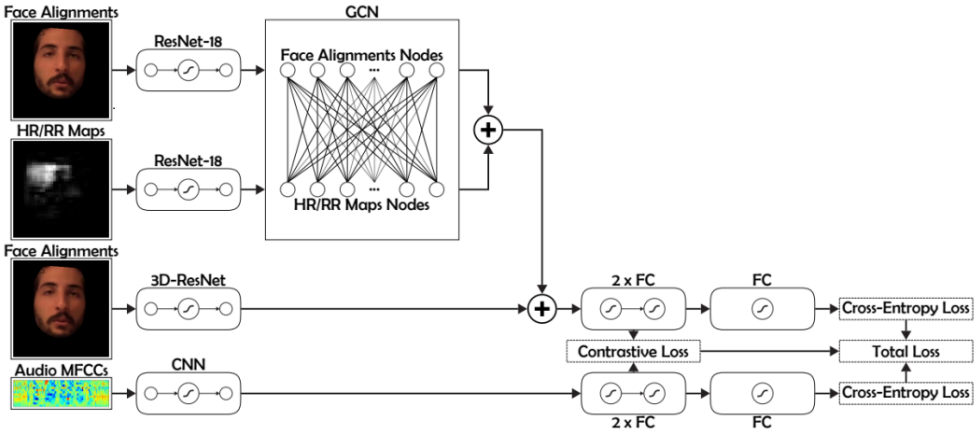


Figure 13. Two approaches combining visual representations with physiological signals Reprinted with permission from Ref. [50], 2024, Stefanov, K.

3.5. Experiment Procedures

3.5.1. Evaluating Metric Explanations

Accuracy is one standard used to evaluate metrics to illustrate model performance. However, there are two shortcomings when evaluating models. The first one is that it is not reliable when positive and negative samples are unbalanced. Another drawback is that it cannot present how many positive or negative samples are truly or wrongly predicted. Thus, it is necessary to calculate the confusion matrix to obtain the true-positive rate (TPR), true-negative rate (TNR), false-positive rate (FPR), and false-negative rate (FNR). Once these four indicators are calculated, the AUC and EER are two metrics that can be used for model evaluation.

There are three main evaluation metrics mentioned in this section, which are accuracy, AUC, and EER. AUC and EER are calculated by the sklearn confusion matrix directly in our experiments. Accuracy is the first evaluating metric that is commonly used in all classification tasks. It is defined as the correctness of the prediction of positive and negative samples. In this experiment, we regard one correct match between the predicted value and the label value as one count. Accuracy is the ratio of count numbers to the total sample number.

From previous analysis of evaluating metrics, there are four additional indicators that illustrate model prediction performance, which are TPR, TNR, FNR, and FPR for binary classification. These indicators can provide the greatest indication of how many

positive and negative samples are truly or falsely predicted. The confidence score of the current sample is used as the threshold by traversing all samples. Through multiple sets of thresholds, true-positive rate and false-positive rate pairs are calculated separately to draw a curve, namely, the ROC curve. Thus, the area under this curve is the area under the curve (AUC). Once the ROC is drawn, the equal error rate can be defined as the point where the false-negative rate equals the false-positive rate. In conclusion, the larger the accuracy and AUC values, the better the classification performance of the model. The smaller the EER value, the better the model performs.

3.5.2. Cross-Dataset Applications

The experiments of this review paper utilize cross-dataset validation. The purpose of cross-validation is to try to use different testing sets on the model to deal with too-one-sided results and insufficient training data. It plays a key role in building machine learning detection models because this process is not limited to the training phases and evaluates a specific model's predictive performance. Due to different forgery methods, the model aimed to detect different kinds of manipulated samples effectively; thus, the motivation is to evaluate model generalization ability and robustness when they have already been perfectly fitted in one training set. If the evaluating metric drops significantly, it proves that the generalization ability does perform well in the current forgery-validating dataset; otherwise, the model is general and robust. However, most detection models cannot effectively predict unseen forgery techniques at the feature level. It is reasonable to observe the evaluating metrics such as accuracy drop slightly in cross-validation; on the contrary, a large reduction in the evaluating metric is not acceptable, which means the detector is useless.

