Pattern Recognition 130 (2022) 108832

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Learning a deep dual-level network for robust DeepFake detection

Wenbo Pu[a], Jing Hu[a,*], Xin Wang[d,*], Yuezun Li[b], Shu Hu[d], Bin Zhu[c], Rui Song[e], Qi Song[a], Xi Wu[a], Siwei Lyu[d]

[a] Chengdu University of Information Technology, Chengdu, China
[b] Ocean University of China, China
[c] Microsoft Research Asia,Beijing, China
[d] University at Buffalo, State University of New York, Buffalo, NY, USA
[e] Department of Statistics, North Carolina State University

ARTICLE INFO

Article history:
Received 24 August 2021
Revised 29 March 2022
Accepted 3 June 2022
Available online 3 June 2022

Keywords:
DeepFake detection
Multitask learning
Imbalanced learning
AUC optimization

ABSTRACT

Face manipulation techniques, especially DeepFake techniques, are causing severe social concerns and security problems. When faced with skewed data distributions such as those found in the real world, existing DeepFake detection methods exhibit significantly degraded performance, especially the AUC score. In this paper, we focus on DeepFake detection in real-world situations. We propose a dual-level collaborative framework to detect frame-level and video-level forgeries simultaneously with a joint loss function to optimize both the AUC score and error rate at the same time. Our experiments indicate that the AUC loss boosts imbalanced learning performance and outperforms focal loss, a state-of-the-art loss function to address imbalanced data. In addition, our multitask structure enables mutual reinforcement of frame-level and video-level detection and achieves outstanding performance in imbalanced learning. Our proposed method is also more robust to video quality variations and shows better generalization ability in cross-dataset evaluations than existing DeepFake detection methods. Our implementation is available online at https://github.com/PWB97/Deepfake-detection.

1. Introduction

Human faces play an important role in our daily life. As a result, the recent proliferation of facial content manipulation has attracted substantial concerns. One type of facial content manipulation is DeepFake, in which a deep learning-based technique is used to swap an original face in a video with the face of another individual while retaining the original facial expressions, head postures, and lighting conditions. With the rapid development of deep neural networks such as convolutional autoencoders [1] and generative adversarial networks (GANs) [38–41], DeepFakes have achieved impressive visual results that are indistinguishable to the naked eye. Due to their widespread usage on social media, DeepFake videos have become a serious social concern and security problem.

Such concerns have brought increasing research activities on DeepFake detection. Most existing works treat DeepFake detection as a binary classification task and use deep neural networks to detect videos or images forged with DeepFakes. For example, MesoNet [2] uses a specially designed convolutional neural network (CNN) backbone to focus on the mesoscopic properties of DeepFake images. Xception [3] is based on training an XceptioNet [4] model using the traditional cross-entropy loss on the FaceForensics++ [3] dataset.

Although these methods achieve good performance on benchmark datasets, their performance degrades significantly when facing imbalanced datasets widely seen in the real world: real videos usually significantly outnumber fake videos, i.e., the distribution is very skewed (see Fig. 1(a)). Our study indicates that the performance of existing DeepFake detection methods generally degrades when the imbalanced level of a dataset increases. As shown in Fig. 1(b), 4 existing state-of-the-art DeepFake detection methods, i.e., Meso4 [2], Xception [3], the Dual Spatial Pyramid For exposing face Warp Artifacts (DSP-FWA) [5], and Capsule [6], have a significant performance drop when the imbalanced level of the dataset increases. To alleviate the adverse impact of the data imbalance problem, it is straightforward to re-sample data from or adjust weights of different classes, but these approaches cannot eliminate the adverse impact of imbalanced data distributions since different weights are required as the dataset varies and removal or repetition of the original data can lead to overfitting of a deep-learning model [7]. Another method for alleviating the imbalanced learning effect is data augmentation, such as flipping, rotating, scaling,

* Corresponding authors.
E-mail addresses: lesley123@cuit.edu.cn (J. Hu), xwang264@buffalo.edu (X. Wang).

https://doi.org/10.1016/j.patcog.2022.108832
0031-3203/

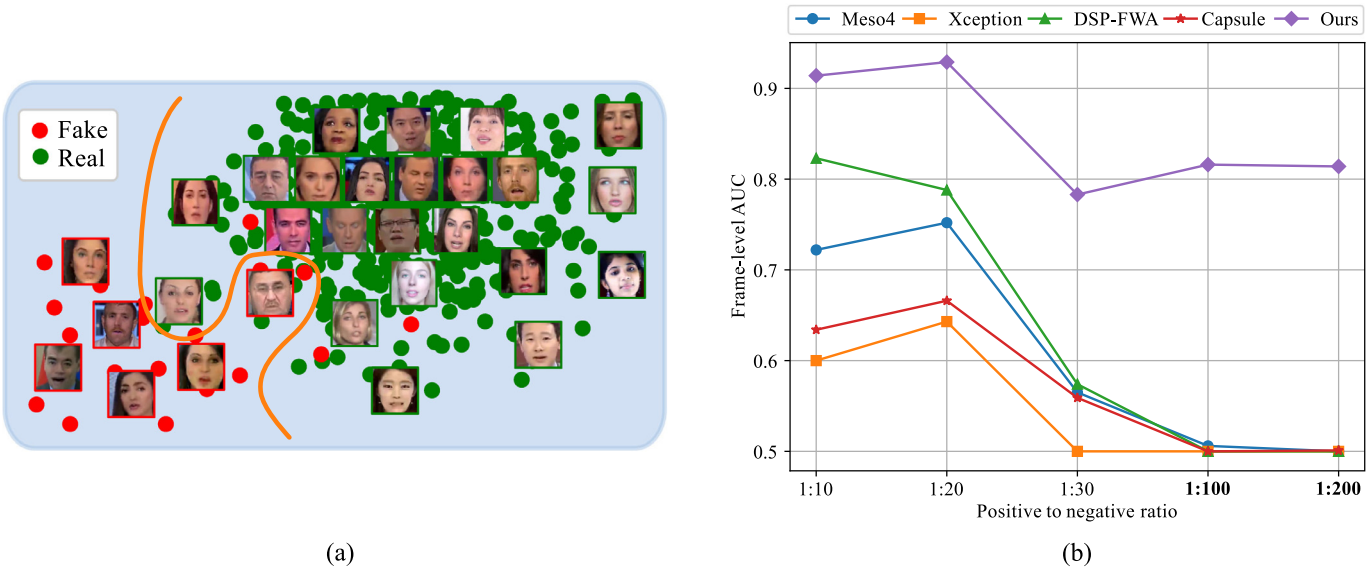

Fig. 1. (a) DeepFake detection on imbalanced data. (b) The performance of frame-level DeepFake detection methods trained on imbalanced subsets from Celeb-DF and **DFDC**. The *X*-axis indicates different positive (fake) to negative (real) ratios, and the *Y*-axis indicates AUC scores of different DeepFake detection methods on these ratios (see Section 4 for the detail of the experimental settings). The performance of the existing methods degrades significantly while our proposed method drops a little when the dataset imbalance level increases.

and cropping, to expand minor class sets. However, the data extended by data augmentation contain basically the same information as the original data. Therefore, data augmentation does not adequately solve the problem. Alternatively, DeepFake methods can be used to generate new data to alleviate the imbalanced level, but this is a time-consuming process. A data-manipulation-free solution to solve the data imbalance problem is desirable.

Furthermore, most DeepFake detection methods classify forged videos only at either frame or video level. Video-level methods [8,9] can detect a forged video but cannot predict forgery of individual frames in a forged video. In practice, it is important to identify specific fake frames when not all video frames are forged. Frame-level methods such as Capsule [6] and Xception [3], on the other hand, detect forgery of a single image or individual video frames but cannot provide an integrated prediction at the video level. A simple frame-level method for handling video-level prediction is to use the maximum of each frame's probability or average them [10]. However, a recent work [9] indicates that this solution may lead to unexpected false alarms or false-negative predictions. More importantly, frame-level methods do not take temporal information in a video into consideration. Temporal information plays an important role in video-level detection and can boost video-level performance [9,10]. It is desirable to combine the capabilities of both frame-level and video-level methods together to not only detect each forged frame but also integrate the temporal frame information to provide an overall prediction for a whole video in an end-to-end system.

In this paper, we propose a novel DeepFake detection method that fulfills dual goals: addressing the data imbalance problem and detecting forgery at both frame-level and video-level simultaneously. To address the data imbalance problem, we introduce a loss function that can directly maximize the area under the receiver operating characteristic (ROC) curve (AUC) to minimize the adverse impact of an imbalanced dataset as much as possible. AUC is a robust metric used to evaluate the classification capability of a model, especially when addressing imbalanced data [7]. However, AUC is a nondifferentiable function that cannot be directly applied to a loss function to train a neural network. Inspired by the work in [11], the Wilcoxon-Mann-Whitney (WMW) statistic is an equivalent function to AUC, and one of its approximations is differentiable and easy to compute. We combine this approximation, denoted as the AUC loss, with the traditional loss function, cross-entropy, to pursue both accuracy (ACC) and AUC performance simultaneously. To detect forgery at both frame-level and video-level, we leverage multitask learning [12] to combine the features of both video-level and frame-level methods into a collaborative network that can detect each individual forged frame and take temporal information into consideration to make an integrated prediction for a whole video.

Our proposed method is evaluated together with existing state-of-the-art methods on two public datasets, Celeb-DF (v2) and FaceForensics++ [3]. The experimental results indicate that our method achieves high ACC and AUC scores at both frame-level and video-level without extra sampling or class weight adjustments. It shows robustness under low-quality conditions and is generalizable in cross-dataset evaluations. Our ablation studies prove that our proposed method has an effective structure design. To evaluate our method's robustness when facing real-world imbalanced conditions wherein real videos significantly outnumber fake videos, we sample the Celeb-DF and DeepFake Detection Challenge (DFDC) [13] datasets to generate five subsets with different positive-to-negative ratios. The experimental results on these subsets show that our method still achieves significantly higher ACC and AUC scores than the existing state-of-the-art methods and the proposed structure improves both frame-level and video-level performance when facing imbalanced data. As shown in Fig. 1(b), our method achieves robust performance, especially on extremely imbalanced datasets such as 1:100 and 1:200.

The main contributions of this paper are as follows:

- When facing real-world imbalanced data in DeepFake detection, instead of traditional sampling or data augmentation to make the dataset more balanced, we minimize its adverse impact by introducing an AUC loss to directly maximize the AUC score while maximizing the traditional prediction accuracy.
- We propose an integrated network that combines the features of frame-level and video-level DeepFake detection methods to detect not only each individual forged frame but also provide an overall prediction for a whole video by taking temporal relations of frames into consideration. This unique struc-

ture also improves the performance under imbalanced data conditions.
- We sample the Celeb-DF and DFDC datasets to simulate real-world imbalanced data conditions and conduct an extensive experimental evaluation and analysis of DeepFake detection performance under imbalanced dataset conditions. Our study indicates that our proposed method outperforms existing DeepFake detection methods and is robust when facing the data imbalance problem.

The paper is organized as follows. Section 2 summarizes related work on DeepFake generation, DeepFake detection, and imbalanced learning. Section 3 introduces the proposed network architecture and the joint loss function. In Section 4, we present and visualize experimental results on popular benchmark datasets and imbalanced learning performance. Finally, we conclude the paper with Section 5.

2. Related work

In this section, we briefly review DeepFake generation and existing DeepFake detection methods. We also review works on imbalanced learning.

