Pattern Recognition 163 (2025) 111469

Contents lists available at ScienceDirect

Pattern Recognition

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

Multi-definition Deepfake detection via semantics reduction and cross-domain training☆

Cairong Zhao [a],*,[1], Chutian Wang [a,1], Zifan Song [a], Guosheng Hu [b], Liang Wang [c], Duoqian Miao [a]

[a] Department of Computer Science and Technology, Tongji University, Shanghai, 201804, China
[b] Oosto, Belfast, United Kingdom
[c] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China

ARTICLE INFO

Keywords:
Deepfake detection
Deep learning
Self-supervised learning
Face Anti-Spoofing

ABSTRACT

The recent development of Deepfake videos directly threatens our information security and personal privacy. Although lots of previous works have made much progress on the Deepfake detection, we empirically find that the existing approaches do not perform well on the low definition (LD) and cross-definition (high and low) videos. To address this problem, in this paper, we follow two motivations: (1) high-level semantics reduction and (2) cross-domain training. For (1), we propose the Facial Structure Destruction and Adversarial Jigsaw Loss to reduce our model to learn high-level semantics and focus on learning low-level discriminative information; For (2), we propose an adversarial domain generalization method and a spatial attention distillation which uses the information of HD videos to guide LD videos. We conduct extensive experiments on public datasets, FaceForensics++ and Celeb-DF v2. Results show the great effectiveness of our method and we also achieve very competitive performance against state-of-the-art methods. Surprisingly, we empirically find that our method is also very effective on Face Anti-Spoofing (FAS) task, verified on OULU-NPU dataset.

1. Introduction

Since the first Deepfake Video was published on *Reddit* in 2017, multiple Deepfake methods based on Variational Auto-Encoders (VAE) [1] and Generative Adversarial Networks (GAN) [2] have been developed successively. Unlike traditional video face manipulation methods based on hand-crafting or computer graphics technology like Face2Face [3], these Deepfake methods train deep neural networks for video face manipulation, thus the visual artifacts of these fake videos are much tinier than previous methods, making it difficult to be observed by human eyes. Existing studies [4] have shown that human observers perform poorly on detecting these videos. Therefore, the Deepfake videos generated using this technique can easily spread false information, and threaten our information security and personal privacy seriously. Fortunately, a couple of Deepfake detectors (e.g. FaceForensics++ [4], Celeb-DF [5] and DFDC [6]) have been developed and have achieved great results on many public datasets.

However, experiments on the c40 compressed version of FaceForensics++ dataset [4] expose a serious problem that current approaches suffer a significant performance drop on low-definition (LD) videos. It is known that the low-level facial textural details are more discriminative than high-level semantics (e.g. holistic facial shape [7]) for Deepfake detection. As shown in Fig. 1, however, (1) the LD videos significantly lose the textural details while keep the high-level semantics, causing major challenges for Deepfake detection. (2) Furthermore, we empirically find that it is even harder to train a model which can generalize well to both HD and LD videos (cross-domain robustness) detailed in Section 4.3.2.

To address the aforementioned two problems, in this work, we propose two corresponding solutions: (1) We propose a Facial Structure Destruction (FSD) module and Adversarial Jigsaw Loss to reduce our model to learn high-level semantics and focus on learning low-level discriminative information. Specifically, we creatively introduce image

☆ This work was supported by National Natural Science Fund of China (62076184, 61673299, 61976160, 62076182), in part by Shanghai Innovation Action Project of Science and Technology, China (20511100700) and by Shanghai Natural Science Foundation of Shanghai, China (22ZR1466700); and in part by Shanghai Municipal Science and Technology Major Project, China (2021SHZDZX0100) and the Fundamental Research Funds for the Central Universities, China.

* Corresponding author.
E-mail addresses: zhaocairong@tongji.edu.cn (C. Zhao), 1652303@tongji.edu.cn (C. Wang), sugger@tongji.edu.com (Z. Song), huguosheng100@gmail.com (G. Hu), wangliang@nlpr.ia.ac.cn (L. Wang).

[1] Cairong Zhao and Chutian Wang contributed equally to this work.

https://doi.org/10.1016/j.patcog.2025.111469
Received 18 June 2022; Received in revised form 28 January 2025; Accepted 11 February 2025
Available online 21 February 2025
0031-3203/© 2025 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

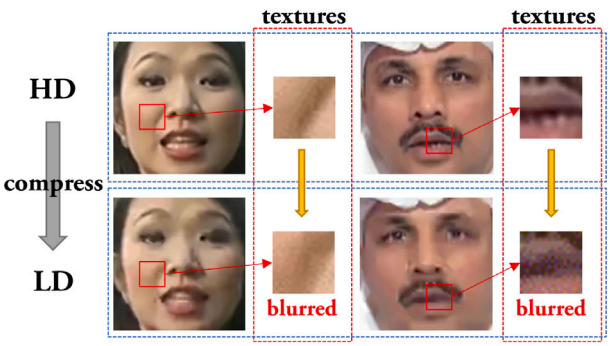


Fig. 1. Examples of face images and their textures. The face images of each column are extracted from the same frame of the same video where each was compressed with factor 23 (high-definition, HD) and factor 40 (low-definition, LD). Obviously, the textures are extremely blurred after compression yet facial structures (semantic information. e.g. facial looking) are consistent.