3.5.3. Capsule Net

This experiment only used the FaceForensics++ [13] database as the training dataset. On average, it selected 10 video frames of each video and obtained facial regions detected by "scrfd kps.onnx"; finally, the training, validating, and testing cropped facial image values are 15,783, 5259, and 5330, respectively, which are strictly followed by the ratio of 3:1:1. The Celeb-DF V2 [12] database followed the same operations, obtaining training, validating, and testing data values of 24827, 8275, and 8278, respectively. For a performance comparison, we performed multi-class training and binary class training separately and then utilized the trained model to perform in-dataset and cross-dataset experiments with a batch size of 16 accordingly. It firstly trained and tested Capsule Net only on multi-classes (Face2Face, DeepFakes, FaceSwap, and Real video) of FaceForensics++ obtaining a testing accuracy ranging from 92.2% to 98.73% in 30 epochs. Then, the binary class (Real or Fake) experiments of FaceForensics++ obtained the best performance accuracy of 93.95% and an EER of 4.34%, which shows good performance. However, when we utilized this model on testing Celeb-DF V2, the evaluating metric decreased largely on the deepfake class, but the original Real video class still displayed a relatively good result with a slight decrease. To observe the testing performances on each class of the forgery method, we performed a single-image prediction test on each class of FaceForensics++ and Celeb-DF via Capsule Network, and the results are shown in Table 6.

Table 6. Experiments of Capsule Networks trained on FF++ only.

Capsule Net	Original	DeepFakes	Face2Face	FaceSwap	Binary Test	Binary Test
	ACC	ACC	ACC	ACC	ACC	EER
Face Forensics++	96.56%	98.58%	96.68%	94.50%	93.95%	4.34%
Celeb-DF V2	91.67%		28.87%		30.50%	34.50%

This experiment result proves that the testing accuracy of deepfakes is quite dependent on the training data themselves by utilizing Capsule Network, and the performance indicators of the forgery class are often greatly reduced in cross-dataset experiments, even though we activated the dropout layer with a rate of 0.5. This is because the quality of databases is different from each other, and we conducted an experiment combining some Celeb samples into the training datasets. The accuracy of the testing results increased a lot, reaching 90% above as well. Thus, we can make two bold assumptions that (1) the network formed by stacking CBL modules and without any advanced data argumentation is not very effective in extracting the features of all kinds of manipulated regions, and (2) the loss that is only under the guidance of cross-entropy cannot provide reliable classification results.

3.5.4. CORE

The experiment utilized FF++ as the training set, which includes DeepFakes, Face2Face, FaceSwap, Neural Textures, and one extra class of deepfake detection. Each video split 30 frames with the relative FF++ facial masks to obtain human facial regions, and the training set and testing set strictly followed the FF++ data split requirements, which had 86,263 fake images for the training phase and 16,641 fake images for the test. The experiments explored and set a balance weight and cosine consistency loss to present the benchmark area under the curve (AUC), which reached 99.96% on the FaceForensics++ in-dataset test in 30 epochs. The cross-dataset test results reached 72.41% and 75.72% on DFDC and Celeb-DF, respectively. These results were also confirmed by our experiments.

From a previous literature review of the consistency representation method, this method supposes the generalization ability of one detection model depends on the consistency of its predicted results; that is, the real samples can correctly be predicted as real samples, no matter the data augmented. Thus, it is necessary to complete an ablation experiment to prove that the consistency of model prediction can be guaranteed to the largest extent by a specific data augmentation method. As above, we acknowledged that different augmentation methods and balance weights will present different evaluating results. Thus, we tested the ablation test on data augmentation, including RaAug, DFDC-Selim, and Random Erasing, finding that the best performance of CORE's data argumentation is DFDC-Selim [53]. The reproduced test results are shown in Table 7.

Table 7. Ablation AUC results via different data argumentation.

Data Augmentation	Celeb-DF V2
Random Erasing	74.78%
RaAug	67.83%
DFDC-Selim	79.45%

From the cross-dataset experiment on Celeb-DF, this detection method indeed largely enhances the ability of generalization, but the performance is largely determined by data augmentation methods and cannot reach a more reliable evaluating metric.