2.1. DeepFake generation

DeepFake is a recent face manipulation technique based on autoencoders [1] and GANs [38–41]. The GAN in the DeepFake generation algorithm comprises two competing neural networks: a generator G that generates fake faces from input noise vectors that mimic real faces from the dataset and a discriminator D that distinguishes the generated fake faces from real faces. After the training procedure, the generator is used to synthesize DeepFake faces.

The two typical DeepFake approaches are face swapping and face re-enactment. In face swapping, the face of a target identity is overlaid on the face of a source identity with computer graphics techniques such as affine transformation. FaceSwap [14], a representative face-swapping approach, is based on computer graphics and replaces the original identity while preserving expressions. Face re-enactment makes a target identity act and speak as if it were the source identity. Face2Face [1] is a face re-enactment method that is more sophisticated than FaceSwap and uses only a red, green, and blue (RGB) camera to enable facial expression re-enactment. Neural Textures [15] is a facial re-enactment generation method that can synthesize photorealistic images from imperfect original 3D content. Liu et al. [16] propose a sequence-to-sequence CNN to automatically synthesize talking face videos with accurate lip-sync, which is a typical facial re-enactment application. Synthesizing Obama [42] is an application that can synthesize Barack Obama speaking in videos with his audio.

2.2. DeepFake detection

DeepFake detection methods can be classified into two categories depending on the detection goal: frame-level methods that detect forgeries of individual frames and video-level methods that detect entire video forgeries.

2.2.1. Frame-level methods

The majority of DeepFake detection methods are frame-specific methods. These methods can generally be further divided into two categories: CNN-based classification and autoencoder-based forgery localization.

CNN-based classification uses a CNN for feature map extraction and detects DeepFake forgeries as a binary classification problem [43]. MesoNet [2] contains a specially designed CNN backbone to focus on the mesoscopic properties of images to identify manipulated frames of videos forged with DeepFake or Face2Face [1] techniques. The method in [3] is based on training an XceptionNet [4] model with the FaceForensics++ dataset. The face warp artifact (FWA) method [5] uses ResNet-50 [17] as its backbone to expose face warping artifacts introduced by the DeepFake synthesis process in a self-supervised way. Capsule [6] introduces a capsule network to detect fake images. Zhu et al. [18] introduce 3D decomposition into forgery detection. FakeCatcher [19] exploits biological signals extracted from facial areas to detect synthetic portraits. PRRNet [20] can capture pixelwise and regionwise relations for face forgery detection. Luo et al. [21] utilize high-frequency noise for face forgery detection.

Forgery localization methods focus on localizing manipulated regions. Bappy et al. [22] propose a localization framework based on encoder-decoder and long short-term memory (LSTM) methods that exploits both frequency domain features and spatial contexts to localize manipulated image regions. But this kind of methods apply supervised training and require a large number of labeled manipulation masks. Face X-ray [23], in contrast, locates manipulated regions in a self-supervised manner.

Frame-level methods ignore the temporal information in videos and cannot directly predict forgery of an entire video. Li et al. [24] address this limitation with a rule-based method that detects an unusual rate of eye blinks to determine whether a video is manipulated. However, the threshold of an abnormal blink rate is laborious to determine and may not fit all videos.

2.2.2. Video-level methods

In contrast to frame-level methods, video-level methods consider all video frames to reveal the temporal flickering of a DeepFake video through sequence learning. Güera and Delp [8] propose a two-stage analysis method comprising a CNN model and a recurrent neural network (RNN) to capture temporal inconsistencies among frames. DeepFakesON-Phys [25] detects fake videos by considering information related to the heart rate using remote photoplethysmography (rPPG). Li et al. [9] propose a sharp multiple-instance learning (S-MIL) network to address the partial face attack problem in DeepFake videos with a multiple-instance learning framework.

Video-level methods lack the ability to detect forgeries of individual frames and have difficulty handling situations when only a subset of frames is manipulated in a video.

2.3. Imbalanced learning

Imbalanced learning is learning from imbalanced data [44,45]. There are basically two approaches to solving the imbalanced learning problem. One focuses on the dataset, while the other focuses on optimization. For the former approach, a commonly used method is random sampling, oversampling and undersampling. To alleviate the adverse impact of imbalanced data on DeepFake detection, Masi et al. [10] use oversampling to train their model on the FaceForensics++ dataset: the minor class set is oversampled, while the major class set is undersampled. However, all sampling methods have the following challenges: oversampling causes repetitive samples in datasets, which leads to model overfitting [7], while undersampling may lose some important information due to removed samples [26]. It is possible to leverage data augmentation solutions such as flipping, rotation, scaling, and cropping to expand minor class sets. Unlike images, however, it is difficult to find a simple method for expanding videos with temporal consistencies. Some works [5,8] use homemade datasets generated by DeepFake generation methods to train their models, which is a time-consuming process.

The latter approach is to apply a loss function to adjust data weights directly. For example, focal loss (FL) [27] addresses a seriously imbalanced ratio of positive to negative samples in one-stage object detection by reshaping the standard cross-entropy loss to focus more on difficult samples. Recently, Zhou et al. [28] propose a unified Bilateral-Branch Network (BBN) to take care of both representation learning and classifier learning for exhaustively boosting long-tailed recognition.

3. Our method

In this section, we first describe the intuitive idea behind our method, then introduce our method's network structure, which leverages multitask learning to combine both video-level and frame-level structures, and finally present a detailed description of our joint loss function to address the data imbalance problem.

3.1. The intuitive idea behind our method

The most commonly used evaluation metric for imbalanced learning is the AUC score. The ROC curve takes the true-positive rate (TPR) as the Y-axis and the false-positive rate (FPR) as the X-axis. Each point on this curve consists of different pairs of (FPR, TPR) under different classification thresholds. Different classification thresholds can cause different results. The AUC score takes into account the change in the threshold. In other words, AUC is a robust measure of classifier's discrimination performance [11]. Suppose a dataset contains a very small proportion of fake samples (assuming 0.1%), and a classifier always predicts an input sample as a real one, i.e., the classifier cannot discriminate fake samples from real ones. For such an extreme case, the ACC score of this classifier is 99.9%, while its AUC score is only 0.5 since the TPR and the FPR are both 1. AUC still provides a reasonable evaluation of a classifier's discrimination capability in face of imbalanced data.

Since AUC is a robust evaluation metric, it is desirable to seek a loss function that can directly maximize the AUC score without any data manipulation techniques. However, AUC is a nondifferentiable function that cannot be directly applied to a loss function to train a neural network. Fortunately, the WMW statistic is equivalent to AUC, and one of its approximations is differentiable and easy to compute [11]. We combine this approximation with the traditional cross-entropy to maximize both ACC and AUC scores simultaneously.

In addition, to enable DeepFake detection at both frame and video levels, we leverage multitask learning [12] to combine the features of both frame-level and video-level methods into a collaborative network that can detect forgery of each individual frame and take temporal information into consideration to make an integrated prediction for a whole video simultaneously. As reported in Section 4, the proposed collaborative structure can also improve the performance with imbalanced data.

3.2. Overview

The pipeline of our proposed method is illustrated in Fig. 2. Faces in each frame of an input video are extracted with the Dlib face detector [29] and used as the video input to our neural network pipeline. The first module of the pipeline is a pretrained CNN module to extract features from each face. It can detect visual inconsistencies or high-level spoofing information in the spatial domain. This pretrained CNN module can be implemented with commonly used neural networks, such as ResNet [17] or Xception [4]. The output of the CNN module, which contains the spatial domain information of each face, is fed into a temporal learning module that can be implemented with LSTM or gated recurrent unit (GRU) [30] layers. This module learns the temporal information among frames, such as frame flickering or other inconsistencies among frames. The temporal learning module output is fed into a frame-level classifier and a video-level classifier simultaneously to predict both frame-level and video-level forgeries. The frame-level classifier can be implemented with a fully connected layer that accepts the output vector of each frame from the temporal learning module. A pooling layer that summarizes the information from each frame and a fully connected layer can be used as the video-level classifier.

AUC is a robust metric for evaluating the classification model performance, especially when the data are imbalanced. A classifier is traditionally trained by minimizing the error rate, which may not maximize the AUC score for imbalanced data. To solve the data imbalance problem, we jointly maximize the AUC score and minimize the traditional cross-entropy (i.e., error rate). Our model achieves good classification performance, even for imbalanced data, without relying on traditional techniques to address imbalanced data, such as oversampling/undersampling or data augmentation.

3.3. Detailed network architecture

As illustrated in Fig. 2, our neural network pipeline consists of three main components: a pretrained CNN module, a temporal learning module, and two classifiers for both frame-level and video-level detection. These components can be implemented in different ways. Figure 2 shows the specific implementation of the three modules that we use in our experimental evaluation of the proposed method. We assume an input video contains m frames, each of which has a size of $w \times h \times 3$. As a result, the input size of our model is $m \times w \times h \times 3$.

3.3.1. Pretrained CNN

The pretrained CNN module is used to extract the spatial features of each frame. We choose ResNet-50 [17] pretrained on ImageNet and without the last fully connected (FC) layer as our implementation, and the weights of CNN will be fine-tuned during the training procedure. The feature map from this module is flattened and fed into the temporal learning module. The size of an output feature map from the pretrained CNN module is $m \times 2048$.

3.3.2. Temporal learning module

The temporal learning module is used to exploit temporal information among frames. We choose 3 GRU [30] layers to implement this module, with 256 hidden nodes for each layer. After the temporal learning module, the size of an output feature map is $m \times 256$.

3.3.3. Video-level classifier and frame-level classifier

A feature map output from the temporal learning module is fed into both the video-level classifier and the frame-level classifier. In the video-level classifier, an average pooling layer summarizes each frame's feature map and generates a 16×16 feature map that contains both the spatial and temporal information of the whole video. This 16×16 feature map is then flattened to a vector of size 256 and fed into the last FC layer for video-level classification. After the sigmoid function, a probability of less than 0.5 indicates that the input video is fake. In addition, the frame-level classifier accepts the feature map of size 256 of each frame from the temporal learning module and uses an FC layer to classify the frame. We set the length of an input video to 300 frames (see Section 4 for more details) and the input frame size to 64×64.