3.5.5. Spatio-Temporal Inconsistency Test

Although this experiment is a video-based detection method, it still requires the input of several video frame sequences. We utilized "scrfd 10 g kps.onnx" to extract facial regions with a crop area of 224 × 224. Then, we selected 8 frames to train this video-based model and 16 frames to test. The batch size was set to 16, and we chose the Adam optimizer to train the STIL model on one GPU within 30 epochs. After reproducing this project with the following parameters, we calculated the in-dataset binary accuracy results of Face Forensics++, Celeb-DF, DFDC, and ForgeryNet, as shown in Table 8. The cross-dataset experiment on Celeb-DF, DFDC, and Wild-DF presented evaluating metrics of total accuracy reaching 99.78%, 89.80%, and 84.12%, respectively.

Table 8. In-dataset testing accuracy results.

Datasets	FF++		Celeb-DF		DFDC		ForgeryNet	
	Real	Fakes	Real	Fakes	Real	Fakes	Real	Fakes
Sample Numbers	140	560	125	790	1034	1500	1474	1836
Accuracy	94.29%	96.25%	90.04%	99.87%	91.20%	95.87%	98.64%	98.37%

3.5.6. Wrong Predicted Samples of Single-Image Prediction Test

Although most deepfake generation algorithms include post-processing blocks such as color blending, the main challenge of the forgery algorithm is a clear forgery clue when facial occlusion happens. To our knowledge, we believe these samples are not hard samples, and the model should have a strong distinguishing ability for them. However, we provided several manipulated samples (see Figure 14) generated by FaceSwap and Roop for a better understanding of another detection challenge. We utilized two image detection methods, which are Capsule Network and CORE, to perform the single-image prediction test, and the following samples' results were both wrongly predicted as "Real Samples". On the other hand, some real samples were wrongly classified as "Fake Samples", which is not explainable by current detection algorithms. Thus, another challenge of deepfake detection is required to increase the precision and false-positive rate as well. Accuracy is a common metric but not the best evaluating metric to rate a model's performance.

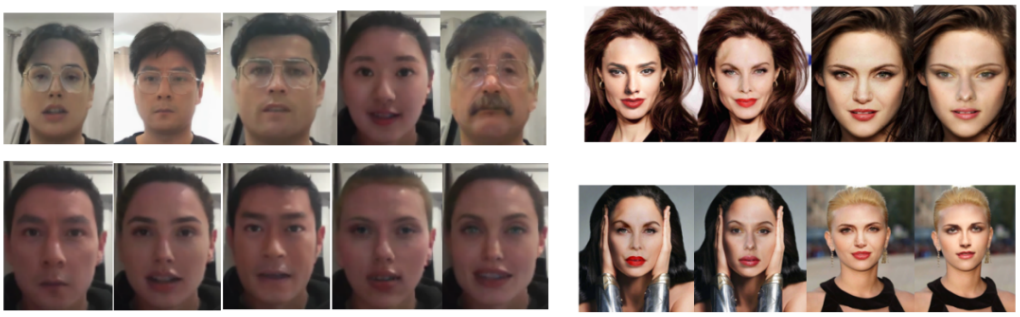


Figure 14. Some fake samples with obvious forgery clues are wrongly predicted as "Real".

4. Conclusions and Challenges

This survey offers a timely summary of deepfake detection datasets with their advantages and disadvantages and classifies deepfake detection methods into three commonly used categories from the perspectives of feature extraction methods. Moreover, we performed several in-dataset and cross-dataset testing experiments to calculate relative evaluating metrics such as accuracy and AUC by reproducing Capsule Net, CORE, and STIL codes. From the investigation, we propose three conclusions:

1. It is difficult for traditional CNN-based methods to provide good generalization ability on unseen data. This is because different forgery methods provide data qualities. For more realistic manipulation, it is difficult for the current model to distinguish from the texture level.
2. Even though semi-supervised methods also utilize the CNN-based backbone to extract features, they put more effort into calculating representation similarity and patch similarity or analyzing the temporal inconsistency of videos. They focus on the data themselves and find forgery clues from artifacts by different data augmentations or by the inconsistency of spatial and temporal information, which is an effective detection method in the current stage.
3. Frame-based detection also has another drawback in that it performs worse on Neural Textures because of the improvement in visual effects and imitating realistic facial structures. The Neural Textures algorithm can ensure the three-dimensional consistency of generative images, which means that an image rendered at different viewing

angles is similar to the observation of real, three-dimensional human faces. This forgery method is similar to the inpainting task to let AI fill in the defective parts to synthesize high-quality fakes. Thus, it is necessary to explore temporal-based detection to find the inconsistent relationship between video clips to solve this limitation.