3.4. Normalized WMW statistic

Most existing classification loss functions, such as cross-entropy, are insufficient for addressing the data imbalance problem

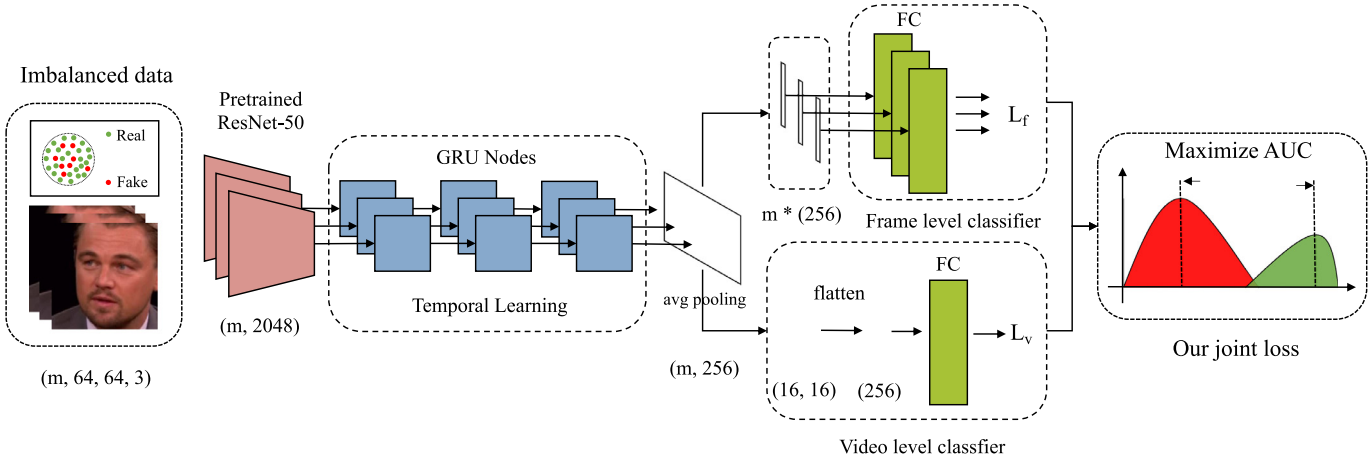

Fig. 2. The detailed network architecture of our proposed method. An extracted face sequence is fed into a pretrained ResNet-50 for feature extraction and GRU layers for temporal learning. The extracted feature map is then fed into two separate classifiers for both video-level and frame-level predictions.

and cause the trained classification model to produce accurate but biased predictions, which may not fit practical application requirements. It is desirable to use a specifically designed loss function to address data imbalances directly. Since AUC is a robust evaluation metric for both balanced and imbalanced data, we want to directly maximize AUC to handle imbalanced situations. An ROC curve is composed of FPRs and TPRs, with different decision thresholds ranging from 0 to 1. The more bowed the ROC curve is toward the upper left corner, the higher the AUC score, and the better the classifier's ability to discriminate between the two classes. This AUC score computation cannot be used in a loss function since it is not easy to compute. However, the normalized WMW statistic [11] is equivalent to AUC. Given a labeled dataset $\{(x_i, y_i)\}_{i=1}^{M}$ with M samples, where each data sample $x_i \in \mathbb{R}^d$ and each corresponding label $y_i \in \{0, 1\}$. We define a set of indices of positive instances as $\mathcal{P} = \{i | y_i = 1\}$. Similarly, the set of indices of negative instances is $\mathcal{N} = \{i | y_i = 0\}$. Let $\mathcal{F} : \mathbb{R}^d \to \mathbb{R}$ be a parametric predict function. $\mathcal{F}(x_i)$ represents the prediction score of the ith sample, where $i \in \{1, \cdots, M\}$. For simplicity, we assume $\mathcal{F}(x_i) \neq \mathcal{F}(x_j)$ for $x \neq j$ (ties can be broken in any consistent way). Then, the normalized WMW can be formulated as follows,

$$\begin{aligned} \text{WMW} &= \frac{1}{|\mathcal{P}||\mathcal{N}|} \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{N}} \mathbb{I}_{[\mathcal{F}(x_i) > \mathcal{F}(x_j)]}, \quad \text{where} \mathbb{I}_{[\mathcal{F}(x_i) > \mathcal{F}(x_j)]} \\ &= \begin{cases} 1, & \mathcal{F}(x_i) > \mathcal{F}(x_j), \\ 0, & \text{otherwise}. \end{cases} \end{aligned} \tag{1}$$

The normalized WMW is computed based on pairwise comparisons between the classifier's outputs for $|\mathcal{P}|$ positive examples and $|\mathcal{N}|$ negative examples. Although it is easier to compute than AUC, the normalized WMW statistic is not differentiable due to discrete computation. It is, therefore, desirable to find a differentiable approximation of the WMW statistic.

3.5. Objective function

Inspired by the work in [11], we use an approximation of the normalized WMW statistic that can be directly applied to our objective function to maximize AUC along with our imbalanced training procedure. More specifically, maximizing the normalized WMW statistic in Eq. (1) can be approximated by minimizing the following loss function:

$$\mathcal{L}_{AUC}(\mathcal{F}(x), y) = \frac{1}{|\mathcal{P}||\mathcal{N}|} \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{N}} R(\mathcal{F}(x_i), \mathcal{F}(x_j)). \tag{2}$$

with the differentiable function:

$$R(\mathcal{F}(x_i), \mathcal{F}(x_j)) = \begin{cases} (-(\mathcal{F}(x_i) - \mathcal{F}(x_j) - \gamma))^p, & \mathcal{F}(x_i) - \mathcal{F}(x_j) < \gamma, \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where γ and p are two hyperparameters, $0 < \gamma \leq 1$ and $p > 1$. This loss term is denoted as the AUC loss function. Our method optimizes a joint loss function comprising the conventional binary cross-entropy (BCE) loss function and the differentiable AUC loss function in Eqs. (2) and (3).

For each of the video-level and frame-level prediction tasks, we optimize the BCE loss and the AUC loss simultaneously. Assume training dataset $\mathcal{X}$ contains N labeled videos: $\mathcal{X} = \{(\mathcal{V}_i, Y_i)\}_{i=1}^{N}$, where $\mathcal{V}_i$ is the ith video containing P frames, represented as $\mathcal{V}_i = \{(\mathcal{I}_i^j, \hat{Y}_i^j)\}_{j=1}^{P}$. Then, Y_i is the label of video $\mathcal{V}_i$ and $\hat{Y}_i^j$ is the label of frame $\mathcal{I}_i^j$. Denote $\mathcal{F}_v$ and $\mathcal{F}_f$ as the predict functions of the video-level classifier and the frame-level classifier, respectively. Then, our video-level loss $\mathcal{L}_v$ and frame-level loss $\mathcal{L}_f$ functions are designed as follows:

$$\begin{aligned} \mathcal{L}_v &= \alpha \mathcal{L}_{BCE}(\mathcal{F}_v(\mathcal{V}), Y) + (1 - \alpha)\mathcal{L}_{AUC}(\mathcal{F}_v(\mathcal{V}), Y), \\ \mathcal{L}_f &= \alpha \mathcal{L}_{BCE}(\mathcal{F}_f(\mathcal{I}), \hat{Y}) + (1 - \alpha)\mathcal{L}_{AUC}(\mathcal{F}_f(\mathcal{I}), \hat{Y}), \end{aligned} \tag{4}$$

where α is a scaling factor that is designed to balance the BCE loss and AUC loss function weights. The final loss function of our model is

$$\mathcal{L} = \beta \cdot \mathcal{L}_v + (1 - \beta) \cdot \mathcal{L}_f, \tag{5}$$

where β is a hyperparameter for the scaling factor of $\mathcal{L}_v$ and $\mathcal{L}_f$ and is designed for adjusting the video-level and frame-level prediction weights.

4. Experimental evaluation

We report two major experiments in this section. In the first experiment (Section 4.3), we compare the performance of our model with that of some state-of-the-art methods on public datasets. In the second experiment (Section 4.4), we evaluate imbalanced learning performance. We sample from the Celeb-DF and DFDC datasets to generate five subsets with different ratios of positive samples to negative samples and use them to evaluate our model and compare it with other state-of-the-art methods.

4.1. Datasets

We evaluate our proposed method on the Celeb-DF (v2) [31] and FaceForensics++ [3] datasets. We also use Celeb-DF and

Table 1
Details of the imbalanced subsets from Celeb-DF and DFDC (Since there is no validation set in Celeb-DF and DFDC, we use the validation set of FaceForensics++ instead to determine hyperparameters on these subsets).

Datasets	Train		Test		Ratio (Real:Fake)
	Real	Fake	Real	Fake	
Celeb-30	712	23	178	5	30:1
Celeb-20	712	35	178	8	20:1
Celeb-10	712	71	178	17	10:1
DFDC-100	500	5	300	3	100:1
DFDC-200	1000	5	600	3	200:1

DFDC [13] to generate subsets to evaluate imbalanced learning performance.

FaceForensics++ consists of videos obtained from four face-swapping algorithms: DeepFakes [32], FaceSwap [14], Face2Face [1], and Neural Texture [15]. To be consistent with Celeb-DF, only the videos manipulated by the DeepFakes algorithm are used in our experiments. They have three video quality levels from high to low, i.e., raw, c23, and c40, with an increasing degree of compression. To mimic the real-world scenario in which many videos are low quality and blurry, our model is trained on medium-quality data, c23, in our experimental evaluation.

The original training and testing split is used in our experiments for both Celeb-DF and FaceForensics++. Since the original Celeb-DF dataset does not include a validation set, we use the validation set of FaceForenics++ to determine the hyperparameters of the proposed loss in this experiment. The Celeb-DF dataset is very imbalanced but in the opposite of the real-world situation: the number of DeepFake videos is approximately 7 times that of real videos. On the other hand, the FaceForensics++ dataset used in our experiments is balanced: the numbers of DeepFake and real videos are the same.

To test the robustness of our model when facing different imbalanced data scenarios in the real world, we need to evaluate it with imbalanced datasets that are similar to real-world situations. The above datasets do not provide a realistic data distribution. To mimic imbalanced data in the real world, a straightforward method is to sample from existing datasets to generate subsets with different ratios of positive to negative samples. We choose Celeb-DF and DFDC as the source datasets to sample due to their large numbers of identities and outstanding quality. We randomly sample from Celeb-DF to generate three subsets, denoted as Celeb-10, Celeb-20, and Celeb-30, with the following ratios of positive to negative samples: 1:10, 1:20, and 1:30, respectively. We regard fake videos as positive samples and real videos as negative samples. We also randomly sample from DFDC to generate two subsets, denoted as DFDC-100 and DFDC-200, with the following ratios of positive to negative samples: 1:100 and 1:200, respectively. Table 1 shows the details of our prepared subsets. We note that we have not generated a validation set for any of these subsets. This is because we use the validation set of FaceForensics++ to determine the hyperparameters for the experiments on these subsets (see Section 4.2.3 for details). The fake videos are randomly sampled from the original fake videos to ensure that the datasets contain as many identities as possible. Combined with the original Celeb-DF dataset, in which there are many more positive videos than negative videos, these datasets cover various scenarios of imbalanced data.

4.2. Experimental setup

4.2.1. Data preprocessing

Raw videos in each dataset are preprocessed with the following steps. First, we use the Dlib [29] face detector to extract human faces in every frame of each video. The extracted faces are then resized to a fixed size. In most of our experiments, this size is set to 64×64 pixels, normalized with the mean and standard deviation of ImageNet and grouped for each video. Since the average number of frames is approximately 300 for videos in the Celeb-DF dataset, we set the frame length of an input video to 300 in our experiments. If a video has fewer than 300 frames, its last frame is repeated to reach 300 frames. For all of our experiments, we evaluate our method and the comparison methods without extra data augmentation or sampling strategies to simulate data imbalance scenarios.

4.2.2. Comparison with existing methods

To evaluate the performance of our proposed method, we choose seven existing methods for comparison, including both frame-level methods and video-level methods, all of them use only the cross-entropy loss without the AUC loss. The code for these frame-level methods is publicly available.

- MesoNet[1] [2] is a CNN-based frame-level method that focuses on mesoscopic image properties. It has two variants, Meso4 and MesoInception4. We compare our method with both variants.
- Xception[2] [3] is a frame-level method that uses XceptionNet [4] to train on the FaceForensics++ dataset.
- Capsule[3] [6] is another frame-level method trained on the FaceForensics++ dataset and uses capsule structures [33] based on VGG19 [34] as the backbone for DeepFake detection.
- DSP-FWA[4] is a frame-level method based on FWA [5]. It includes a spatial pyramid pooling (SPP) [35] module to handle resolution variations in the original target faces.

The long-term recurrent convolutional network (LRCN) [36] uses a network structure that combines CNN and LSTM. It is widely used for video-level DeepFake detection [8]. ResNet-50 is used as the backbone for LRCN in our experimental evaluation, which is basically the same architecture as that used in [8]. We also add a variant that replaces LSTM with GRU [30] in our comparison. These two methods are denoted as CNN+LSTM and CNN+GRU, respectively.

We train our proposed method and all the comparison methods five times except DSP-FWA [5] on the low-resolution datasets in our experimental evaluation. For DSP-FWA, a model pretrained on its native resolution (224×224) is used in our experiments to evaluate its performance on the Celeb-DF and FaceForensics++ datasets, since DSP-FWA incorporates an SPP module that cannot be retrained with low-resolution images. As a result, we don't repeatedly test DSP-FWA like other methods.

In our evaluation of the frame-level methods, we report only the frame-level results of the compared methods. Although their video-level results can be generated by using averaging or the maximum of each frame's probability, such a simple solution may lead to unexpected alarms or false-negative predictions [9]. Thus, video-level results are not provided in our experiments if their original methods cannot predict video-level results.

In our imbalanced learning experiments, we evaluate all methods on our subsets, Celeb-10, Celeb-20, Celeb-30, DFDC-100, and DFDC-200. On these subsets, we load pretrained models for the comparison frame-level methods, and fine-tune them with imbalanced training data, while our method, CNN+LSTM, and CNN+GRU are loaded with ResNet-50 trained with ImageNet as the backbone. DSP-FWA is loaded as its pretrained model and fine-tuned on our imbalanced subsets with input resolution 224×224. The other

[1] https://github.com/DariusAf/MesoNet
[2] https://github.com/ondyari/FaceForensics
[3] https://github.com/nii-yamagishilab/Capsule-Forensics-v2
[4] https://github.com/danmohaha/DSP-FWA

Table 2
Performance results of different detection methods on FaceForensics++.

Method	Video Level					Frame Level				
	ACC	AUC	F1	Precision	Recall	ACC	AUC	F1	Precision	Recall
DSP-FWA [5][a]	-	-	-	-	-	51.2	51.4	67.0	50.9	100
Meso4 [2]	-	-	-	-	-	70.4±1.02	77.6±0.85	72.7±2.69	63.2±1.42	85.6±1.01
MesoInception4 [2]	-	-	-	-	-	82.3±1.32	83.9±0.93	79.4±2.23	77.3±1.57	84.3±1.69
Xception [3]	-	-	-	-	-	83.5±0.75	89.9±0.53	80.7±0.79	81.6±0.45	92.0±0.78
Capsule [6]	-	-	-	-	-	84.6± 0.12	84.7±0.13	84.6±0.13	87.9±0.58	81.4±0.62
CNN+LSTM	89.2±0.23	88.7±0.31	86.8±0.59	89.4±0.61	87.3±0.45	-	-	-	-	-
CNN+GRU	91.2±0.65	89.9±0.27	86.6±0.74	88.6±0.53	92.4±0.39	-	-	-	-	-
Ours	**95.6±0.29**	**99.0±0.11**	**93.2±0.63**	**94.3±0.33**	**98.2±0.67**	**94.8±0.25**	**98.4±0.23**	**95.9±0.73**	**94.7±0.53**	99.4±0.64

[a] DSP-FWA is tested on its recommended resolution.

methods are trained on low-resolution data with size 64×64. We also conduct an experiment comparing our method using our joint loss with our method using focal loss [27]. In addition, we train Xception [3] using our proposed joint loss to illustrate the efficacy of the AUC loss.

4.2.3. Implementation details

We use ResNet-50 [17] as the backbone of our method. The ResNet-50 model is initialized with the pretrained model on the ImageNet. Other layers are initialized randomly with a Gaussian distribution of mean 0 and standard deviation 0.01, with the biases being initialized to 0. In all experiments, we use the Adam optimizer for training. The learning rate is set to 1×10^{-4} and the minibatch is set to 3000 for the frame-level methods and 16 for the video-level methods. All models are trained with 20 epochs, and all experiments are performed on four NVIDIA Tesla P100 GPUs, each with 16 GB of onboard memory.

For the selection of the hyperparameter values in Eq. (3), we fix p to 2 according to the description of [11] and determine γ on the evaluation of FaceForensics++ by using the validation set of FaceForensics++. As a result, γ is set to 0.15 as the default value. Additionally, to explore whether the default value of γ is still optimal for the subsets from Celeb-DF (Celeb-10/20/30), we adjust γ from 0 to 1 with a step of 0.1 using the validation set of FaceForensics++ since Celeb-DF does not include a validation set. As default values, we set $\alpha = 0.5$ in Eq. (4) to achieve balanced performance between accuracy (ACC) and AUC and $\beta = 0.5$ in Eq. (5) for equal importance of the video-level performance and the frame-level performance.

4.2.4. Evaluation metrics

We use ACC, AUC, F1 score (F1), recall (R), and precision (P) as the evaluation metrics for both video-level and frame-level evaluations. For the experimental results on FaceForensics++ and Celeb-DF, we report the average and standard deviation for each method that has been trained 5 times in Sections 4.3.1 and 4.3.2. Since the standard deviation is small, we use the best model for each method in the remaining experiments.

4.3. Evaluation on public datasets

4.3.1. Evaluation on FaceForensics++

We first evaluate our model's performance on FaceForensics++. This is a balanced dataset. Table 2 shows the average and standard deviation of the ACC, AUC, F1, recall, and precision scores of our method and the comparison methods on FaceForensics++. Compared with the best-performing comparison method, our model scores at least 10% higher on ACC, 8% higher on AUC, 11% higher on F1, and 6% higher on precision for the frame-level detection and at least 4% higher on ACC, 9% higher on AUC, 6% higher on F1, 4% higher on precision, and 5% higher on recall for the video-level detection. The frame-level recall of DSP-FWA is slightly higher than ours (100 vs 99.4), and both are at least 7% higher than the other frame-level methods. We conclude that our method significantly outperforms all the comparison methods on every performance metric for both video-level and frame-level detection except the frame-level recall.

4.3.2. Evaluation on Celeb-DF

We also evaluate the performance on Celeb-DF dataset. Celeb-DF is a considerably imbalanced dataset that fake videos significantly outnumber real videos, which is the opposite of the real-world distribution. Table 3 shows the average and standard deviation of the ACC, AUC, F1, recall, and precision scores on Celeb-DF of our model and the comparison methods. In these experiments, the original imbalanced data is used for training and testing for all the methods without extra class weight adjustment or sampling steps. Similar to the results on FaceForensics++, our method significantly outperforms all the comparison methods on every performance metric for both video-level and frame-level detection except the frame-level recall: compared with the best-performing comparison method, our model scores at least 3% higher on ACC, 6% higher on AUC, 4% higher on F1, and 5% higher on recall for the frame-level detection and at least 4% higher on ACC, 9% higher on AUC, 2% higher on F1, 5% higher on precision, and 4% higher on recall for the video-level detection. The frame-level recall of DSP-FWA is slightly higher than ours (99.3 vs 98.8), and both are at least 15% higher than the other frame-level methods. Table 3 indicates that our model achieves great performance on this imbalanced dataset, with an AUC score above 97% for both frame-level and video-level detection, which is significantly higher than those of the comparison methods in both cases. This proves the robustness of our proposed joint loss function for this type of imbalanced data.

We can also see from both tables that our method achieves similar outstanding performance for both frame-level and video-level detection on these two datasets, one is a balanced dataset, and the other is an imbalanced dataset heavily favoring DeepFake videos. These results indicate the power and robustness of our joint loss function in handling datasets of these two types of distributions.

To provide more details on how our model detects DeepFake videos, Fig. 3 shows Grad-CAM [37] visualizations of video frames detected by our model on Celeb-DF and FaceForensics++, where images framed in red are fake video frames and their Grad-CAM visualizations while images framed in green are real video frames and their Grad-CAM visualizations. In Fig. 3, we can see that the attention areas are concentrated around boundary regions between a face and its background, which suggests that our model detects frame forgeries by revealing face boundary artifacts.

4.3.3. Cross-dataset evaluation

We conduct a cross-dataset evaluation to evaluate our method's generalization ability and compare it with the frame-level meth-

Table 3
Performance results of different detection methods on Celeb-DF.

Method	Video Level					Frame Level				
	ACC	AUC	F1	Precision	Recall	ACC	AUC	F1	Precision	Recall
DSP-FWA [5][a]	-	-	-	-	-	65.3	50.0	79.1	64.8	99.3
Meso4 [2]	-	-	-	-	-	72.0±0.99	83.0±1.65	35.9±4.17	93.5±1.23	22.3±3.27
MesoInception4 [2]	-	-	-	-	-	85.3±1.53	89.7±2.11	76.3±3.80	88.1±2.41	66.4±5.41
Xception [3]	-	-	-	-	-	93.6±0.15	91.4±0.26	89.9±0.59	97.9±0.21	83.7±0.62
Capsule [6]	-	-	-	-	-	91.0±0.35	88.5±0.26	86.2±0.51	93.2±0.92	80.2±0.63
CNN+LSTM	87.4±0.23	85.5±0.35	89.2±0.54	90.1±0.82	92.7±0.66	-	-	-	-	-
CNN+GRU	92.3±0.17	89.9±0.37	93.2±0.45	91.7±0.76	94.9±0.68	-	-	-	-	-
Ours	**96.5±0.19**	**98.9±0.45**	**95.6±0.34**	**96.4±0.69**	**98.6±0.72**	**96.2±0.26**	**97.4±0.29**	**94.3±0.56**	**98.2±0.58**	98.8±0.66

[a] DSP-FWA is tested on its recommended resolution.

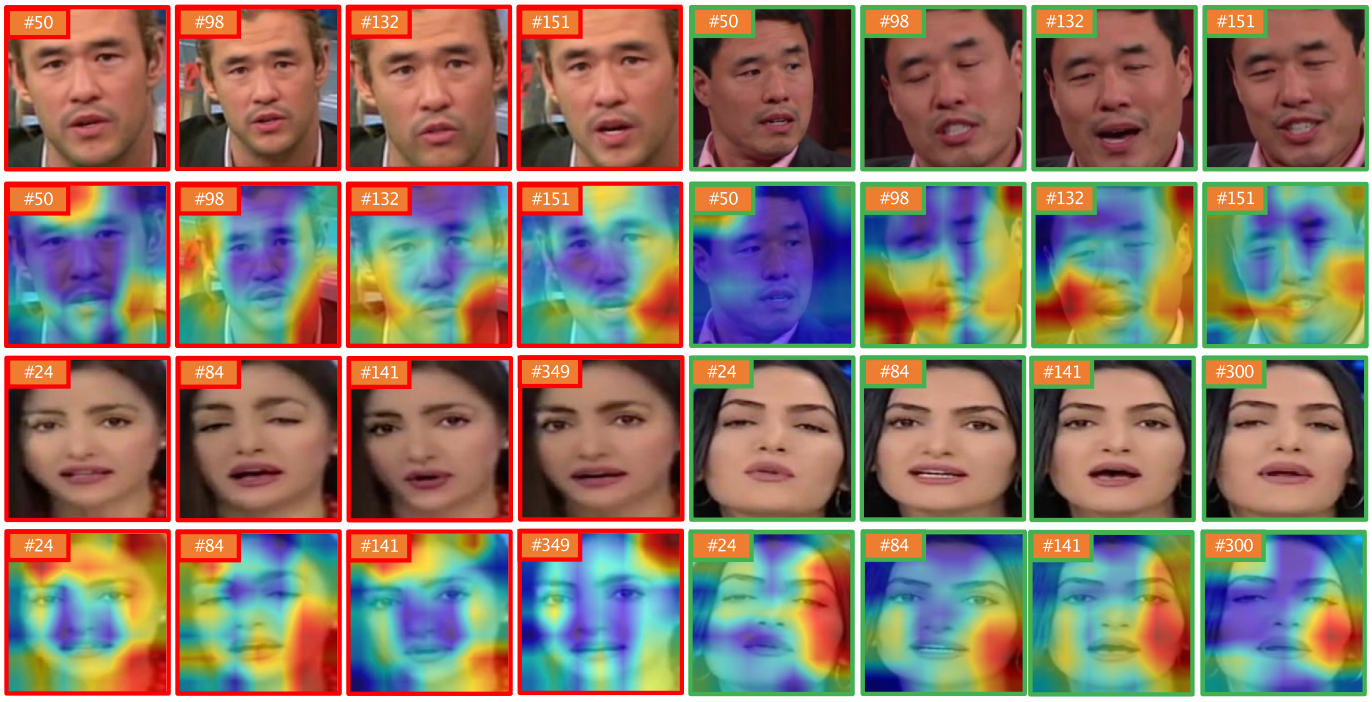

Fig. 3. Grad-CAM visualizations of Celeb-DF (first two rows) and FaceForensics++ (last two rows) samples. Images framed in red boxes (left) are **fake** video frames and their Grad-CAMs. Images framed in green boxes (right) are **real** video frames and their Grad-CAMs. The Grad-CAMs suggest that our model detects frame-level DeepFakes by revealing face boundary artifacts.