Vision transformer has already proven that it can perform better benchmark on image classification tasks because it can provide the correlation between each patch feature, which is hard to complete in the CNN backbone. We investigated the potential of transformer blocks and assumed it could combine with consistency or inconsistency learning, for example, the extracted features obtained from the transformer can directly feed to the similarity calculation block by dividing the extracted features into several patches, and then the loss function is more diverse to filter out more "hard" samples. This aim is to find the inconsistency relationship between patches' PRNU noises, which can be an effective method to detect deepfakes. Since in the video transformer developed the attention is not only on different spatial features anymore, it can focus on temporal inconsistency by reshaping extracted features.

In addition, the main challenge of the transformer block is it requires corresponding pre-trained models. We utilized the transformer encoder block with initialized parameters and cross-entropy loss on the FF++ training set to train. It always showed low accuracy, reaching 64% on the validating phase, which is even worse than Capsule Network. Thus, it is important to select well-trained model parameters with the corresponding model architecture. Finally, we additionally propose three ideas for future deepfake detection trends:

1. For frame-based detection methods, consistency and inconsistency learning will become mainstream by combining transformer blocks. The total loss is also under the guidance of classification loss, consistency loss, and inconsistency loss, and the backbone of feature extraction can be replaced by transformer architectures to obtain a better corresponding feature.
2. Not only can deepfake detection detect forgery clues via videos, but detection models will develop multi-modalities such as the inconsistency of audio and detecting manipulated source generators. This is because of the development of biological signal-based detectors. They fuse biological signals such as PPG and provide and generate multi-modal data from different previous computer vision methods, which greatly enriches the diversity of forgery information and the model's learning ability.
3. There is still a lot of room to develop spatial and temporal detection methods with different methods to extract and fuse features. Video transformers can play a significant role in performing spatial and temporal feature extraction and attention mechanisms in this field.

Altogether, researchers face two main challenges in future deepfake detection. Firstly, the biggest challenge is the lack of datasets with large scale, various quality levels, and multiple attack methods. It is significant to enrich the diversity of data, including different ethnicities and shooting scenarios at the current stage, because models can learn different forgery features from data level while training, and it will be the first step to solving the problem of model generalization ability. Thus, datasets should develop with the deepfake generation development and maintain a long-term attacker and defender competition. Secondly, most forgery detectors are based on video frames. There is a lack of inter-frame temporal consistency detection methods to solve the problems of temporal abnormalities and real/fake frames in consecutive intervals. Finally, the ethical implications of deepfake detection will be discussed. As both generation and detection technologies continue to evolve in a competitive relationship, it is difficult to judge the ethics of deepfake detectors in the current stage. However, it can be concluded that unsophisticated detection skills and highly resourced sponsors' misinformation sometimes are at an intersection, which often misleads technical updates and ethics in this field. Thus, there is no "easy" approach to navigate the detector dilemma, but a set of implications derived from multi-stakeholders can better inform detection to process decisions and policy in the practice.

Author Contributions: Conceptualization, L.Y.G. and X.J.L.; methodology, L.Y.G. and X.J.L.; software, L.Y.G.; validation, L.Y.G.; formal analysis, L.Y.G.; investigation, L.Y.G.; resources, L.Y.G. and X.J.L.; data curation, L.Y.G. and X.J.L.; writing—original draft preparation, L.Y.G. and X.J.L.; writing—review and editing, L.Y.G. and X.J.L.; visualization, L.Y.G. and X.J.L.; supervision, X.J.L.; project administration, X.J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The relative datasets are downloaded from [8,13]. The reproduced codes are from [3,11,15,26,27,53].