Table 4
Cross-dataset evaluation of frame-level AUC from FaceForensics++ to Celeb-DF.

Method	Input Size	Frame-level AUC	
		FaceForensics+	Celeb-DF
Meso4 [2]	64	73.9	62.1
MesoInception4 [2]		86.3	57.9
Xception [3]		91.2	55.4
Capsule [6]		87.3	58.7
Ours		**99.2**	**70.3**
Two-branch [10]	224	93.2	73.4
Face X-ray [23]	256	99.2	74.8
High-frequency [21]	256	99.3	79.4

ods listed in Section 4.2.2 except DSP-FWA. As mentioned before, a pretrained model of DSP-FWA is used in experiments reported previously in Sections 4.3.1 and 4.3.2. Its results reported in Tables 2 and 3 are already cross datasets. As a result, we exclude DSP-FWA in the current cross-dataset evaluation.

All the methods are trained on FaceForensics++ c23 and transferred to Celeb-DF. Table 4 reports the experimental results using 64 × 64 low-resolution data, which also includes the performance results on the original FaceForensics++ dataset for completeness. We can see that our method achieves frame-level AUC 70.3% on Celeb-DF, outperforming the other frame-level detection methods by at least 8%. We have also evaluated the cross-dataset performance of our method with a larger input size of 128 × 128. Our method achieves frame-level AUC at 71.7% on Celeb-DF and 99.4% on the original FaceForensics++ dataset, improving by 1.4% and 0.2%, respectively, compared with the data resolution at 64 × 64 reported in Table 4. These results indicate that a larger input size helps improve our method's cross-dataset performance.

For further comparison, Table 4 also shows the AUC performance of other state-of-the-art methods reported in their papers since we have no access to their code. These methods are trained and tested with much higher resolution data. We can see from the table that our method is quite competitive compared with these state-of-the-art methods' performance on much higher resolution data, which proves the generalization ability of our method and the efficacy of our joint loss.

4.3.4. Robustness analysis

With advances in compression algorithms and network technologies, the quality of videos in social media networks has grad-

Table 5
Robustness results on FaceForensics++: trained on c23 and tested on raw, c23, and c40.

Method	Video-level AUC			Frame-level AUC		
	raw	c23	c40	raw	c23	c40
DSP-FWA	-	-	-	51.0	51.4	50.3
Meso4	-	-	-	74.6	73.9	73.1
MesoInception	-	-	-	86.3	86.3	83.0
Xception	-	-	-	91.0	91.2	87.8
Capsule	-	-	-	86.8	87.3	84.2
CNN+LSTM	87.6	88.3	82.2	-	-	-
CNN+GRU	88.6	89.4	84.3	-	-	-
Ours	**99.2**	**99.4**	**95.4**	**98.9**	**99.2**	**95.2**

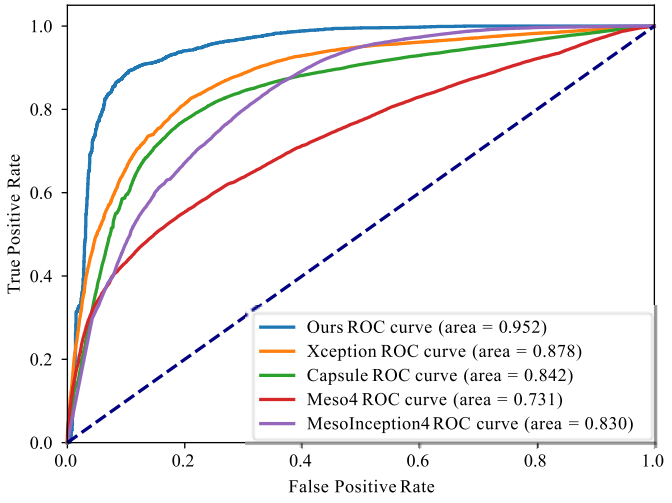

Fig. 4. The ROC curves and AUC values of different frame-level methods tested on FaceForensics++ c40.

ually improved. Nevertheless, low-quality videos are still widely used on social media networks. A detection method should be robust to videos of varying quality. To evaluate our method's robustness in detecting DeepFake videos at different levels of quality, we use FaceForensics++ datasets with different levels of compression. We train our model and the comparison models with c23 (the medium-quality version) and test them with both raw (the original version) and c40 (the lowest-quality version). Table 5 shows the AUC results of our method and the comparison methods, including the results on c23.

As shown in the table, the AUC score on raw is similar to that on c23 for every method. This is because the input image size is only 64 × 64, resulting in the quality of c23 nearly the same as that of raw. Compared with raw and c23, the AUC score of our model on low-quality dataset c40 is slightly lower but still outstanding. Our method achieves an AUC score greater than 95% for both frame-level and video-level detection when tested on the three datasets, at least 10% higher than the best result of the comparison video-level detection methods and 7% higher than the best result of the comparison frame-level detection methods. We conclude that our model significantly outperforms the comparison methods on datasets of different levels of quality for both frame-level and video-level detection.

To provide more details on the AUC performance of these methods, we plot the ROC curves of the comparison frame-level methods and our method on c40 in Fig. 4. The ROC curves in this figure confirm that our method outperforms the comparison frame-level methods.

We also compare the classification results of our method and the comparison frame-level methods on c23 and c40 of FaceForensics++. Figure 5 shows some examples and their detection results. The figure indicates that image quality may have a significant impact on a model's decision: lower quality tends to adversely impact a model's predictions. It also indicates that our method is much more robust to variations in image quality than the comparison frame-level methods.

Table 6
Ablation study results on Celeb-DF with the frame-level classifier (FLC), video-level classifier (VLC), temporal learning module (TLM), and AUC loss removed.

Method	Video-level		Frame-level	
	ACC	AUC	ACC	AUC
CNN +FLC	-	-	74.5	68.2
CNN +FLC,+AUC loss	-	-	81.0	84.9
CNN +VLC	66.1	52.0	-	-
CNN +VLC,+TLM	91.9	89.9	-	-
CNN +VLC,+TLM,+FLC	97.2	96.0	94.9	93.4
CNN +VLC,+TLM,+FLC,+AUC loss	**96.6**	**99.4**	**95.8**	**98.3**

4.3.5. Ablation study

We evaluate the effect of our method's multitask structure through an ablation study on the Celeb-DF dataset. Table 6 shows the experimental results of the ablation study. We first study the effect of the AUC loss function. When the AUC loss is removed (CNN +VLC,+TLM,+FLC), the AUC score decreases by 3.4% for video-level detection and 4.9% for frame-level detection. These results indicate that our AUC loss boosts the AUC score for both frame-level and video-level detection.

We further remove the frame-level classifier (FLC). The resulting network (CNN +VLC,+TLM) can predict only video-level forgeries. Table 6 shows that the video-level ACC and AUC scores decrease by 5.3% and 6.1%, respectively. These results indicate that the FLC reinforces the video-level classifier (VLC) performance and that they boost each other's performance.

Then, we further remove the temporal learning module (TLM) and leave only the CNN encoder and VLC for detection (CNN +VLC). The resulting performance becomes very poor: 66.1% for ACC and 52.0% for AUC, corresponding to a significant decrease of 25.8% for ACC and 37.9% for AUC as compared with the results with the TLM. These results indicate that the TLM plays an extremely important role in video-level detection and that temporal information is important when predicting whether a video is a forgery.

Finally, by removing the VLC, TLM, and AUC loss, our method degrades to the ResNet-50 (CNN +FLC). Compared to our model without AUC loss, the frame-level scores of ACC and AUC decrease by 20.4% and 25.2%, respectively, which shows the importance of TLM and indicates that VLC and FLC can boost each other's performance. In addition, by removing both VLC and TLM from our model (CNN +FLC,+AUC loss), the performance achieves 81.0% for ACC and 84.9% for AUC, which are close to some specially designed frame-level methods such as MesoInception4 and Capsule. It demonstrates the effectiveness of the proposed AUC loss.

4.4. Evaluation of imbalanced learning performance

We evaluate the detection performance of our proposed method and the comparison methods when facing imbalanced data in real-world applications. To mimic real-world situations in which real videos significantly outnumber DeepFake videos, we use three subsets of Celeb-DF, i.e., Celeb-10, Celeb-20, and Celeb-30, and two subsets of DFDC, i.e., DFDC-100 and DFDC-200 (see Section 4.1 for details). For these subsets, the ending number in the names of these subsets means the ratio of real videos to fake videos. On these subsets from Celeb-DF, we not only report the performance

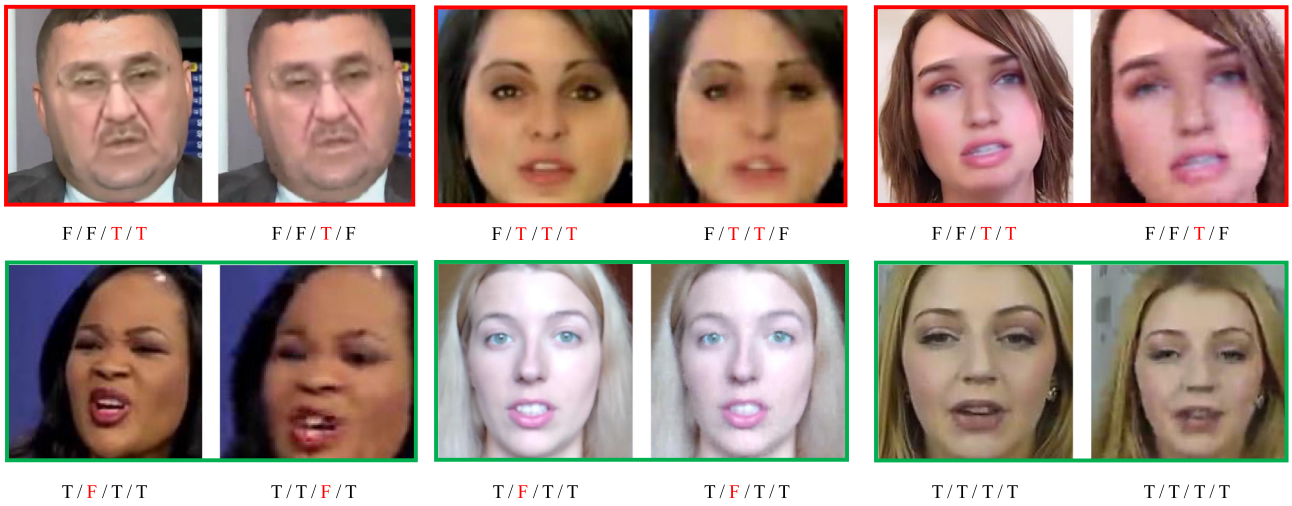

Fig. 5. Comparison of the frame-level classification results on c23 and c40 of FaceForensics++. The top images are all **fake** images, and the bottom images are all **real** images. Each group contains two images: the left is from c23, and the right is from c40. The prediction results of different frame-level methods are listed under each image in the following order from left to right: our method/Capsule/MesoInception4/Xception. "T" and "F" indicate the model predictions. False predictions are highlighted in red. The results show that our method is robust to the quality variation in FaceForensics++.

Table 7
Performance results on Celeb-30 (P means precision and R means recall).

Method	Video Level					Frame Level				
	ACC	AUC	F1	P	R	ACC	AUC	F1	P	R
DSP-FWA [5]	-	-	-	-	-	92.2	57.4	96.2	94.8	94.8
Meso4 [2]	-	-	-	-	-	97.3	56.5	98.4	95.3	100
MesoInception4 [2]	-	-	-	-	-	97.3	50.0	98.6	96.2	100
Capsule [6]	-	-	-	-	-	95.1	55.9	96.9	98.4	97.1
Xception [3]	-	-	-	-	-	97.3	50.0	98.1	97.3	100
Xception w AUC loss	-	-	-	-	-	97.3	54.8	99.2	96.6	100
BBN [28]	-	-	-	-	-	95.4	50.0	97.1	97.3	97.8
CNN+LSTM	94.9	58.7	98.8	95.3	98.6	-	-	-	-	-
CNN+GRU	97.1	59.6	97.1	97.0	97.0	-	-	-	-	-
Ours w/o AUC loss	96.2	70.4	98.0	96.2	98.3	95.7	72.1	98.0	95.3	98.3
Ours w Focal Loss [27]	97.1	72.4	92.4	96.6	98.9	91.0	62.1	97.8	96.7	98.8
Ours (default γ)	97.2	**76.9**	98.6	96.7	100	95.1	78.0	98.6	96.7	100
Ours ($\gamma = 0.1$)	97.2	73.7	98.6	96.7	100	95.1	**78.3**	98.6	96.7	100

of our method obtained with the default γ value ($\gamma = 0.15$), but also report the results of using the optimal γ on these three subsets by adjusting γ from 0 to 1 with a step of 0.1 using the validation set of FaceForensics++ since Celeb-DF does not include a validation set. The optimal γ is selected as 0.1 for Celeb-30, 0.6 for Celeb-20, and 0.3 for Celeb-10. The reason why we adjust the hyperparameter γ in this subsection is that we want to explore whether the default value of γ is still suitable for these subsets.

4.4.1. Performance evaluation on Celeb-30

Celeb-30 has the most skewed distribution among the three subsets sampled from Celeb-DF and includes only 23 fake videos in the training set. The experimental results of our model and the comparison methods are listed in Table 7. For this extremely data-imbalanced situation, the AUC score of our model with $\gamma = 0.1$ achieves 73.7% for video-level detection and 78.3% for frame-level detection, both are significantly lower than those on FaceForensics++ and Celeb-DF reported in Tables 2 and 3. The comparison methods also have much lower AUC scores than those in Tables 2 and 3: approximately a decrease of 50% for the frame-level methods and of 60% for the video-level methods. In terms of the AUC score, our method with $\gamma = 0.1$ outperforms the best comparison video-level method by 14.1% (59.6% vs. 73.7%) and the best comparison frame-level method by 21.0% (57.4% vs. 78.4%).

It is not surprising to see that the other performance metrics on Celeb-30, such as ACC and precision, are improved over those on Celeb-DF and FaceForensics++ reported in Tables 2 and 3 for both frame-level and video-level detection, especially of the comparison methods. This is because these performance metrics are dominated by the overwhelming number of real videos in this extremely data-imbalanced dataset. These results again indicate that the AUC score is a more effective performance metric and that the traditional cross-entropy loss function that minimizes classification errors results in poor performance when facing very skewed data distributions.

Table 7 includes the performance results when the Xception method is modified by adding our AUC loss to its original cross-entropy loss. The AUC score increases by 4.8%, from 50.0% to 54.8%. The table also shows the performance results of our model trained with focal loss [27] by using hyperparameter γ[5] fixed to 2, as recommended in [27], and $\alpha = 0.6$ which achieved the best performance on the validation set of FaceForensics++.

Note that we also adjust the value of α in focal loss using the validation set of FaceForensics++ for experiments on Celeb-20 and Celeb-10 in Sections 4.4.2 and 4.4.3. Although the hyperparameter in focal loss is optimized, its AUC score is 1.3% and 16.2% lower

[5] This γ is a hyperparameter of focal loss, which is unrelated to the γ in Eq. (3) of our AUC loss function.

Table 8
Performance results on Celeb-20 (P means precision and R means recall).

Method	Video level					Frame Level				
	ACC	AUC	F1	P	R	ACC	AUC	F1	P	R
DSP-FWA [5]	-	-	-	-	-	96.8	78.8	98.3	97.1	98.4
Meso4 [2]	-	-	-	-	-	96.4	75.2	98.2	94.1	100
MesoInception4 [2]	-	-	-	-	-	93.0	68.8	96.3	95.3	95.3
Capsule [6]	-	-	-	-	-	95.6	66.6	97.7	97.1	98.3
Xception [3]	-	-	-	-	-	93.8	64.3	97.0	96.9	97.1
Xception w AUC loss	-	-	-	-	-	95.6	78.8	97.7	96.3	100
CNN+LSTM	70.3	84.4	81.7	71.0	69.1	-	-	-	-	-
CNN+GRU	87.6	87.2	93.1	86.9	87.6	-	-	-	-	-
Ours w/o AUC loss	96.2	90.1	98.1	95.4	98.8	95.7	87.7	98.1	96.2	98.8
Ours w Focal Loss [27]	95.7	94.9	98.2	97.2	99.0	95.7	92.6	98.3	97.3	99.5
Ours (default γ)	95.7	95.2	97.8	95.7	100	95.7	92.7	97.8	95.7	100
Ours ($\gamma = 0.6$)	95.7	**95.4**	97.8	95.7	100	95.7	**92.9**	97.8	95.7	100

than our method for video-level detection and frame-level detection, respectively, which shows the superiority of our AUC loss function over focal loss in addressing very skewed data distributions.

Table 7 also includes the performance results of BBN [28], which are obtained by using the best pretrained model in [28] fine-tuned on Celeb-30 with traditional cross-entropy loss. It achieves high frame-level ACC and precision scores (95.4% and 97.3%, respectively) but gives poor AUC performance, which means that BBN has a poor ability to classify DeepFake videos when facing skewed data distributions in the real world. This phenomenon is also observed in other comparison frame-level methods.

In addition, Table 7 reports the performance results of our method with the AUC loss removed. Its AUC score is 3.3% and 6.2% lower than our method with the joint loss (labeled as ours in Table 7) for video-level and frame-level detection, respectively, but is still much higher than the comparison video-level and frame-level methods. These results indicate that the special structure of our method also helps improve AUC performance when facing very imbalanced data. We also report the results of our method obtained with the default γ value on Celeb-30. Compared with the results of our method obtained with $\gamma = 0.1$, using the default γ value 0.15 on Celeb-30 gives a comparable performance, which shows that our method is robust to hyperparameter γ.

4.4.2. Performance evaluation on Celeb-20

Celeb-20 is less skewed in the data distribution than Celeb-30, with 35 DeepFake videos, 1.5 times more than Celeb-30. Table 8 shows the experimental results on Celeb-20. Compared with the results on Celeb-30 shown in Table 7, the AUC score is improved significantly for every method. Our method with $\gamma = 0.6$ achieves an AUC score of 95.4% for video-level detection and and 92.9% for frame-level detection, while the highest AUC score of the existing methods is 87.2% for video-level detection and 78.8% for frame-level detection. Our method still significantly outperforms existing video-level and particularly frame-level methods.

Table 8 also includes the experimental results of several variations of our proposed method. The AUC performance of our method achieved with the default γ is still comparable to the performance of our method with $\gamma = 0.6$, which proves the robustness of our AUC loss to hyperparameter γ. The AUC score of our model without the AUC loss is 5.3% and 5.2% lower than that of our method with $\gamma = 0.6$ for video-level and frame-level detection, respectively. When our AUC loss is added to Xception's original cross-entropy loss, the AUC score increases by 14.5%. These results demonstrate the efficacy of our AUC loss in improving the AUC score. Table 8 includes the experimental results of our method with its AUC loss replaced with focal loss [27] with hyperparameter $\alpha = 0.6$. We can see from the table that this variant of our method achieves similar performance as our method, and both achieve the best performance on Celeb-20.

4.4.3. Performance evaluation on Celeb-10

Celeb-10 is the least skewed dataset among the three datasets sampled from Celeb-DF. It contains 71 DeepFake videos, twice as many as Celeb-20. Table 9 shows the experimental results on Celeb-10. From the table, we can see that the AUC score of our method with $\gamma = 0.3$ is 14.9% and 9.1% higher than the best existing video-level and frame-level methods, respectively, and our AUC loss improves the AUC score of our method by 3.2% over using focal loss [27] with $\alpha = 0.3$ and by 14.5% for Xception on frame-level detection. These results confirm the power of our AUC loss in improving the AUC score on Celeb-10. The performance of our method obtained with default γ is closed to the performance obtained with $\gamma = 0.3$. This shows that our method is robust to hyperparameter γ.

4.4.4. Impact of hyperparameter γ on imbalanced datasets from celeb-DF

We also study the impact of hyperparameter γ of the AUC loss in Eq. (3) on the detection performance on subsets from Celeb-DF: Celeb-10, Celeb-20, and Celeb-30. The experimental results are shown in Fig. 6. We can see in the figure that the AUC scores of both video-level and frame-level detection vary, with similar trends, when γ changes from 0.0 to 1.0 with a step of 0.1. When γ increases from 0, the AUC performance of both frame-level and video-level detection improves until reaching the best performance, potentially maintains the performance roughly around the best for a small range, and then drops significantly when γ passes a certain threshold in a rough range between 0.4 and 0.6. Despite variations of the AUC performance on different values of γ, our method, even at a non-optimal value of γ, still achieves a much higher AUC score than the existing methods, which use the traditional cross-entropy loss function, for both video-level and frame-level detection. This indicates the power of our AUC loss on improving the AUC performance. Besides that, the AUC performance based on the optimal γ value obtained using the validation set of FaceForensics++ is close to the AUC performance corresponding to the optimal γ value shown in Fig. 6, which demonstrates the applicability of using the validation set of FaceForensics++ to select the γ value for Celeb-30, Celeb-20, and Celeb-10.