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Abdulreda, A.S.; Obaid, A.J. A landscape view of deepfake techniques and detection methods. *Int. J. Nonlinear Anal. Appl.* **2022**, *13*, 745–755.
2. Zhang, L.; Lu, T. Overview of Facial Deepfake Video Detection Methods. *J. Front. Comput. Sci. Technol.* **2022**, *17*, 1–26.
3. FaceSwap-GAN. Available online: https://github.com/shaoanlu/faceswap-GAN (accessed on 15 December 2018).
4. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Wared-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Network. *Proceeding Commun. ACM* **2018**, *63*, 139–144. [CrossRef]
5. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. In Proceedings of the Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
6. Ho, J.; Jain, A.; Abbeel, P. Denoising Diffusion Probabilistic Models. In Proceedings of the 34th International Conference on Neural Information Processing System, Red Hook, NY, USA, 6–12 December 2020; pp. 6840–6851.
7. Rana, M.S.; Nobi, M.N.; Murali, B.; Sung, A.H. Deepfake Detection: A Systematic Literature Review. *IEEE Access* **2022**, *10*, 25494–25513. [CrossRef]
8. Kaggle. Available online: https://www.kaggle.com/c/deepfake-detection-challenge/overview (accessed on 12 December 2023).
9. Dimensions Scholarly Database. Available online: https://app.dimensions.ai/ (accessed on 10 December 2023).
10. DeepFaceLive. Available online: https://github.com/iperov/DeepFaceLive (accessed on 9 November 2023).
11. Roop. Available online: https://github.com/s0md3v/roop (accessed on 11 October 2023).
12. Li, Y.Z.; Yang, X.; Sun, P.; Qi, H.G.; Lyu, S. Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics. In Proceedings of the Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020.
13. Zhou, T.F.; Wang, W.G.; Liang, Z.Y.; Shen, J.B. Face Forensics in the Wild. In Proceedings of the Computer Vision and Pattern Recognition (CVPR), Virtual, 19–25 June 2021.
14. Guo, J.; Deng, J.; Lattas, A.; Zafeirioul, S. Sample and Computation Redistribution for Efficient Face Detection. In Proceedings of the Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020.
15. FaceSwap. Available online: https://github.com/deepfakes/faceswap (accessed on 10 November 2020).
16. Tolosana, R.; Romero-Tapiador, S. DeepFakes Evolution: Analysis of Facial Regions and Fake Detection Performance. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020.
17. Dang, H.; Liu, F.; Stehouwer, J.; Liu, X.; Jain, A. On the Detection of Digital Face Manipulation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020.
18. Karras, T.; Laine, S.; Aila, A. A Style-Based Generator Architecture for Generative Adversarial Networks. In Proceedings of the Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019.
19. Li, Y.; Chang, M.C.; Lyu, S. In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking. In Proceedings of the IEEE International Workshop on Information Forensics and Security, Hong Kong, China, 11–13 December 2018; pp. 1–7.
20. Nguyen, T.T.; Nguyen, Q.; Nguyen, D.; Nguyen, D.T.; Huynh-The, T.; Nahavandi, S.; Nguyen, T.; Pham, Q.; Nguyen, C. Deep Learning for Deepfakes Creation and Detection: A Survey. *arXiv* **2022**, arXiv:1909.11573.
21. Alahamari, F.; Naim, A.; Alqahtani, H. IoT-enabled Convolutional Neural Networks: Techniques and Applications. Chapter: E-Learning Modelling Technique and Convolution Neural Networks in Online Education. 2023. Available online: https://www.taylorfrancis.com/chapters/edit/10.1201/9781003393030-10/learning-modeling-technique-convolution-neural-networks-online-education-fahad-alahmari-arshi-naim-hamed-alqa (accessed on 5 January 2024).
22. Sabour, S.; Frosst, N.; Hinton, G.E. Dynamic Routing between Capsules. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 3859–3869.
23. Nguyen, H.H.; Yamagishi, J.; Echizen, I. Use of a Capsule Network to Detect Fake Images and Videos. In Proceedings of the Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019.
24. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014.

25. ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Available online: https://image-net.org/challenges/LSVRC/ (accessed on 13 December 2023).
26. Capsule-Forensics-v2: Implementation of the Capsule-Forensics-v2. Available online: https://github.com/nii-yamagishilab/Capsule-Forensics-v2 (accessed on 29 October 2019).
27. Ni, Y.; Meng, D.; Yu, C.; Quan, C.B.; Ren, D.; Zhao, Y. CORE: Consistent Representation Learning for Face Forgery Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022.
28. Zhong, Z.; Zheng, L.; Kang, G.; Li, S.; Yang, Y. Random Erasing data argumentation. In Proceedings of the AAAI, New York, NY, USA, 7–12 February 2020.
29. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
30. Li, L.; Bao, J.; Zhang, T.; Yang, H.; Chen, D.; Wen, F.; Guo, B. Face X-ray for More General Face Forgery Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 5000–5009.
31. Li, Y.; Lyu, S. Exposing DeepFake Videos By Detecting Face Warping Artifacts. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 16–17 June 2019; pp. 16–20.
32. Zhao, T.; Xu, X.; Xu, M.; Ding, H.; Xiong, Y.; Xia, W. Learning Self-Consistency for Deepfake Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021.
33. Chen, M.; Fridrich, J.; Goljan, M.; Lukas, J. Determining Image Origin and Integrity Using Sensor Noise. *IEEE Trans. Inf. Forensics Secur.* **2008**, *3*, 74–90. [CrossRef]
34. Barni, M.; Bondi, L.; Bonettini, N.; Bestagini, P.; Costanzo, A.; Maggini, M.; Tondi, B.; Tubaro, S. Aligned and Non-Aligned Double JPEG Detection Using Convolutional Neural Networks. *J. Vis. Commun. Image Represent.* **2017**, *49*, 153–163. [CrossRef]
35. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
36. Sun, K.; Yao, T.; Chen, S.; Ding, S.; Li, J.; Ji, R. Dual Contrastive Learning for General Face Forgery Detection. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2022; Volume 36, pp. 2316–2324.
37. Fridrich, J.; Kodovsky, J. Rich Models for Steganalysis of Digital Images. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 868–882. [CrossRef]
38. Gutmann, M.; Hyvarinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Sardinia, Italy, 13–15 May 2010; Volume 9, pp. 297–304.
39. Shi, X.; Chen, Z.; Wang, H.; Yeung, D. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Network. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015.
40. Gu, Z.; Chen, Y.; Yao, T.; Ding, S.; Li, J.; Huang, F.; Ma, L. Spatiotemporal Inconsistency Learning for Deepfake Video Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021.
41. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16 $\times$ 16 Words: Transformers for Image Recognition at Scale. In Proceedings of the International Conference on Learning Representations, Virtual, 26 April–1 May 2020.
42. Khan, S.A.; Dai, H. Video Transformer for Deepfake Detection with Incremental Learning. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, 20–24 October 2021; pp. 1821–1828.
43. Wodajo, D.; Atnafu, S. Deepfake Video Detection Using Convolutional Vision Transformer. In Proceedings of the Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021.
44. Khormali, A.; Yuan, J. DFDT: An End-to-End DeepFake Detection Framework Using Vision Transformer. *Appl. Sci.* **2022**, *12*, 2953. [CrossRef]
45. Trockman, A.; Zico Kolter, J. Patches Are All You Need? In Proceedings of the Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022.
46. Zhao, C.; Wang, C.; Hu, G.; Chen, H.; Liu, C.; Tang, J. ISTVT: Interpretable Spatial-Temporal Video Transformer for Deepfake Detection. *IEEE Trans. Inf. Forensics Secur.* **2023**, *18*, 1335–1348. [CrossRef]
47. Shaw, P.; Uszkoreit, J.; Vaswani, A. Self-attention with Relative Position Representations. In Proceedings of the NAACL 2018, New Orleans, LA, USA, 1–6 June 2018.
48. Ciftci, U.A.; Demir, İ.; Yin, L. How Do the Hearts of Deep Fakes Beat? Deep Fake Source Detection via Interpreting Residuals with Biological Signals. In Proceedings of the IEEE International Joint Conference on Biometrics (IJCB), Houston, TX, USA, 8 September–1 October 2020; pp. 1–10.
49. Wu, J.; Zhu, Y.; Jiang, X.; Liu, Y.; Lin, J. Local attention and long-distance interaction of rPPG for deepfake detection. In Proceedings of the Visual Computer, Lake Tahoe, NV, USA, 16–18 October 2023.
50. Stefanov, K.; Paliwal, B.; Dhall, A. Visual Representation of Physiological Signals for Fake Video Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022.
51. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. In Proceedings of the 5th International Conference on Learning Representations (ICLR 2017), Toulon, France, 24–26 April 2017.

52. Liu, X.; Fromm, J.; Patel, S.N.; McDuff, D. Multi-task temporal shift attention networks for on-device contactless vital measurements. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 19400–19411.
53. DFDC-Selium. Available online: https://github.com/selimsef/dfdc_deepfake_challenge (accessed on 11 December 2022).