4.4.5. Performance evaluation on imbalanced subsets from DFDC

To fully evaluate our method's imbalanced learning performance, we also generate more skewed subsets with ratios of positive samples to negative samples of 1:100 and 1:200 from DFDC [13], denoted DFDC-100 and DFDC-200, respectively, to evaluate

Table 9
Performance results on Celeb-10 (P means precision and R means recall).

Method	Video Level					Frame Level				
	ACC	AUC	F1	P	R	ACC	AUC	F1	P	R
DSP-FWA [5]	-	-	-	-	-	94.2	82.3	96.8	97.2	96.7
Meso4 [2]	-	-	-	-	-	91.4	72.2	95.5	90.1	100
MesoInception4 [2]	-	-	-	-	-	88.4	54.2	91.8	89.3	92.1
Capsule [6]	-	-	-	-	-	85.0	63.4	91.2	92.2	89.6
Xception [3]	-	-	-	-	-	85.9	60.0	91.5	93.4	91.2
Xception w AUC loss	-	-	-	-	-	91.0	74.5	94.8	91.4	100
CNN+LSTM	85.1	78.6	91.4	84.3	86.5	-	-	-	-	-
CNN+GRU	92.1	78.6	94.7	93.5	97.1	-	-	-	-	-
Ours w/o AUC loss	88.2	87.3	92.9	87.9	91.6	88.9	88.2	92.9	88.2	91.6
Ours w Focal Loss [27]	90.3	90.5	94.3	93.4	96.4	90.0	90.3	95.4	92.5	95.7
Ours (default γ)	91.3	92.5	95.4	91.2	100	91.3	**93.2**	95.4	91.2	100
Ours ($\gamma = 0.3$)	91.3	**93.5**	95.4	91.2	100	91.3	91.4	95.4	91.2	100

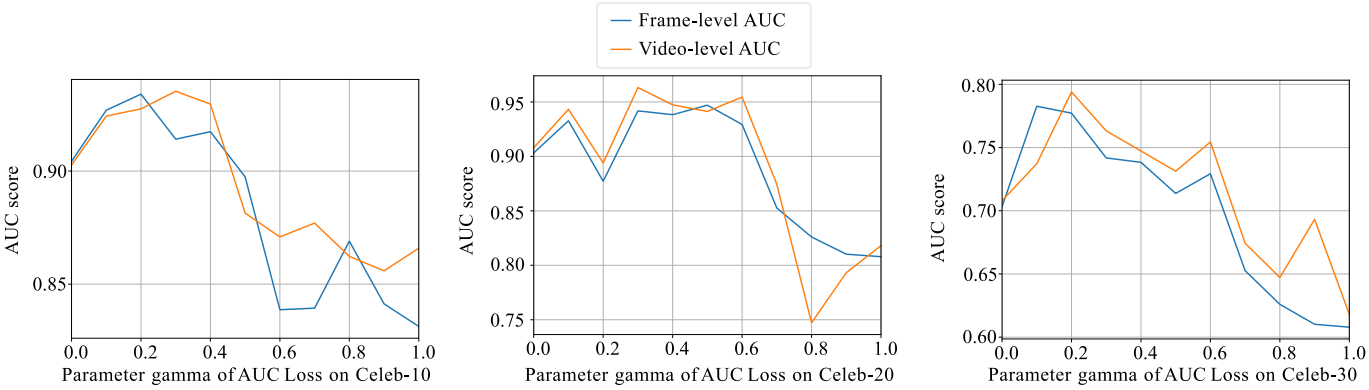

Fig. 6. Dependence of the AUC score on γ in Eq. (3) for both video-level and frame-level detection.

Table 10
Performance results on DFDC-100 (P means precision and R means recall).

Method	Video level					Frame Level				
	ACC	AUC	F1	P	R	ACC	AUC	F1	P	R
DSP-FWA [5]	-	-	-	-	-	99.1	50.0	98.7	98.7	98.5
Meso4 [2]	-	-	-	-	-	88.5	50.6	93.8	99.0	90.2
MesoInception4 [2]	-	-	-	-	-	99.0	63.7	99.2	99.3	100
Capsule [6]	-	-	-	-	-	98.5	50.0	98.7	99.4	100
Xception [3]	-	-	-	-	-	98.7	50.0	98.7	99.0	100
CNN+GRU	98.6	50.0	98.6	99.2	100	-	-	-	-	-
Ours w/o AUC loss	98.7	65.6	98.5	99.3	98.9	99.1	65.5	98.9	98.8	100
Ours	98.7	**79.9**	**99.1**	99.1	100	99.1	**81.6**	99.0	99.2	100

the performance of our method and compare with the existing frame-level methods and a video-level method, CNN+GRU, that are used in the previous experiments. Furthermore, we also compare our method without the AUC loss (BCE only) to show the impact of the AUC loss and to assess the ability of our method's architecture to handle imbalanced learning. Since the AUC loss (see Eq. (2)) requires both negative and positive samples in a training mini-batch, we apply a weighted probability in selecting samples for each mini-batch when the AUC loss is used to ensure that at least one positive sample is selected. Other settings are the same as in the previous experiments on imbalanced datasets from Celeb-DF.

Table 10 shows the experimental results on DFDC-100. Our method achieves an AUC score of 81.6% at the frame level, 17.9% higher than then best AUC score of 63.7% achieved by MesoInception among the comparison frame-level methods. Our method without the AUC loss achieves an AUC score of 65.6% at video-level detection, 15.6% higher than CNN+GRU, and of 65.5% at frame-level detection, a little higher than the best comparison frame-level method.

The experimental results on DFDC-200 are shown in Table 11. Our method achieves an AUC score of 81.4% at the frame level and of 85.6% at the video level. The former is about the same as its counterpart on DFDC-100 while the latter is 5.7% higher than its counterpart on DFDC-100. Like the results on DFDC-100, our method achieves significantly higher AUC scores on DFDC-200 at both frame-level and video-level detection than the existing frame-level methods and video-level methods.

The experimental results on DFDC-100 and DFDC-200 prove that both our joint loss and our architecture improve the AUC performance of imbalanced learning. On the other hand, Tables 10 and 11 show that all methods achieve great ACC, F1, precision, and recall scores on both DFDC-100 and DFDC-200, which illustrates the robustness of AUC as an performance metric when facing excessively skewed data distributions.

4.4.6. Analysis of imbalanced learning performance

Based on the above experiments on five subsets with different ratios of positive to negative samples to simulate real-world

Table 11
Performance results on DFDC-200 (P means precision and R means recall).

Method	Video level					Frame Level				
	ACC	AUC	F1	P	R	ACC	AUC	F1	P	R
DSP-FWA [5]	-	-	-	-	-	99.3	50.0	99.2	99.0	100
Meso4 [2]	-	-	-	-	-	98.8	50.0	98.8	98.7	100
MesoInception4 [2]	-	-	-	-	-	98.8	61.5	98.5	98.6	100
Capsule [6]	-	-	-	-	-	98.9	50.1	99.1	99.4	100
Xception [3]	-	-	-	-	-	98.5	50.0	99.0	99.4	100
CNN+GRU	98.8	50.0	98.5	99.1	100	-	-	-	-	-
Ours w/o AUC loss	92.9	62.6	96.6	99.4	94.2	91.3	62.2	96.3	98.7	91.7
Ours	98.9	**85.6**	**98.6**	99.0	100	98.5	**81.4**	**99.3**	98.6	100

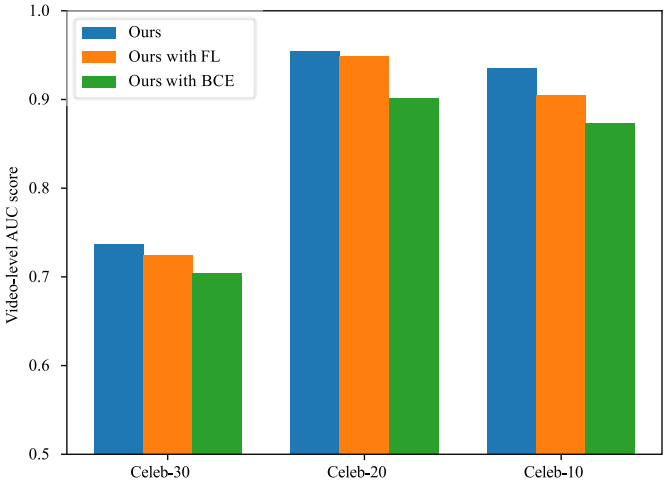

Fig. 7. Comparison of video-level performance of our model ("Ours"), our model with focal loss that replaces our AUC loss ("Ours with FL"), and our model without the AUC loss ("Ours with BCE") on imbalanced subsets from Celeb-DF. The results confirm the efficacy of our proposed joint loss.

scenarios of imbalanced data, we can draw the following conclusions. First, our joint loss function (i.e., adding the AUC loss function to the traditional cross-entropy loss function) can effectively boost AUC performance for imbalanced data, as indicated by the AUC score gaps between using and without using the AUC loss in our model as well as in Xception on the three imbalanced subsets from Celeb-DF. Second, our joint loss outperforms state-of-the-art focal loss [27], which is widely used to address the data imbalance problem, as our model with our joint loss outperforms our model with the AUC loss replaced by focal loss in most cases. Figure 7 shows the comparison of video-level performance between our model with the joint loss ("Ours"), our model with the AUC loss replaced by focal loss ("Ours with FL"), and our model without AUC (i.e., BCE loss only, "Ours with BCE") on the imbalanced subsets from Celeb-DF. Third, even without using the AUC loss, our method outperforms existing video-level methods and frame-level methods, which shows the power and robustness of our multitask structure in addressing imbalanced data.

5. Conclusion

In this paper, we presented an effective method to detect DeepFake videos at both the frame level and video level for datasets of variable distributions, especially excessively skewed data distributions such as those found in the real world. More specifically, we proposed a dual-level collaborative framework based on multitask learning to simultaneously detect DeepFakes at both the frame level and the video level and proposed a joint loss function that combines AUC loss and cross-entropy loss to optimize the AUC performance and minimize the adverse impact of imbalanced distributions. We have conducted an extensive experimental evaluation of our method and compared with existing state-of-the-art frame-level and video-level detection methods. The experimental results indicate that our proposed method outperforms all existing methods at both the frame level and video level and is more robust to video quality and dataset variations.

Our proposed method is restrictive in some situations. In this paper, we focus on the detection of DeepFakes. As a future work, we would like to generalize our method to other types of facial manipulated images and videos, such as GAN-generated images and videos. Moreover, our joint loss function includes two hyperparameters that need to be manually selected, we would also like to learn the hyperparameters automatically during the training process in the future.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by National Key Research and Development Program of China (2020YFA0608001), National Natural Science Foundation of China (61602065), Sichuan Science and Technology Program (2020JDTD0020) and Sichuan Province Key Technology Research and Development project (2021YFG0038).

References

[1] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, M. Nießner, Face2Face: real–time face capture and reenactment of RGB videos, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2387–2395.
[2] D. Afchar, V. Nozick, J. Yamagishi, I. Echizen, MesoNet: a compact facial video forgery detection network, in: 2018 IEEE International Workshop on Information Forensics and Security (WIFS), 2018, pp. 1–7, doi:10.1109/WIFS.2018.8630761.
[3] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, M. Nießner, FaceForensics++: learning to detect manipulated facial images, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1–11.
[4] F. Chollet, Xception: deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.
[5] Y. Li, S. Lyu, Exposing DeepFake videos by detecting face warping artifacts, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019.
[6] H.H. Nguyen, J. Yamagishi, I. Echizen, Capsule-forensics: using capsule networks to detect forged images and videos, in: ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 2307–2311, doi:10.1109/ICASSP.2019.8682602.
[7] H. He, E.A. Garcia, Learning from imbalanced data, IEEE Trans. Knowl. Data Eng. 21 (9) (2009) 1263–1284.
[8] D. Güera, E.J. Delp, DeepFake video detection using recurrent neural networks, in: 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2018, pp. 1–6, doi:10.1109/AVSS.2018.8639163.
[9] X. Li, Y. Lang, Y. Chen, X. Mao, Y. He, S. Wang, H. Xue, Q. Lu, Sharp multiple instance learning for DeepFake video detection, in: Proceedings of the 28th ACM International Conference on Multimedia, 2020, pp. 1864–1872.

[10] I. Masi, A. Killekar, R.M. Mascarenhas, S.P. Gurudatt, W. AbdAlmageed, Two-branch recurrent network for isolating DeepFakes in videos, in: European Conference on Computer Vision, Springer, 2020, pp. 667–684.
[11] L. Yan, R.H. Dodier, M. Mozer, R.H. Wolniewicz, Optimizing classifier performance via an approximation to the Wilcoxon-Mann-Whitney statistic, in: Proceedings of the 20th International Conference on Machine Learning (ICML-03), 2003, pp. 848–855.
[12] R. Collobert, J. Weston, A unified architecture for natural language processing: deep neural networks with multitask learning, in: Proceedings of the 25th International Conference on Machine Learning, 2008, pp. 160–167.
[13] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, C.C. Ferrer, The DeepFake detection challenge (DFDC) dataset, 2020,
[14] Faceswap, Accessed July 1, 2021 (https://github.com/MarekKowalski/FaceSwap).
[15] J. Thies, M. Zollhöfer, M. Nießner, Deferred neural rendering: image synthesis using neural textures, ACM Trans. Graph. (TOG) 38 (4) (2019) 1–12.
[16] N. Liu, T. Zhou, Y. Ji, Z. Zhao, L. Wan, Synthesizing talking faces from text and audio: an autoencoder and sequence-to-sequence convolutional neural network, Pattern Recognit. 102 (2020) 107231.
[17] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[18] X. Zhu, H. Wang, H. Fei, Z. Lei, S.Z. Li, Face forgery detection by 3D decomposition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 2929–2939.
[19] U.A. Ciftci, I. Demir, L. Yin, FakeCatcher: detection of synthetic portrait videos using biological signals, IEEE Trans. Pattern Anal. Mach. Intell. (2020) 1, doi:10.1109/TPAMI.2020.3009287.
[20] Z. Shang, H. Xie, Z. Zha, L. Yu, Y. Li, Y. Zhang, PRRNet: pixel-region relation network for face forgery detection, Pattern Recognit. 116 (2021) 107950.
[21] Y. Luo, Y. Zhang, J. Yan, W. Liu, Generalizing face forgery detection with high--frequency features, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16317–16326.
[22] J.H. Bappy, C. Simons, L. Nataraj, B.S. Manjunath, A.K. Roy-Chowdhury, Hybrid LSTM and encoder-decoder architecture for detection of image forgeries, IEEE Trans. Image Process. 28 (7) (2019) 3286–3300, doi:10.1109/TIP.2019.2895466.
[23] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, B. Guo, Face X-ray for more general face forgery detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 5001–5010.
[24] Y. Li, M.-C. Chang, S. Lyu, In Ictu Oculi: exposing AI created fake videos by detecting eye blinking, in: 2018 IEEE International Workshop on Information Forensics and Security (WIFS), 2018, pp. 1–7, doi:10.1109/WIFS.2018.8630787.
[25] J. Hernandez-Ortega, R. Tolosana, J. Fierrez, A. Morales, DeepFakesON-Phys: DeepFakes detection based on heart rate estimation, in: Proc. 35th AAAI Conference on Artificial Intelligence Workshops, 2021.
[26] X. Liu, J. Wu, Z. Zhou, Exploratory undersampling for class-imbalance learning, IEEE Trans. Syst. Man Cybern.Part B 39 (2) (2009) 539–550, doi:10.1109/TSMCB.2008.2007853.
[27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2980–2988.
[28] B. Zhou, Q. Cui, X.-S. Wei, Z.-M. Chen, BBN: bilateral-branch network with cumulative learning for long-tailed visual recognition, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 9716–9725.
[29] D.E. King, Dlib-ml: a machine learning toolkit, J. Mach. Learn. Res. 10 (2009) 1755–1758.
[30] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, ACL, 2014.
[31] Y. Li, X. Yang, P. Sun, H. Qi, S. Lyu, Celeb-DF: a large-scale challenging dataset for DeepFake forensics, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3207–3216.
[32] DeepFakes, Accessed July 1, 2021 (https://github.com/deepfakes/faceswap).
[33] S. Sabour, N. Frosst, G.E. Hinton, Dynamic routing between capsules, in: Advances in Neural Information Processing Systems, vol. 30, Curran Associates, Inc., 2017, pp. 3856–3866.
[34] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: 3rd International Conference on Learning Representations, ICLR, 2015.
[35] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (9) (2015) 1904–1916, doi:10.1109/TPAMI.2015.2389824.
[36] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, T. Darrell, Long-term recurrent convolutional networks for visual recognition and description, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 2625–2634.
[37] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad–CAM: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.
[38] H. Guo, S. Hu, X. Wang, M.-C. Chang, S. Lyu, Robust attentive deep neural network for detecting gan-generated faces, IEEE Access 10 (2022) 32574–32583.
[39] H. Guo, S. Hu, X. Wang, M.-C. Chang, S. Lyu, Open-eye: An open platform to study human performance on identifying ai-synthesized faces, International Conference on Multimedia Information Processing and Retrieval.
[40] H. Guo, S. Hu, X. Wang, M.-C. Chang, S. Lyu, in: Eyes tell all: Irregular pupil shapes reveal gan-generated faces, IEEE, 2022, pp. 2904–2908.
[41] S. Hu, Y. Li, S. Lyu, in: Exposing gan-generated faces using inconsistent corneal specular highlights, IEEE, 2021, pp. 2500–2504.
[42] S. Suwajanakorn, S.M. Seitz, I. Kemelmacher-Shlizerman, Synthesizing obama: learning lip sync from audio, ACM Transactions on Graph-ics (TOG) 36 (4) (2017) 1–13.
[43] X. Wang, H. Guo, S. Hu, M.-C. Chang, S. Lyu, Gan-generated faces detection: A survey and new perspectives (2022), arXiv preprintarXiv:2202.07145
[44] S. Hu, Y. Ying, S. Lyu, et al., Learning by minimizing the sum of ranked range, Advances in Neural Information Processing Systems 33 (2020) 21013–21023.
[45] S. Hu, Y. Ying, X. Wang, S. Lyu, Sum of ranked range loss for supervised learning, Journal of Machine Learning Research 23 (112) (2022) 1–44.

Wenbo Pu received the BEng degree in Computer Science from the Chengdu University of Information Technology, China, in 2019. He is currently pursuing the MS degree with the School of Computer Science, Chengdu University of Information Technology. His current research interests include DeepFake detection, computer vision, and deep reinforcement learning.

Jing Hu (Member, IEEE) received the bachelor's degree from the University of Electronic Science and Technology of China, in 2009, and the PhD degree from Tsinghua University, in 2015. She is currently a full professor with the Chengdu University of Information Technology. Her current research interests include image processing, deep learning, and reinforcement learning.

Xin Wang is currently a Senior Machine Learning Scientist at the Keya Medical. He received his PhD degree in Computer Science from the University at Albany, State University of New York in 2015. His research interests are in artificial intelligence, machine learning, optimization, computer vision and medical image computing. He is a senior member of IEEE.

Yuezun Li is currently an assistant professor in the institute of Artificial Intelligence, at Ocean University of China. He received PhD degree in computer science at University at Albany, SUNY in 2020. His research interest is mainly focused on computer vision and multimedia forensics.

Shu Hu received his Ph.D. degree in Computer Science and Engineering from University at Buffalo, the State University of New York (SUNY) in 2022. He received his M.A. degree in Mathematics from University at Albany, SUNY in 2020, and M.Eng. degree in Software Engineering fromyi University of Science and Technology of China in 2016. His research interests include machine learning, digital media forensics, and computer vision.

Bin Zhu received the BS degree in physics from the University of Science and Technology of China, Hefei, China, in 1986, and the MS and PhD degrees in electrical engineering from the University of Minnesota, Minneapolis, MN, in 1993 and 1998, respectively. He is currently a Principal Researcher with Microsoft Research Asia, Beijing, China. His research interests include machine learning and multimedia processing.

Rui Song is faculty member of the Department of Statistics at North Carolina State University. Her current research interests include Machine Learning, Causal Inference, Precision Health, Knowledge Graph. Her research has been continuously supported as sole principle investigator by National Science Foundation (NSF). She received the prestigious NSF Faculty Early Career Development (CAREER) Award in 2016. She has published over 80 papers in top tier journals of statistics and ML conferences. She is an elected fellow of the American Statistical Association and Institute of Mathematical Statistics.

Qi Song received the bachelor's degree from the University of Electronic Science and Technology, in 2003, the master's degree from Tsinghua University, in 2006, and the PhD degree from The University of Iowa, USA, in 2011. He was a Research Assistant with The University of Iowa, in 2011. He became a Scientist of the Global Research and Development Center, General Electric Company, New York, in 2014. He was a Senior Scientist of HeartFlow Corporation, USA, from 2014 to 2015, and a Founder and the CEO of Keya Medical, Seattle, USA, from 2016 to 2017. Since 2017, he has been the Founder and the General Manager of Shenzhen LE Medical Technology Company Ltd.

Siwei Lyu is an Empire Innovation Professor at the Department of Computer Science and Engineering and the founding Director of UB Media Forensic Lab (UB MDFL) of University at Buffalo, State University of New York. Before joining UB, Dr. Lyu was an Assistant Professor from 2008 to 2014, a tenured Associate Professor from 2014 to 2019, and a Full Professor from 2019 to 2020, at the Department of Computer Science, University at Albany, State University of New York. From 2005 to 2008, he was a Post-Doctoral Research Associate at the Howard Hughes Medical Institute and the Center for Neural Science of New York University. He was an Assistant Researcher at Microsoft Research Asia (then Microsoft Research China) in 2001. Dr. Lyu received his PhD degree in Computer Science from Dartmouth College in 2005,

and his MS degree in Computer Science in 2000 and B.S. degree in Information Science in 1997, both from Peking University, China. Dr. Lyu's research interests include digital media forensics, computer vision, and machine learning. Dr. Lyu has published over 150 refereed journal and conference papers. Dr. Lyu's research projects are funded by NSF, DARPA, NIJ, UTRC, IBM and Department of Homeland Security. He is the recipient of the IEEE Signal Processing Society Best Paper Award (2011), the National Science Foundation CAREER Award (2010), SUNY Albany's Presidential Award for Excellence in Research and Creative Activities (2017), SUNY Chancellor's Award for Excellence in Research and Creative Activities (2018) and Google Faculty Research Award (2019). Dr. Lyu currently serves on the IEEE Signal Processing Society's Information Forensics and Security Technical Committee, and is on the Editorial Board of IEEE Transactions on Information Forensics and Security. Dr. Lyu is a senior member of IEEE, a member of ACM, a member of Sigma Xi, and a member of Omicron Delta Kappa.