patch shuffle operation, which can well achieve our target (destroying semantics while preserving textural details). To maximize this 'destruction', inspired by cryptography [8], we propose a score, termed as 'disorder score', to quantify the 'disorder' of image patches. Guided by this disorder score, we can achieve the best (most disordered) patch shuffle pattern for our task. In addition, we propose an Adversarial Jigsaw Loss equipped with the patch shuffle to reduce our model to learn the semantics via self-supervised learning. Specifically, the shuffled pattern of FSD can be considered as a jigsaw puzzle, and we use a jigsaw solving loss, the error between the prediction of a Jigsaw Solver (analogous to a discriminator of Generative Adversarial Networks [2]) and the groundtruth (shuffled pattern). Same as the adversarial alternating optimization of GAN, our model learns as little semantics as possible. (2) We propose the use of domain generalization and knowledge distillation to achieve the cross-domain (HD and LD) robustness. Specifically, we train a Domain Classifier to classify the domain label (HD or LD) of the feature vectors extracted by the Deepfake detector. Guided by a domain adversarial loss, our model learns to be unable to distinguish the domain labels by introducing a domain classifier (a discriminator from a perspective of GAN), leading to a domain-robust feature learning. Furthermore, inspired by knowledge distillation, we introduce the spatial attention module to the model and use the more precise attention maps from HD videos to supervise those from LD videos for training, reducing the domain gap between LD and HD videos.

Our contributions can be summarized as:

- We deeply investigate the generalization capacity of the Deepfake detection solutions from LD to HD videos, and propose a robust Deepfake detection method.
- We propose the Facial Structure Destruction and Adversarial Jigsaw Loss to reduce our model to learn high-level semantics and focus on learning low-level discriminative information, leading to a more discriminative Deepfake detector.
- To learn the cross-domain (HD and LD) robustness, we propose an adversarial domain generalization method. Furthermore, we use the richer information of HD videos to guide the training of LD videos via a Spatial Attention Distillation method.
- We conduct extensive experiments on FaceForensics++ [4] and Celeb-DF v2 [5] dataset. Results show our system achieves very competitive performance against the state-of-the-art methods. Last but not least, surprisingly, we empirically find that our system is also very effective for Face Anti-Spoofing verified on OULU-NPU dataset [9].

Portions of this work were presented at the *IEEE International Conference on Multimedia and Expo (ICME)* in 2022 [10]. Compared to the previous conference article, this paper propose a novel Spatial Attention Distillation method to better perform the cross-domain training. We also conduct additional detailed ablation studies and extensive experiments on more datasets, backbones, and tasks.

This paper is organized as follow: Section 2 introduces the technology and previous works related to this papaer; Section 3 details the methodology of this work including high-level semantics reduction (Section 3.1) and cross-domain training (Section 3.2); Section 4 introduces our extensive experiments; Finally, we conclude our work in Section 5.

2. Related work

Deepfake Video Synthesis. With the rapid development of computer vision technology, face manipulated videos now can automatically be generated by computer algorithms. Thies et al. [3] proposed a real time face reenactment system based on face 3D model estimation and it can easily implement with a simple RGB camera. *FaceSwap* use face alignment, Gauss Newton optimization and image blending to swap the face completely to another person. Fortunately, videos manipulated by these methods have obvious visual artifacts and can be easily detected by human observers. Thus they are relatively less harmful to our society. However, based on deep learning technology, the latest face manipulation methods (i.e. Deepfake methods) use GANs [2] and VAEs [1] to generate much more realistic fake faces. They are much more harmful since they can synthesize fake videos (i.e. Deepfake videos) with tiny visual artifacts included to spread fake information or defame somebody. These videos are difficult to discriminate by human since the artifacts of these Deepfake videos are hidden in the low-level texture information. These methods now can easily be implemented by open source codes or publicly available software such as *FaceApp* and *ZAO*.

Deepfake Detection. Latest research shows that it is difficult for human to identify whether the video is manipulated [4]. To make matters worse, traditional digital forensics methods cannot apply to these videos since the neural network generated faces are near-realistic and their visual artifacts are different from the hand-crafted manipulated videos. Fortunately, many approaches have made great progress in Deepfake detection. An early attempt is MesoNet [11], a shallow CNN devised for mesoscopic analyses. They achieved promising performance on the FaceForensics [12] dataset. Nguyen et al. [13] attempt to use capsule networks to detect Deepfake videos, and also achieve interesting result. Rossler et al. [4] test certain previous methods on their benchmarks, the result shows the Xception [14] network pretrained on ImageNet [15] performs greatly and has achieved over 99% accuracy on raw data. But their further experiments show when the videos are compressed to low definition, the detection accuracy will drop dramatically. Especially for *NeuralTextures* [16] generated videos, the decrease is over 17%. Dang et al. [17] utilize the attention mechanism to localize the manipulated region and improve the performance of the backbone Xception network. In order to improve the robustness for post-processing of detection methods, some methods based on remote visual photoplethysmography (rPPG) have been proposed. Qi et al. [18] use a dual-spatial–temporal attention network to classify the motion-magnified rPPG signal extracted from the video, their method achieves slightly better detection performance than Xception network and shows high robustness for JPEG compression. Li et al. [19] utilize multi-instance learning in Deepfake detection and achieve great success on the performance of video-level detection. [20] is one of the latest Deepfake detection methods based on the spatial–temporal features in the videos which has state-of-the-art detection performance. Ciftci et al. [21] did a detailed study on rPPG based Deepfake detection methods, and their CNN-based classification result shows robustness for Gaussian blur and Median filtering. [7] claims that the artifacts in Deepfake videos exist in multiple places, and propose a multi-attention

mechanism to capture them. Their method achieves success on the HD videos, but performs not promising on the LD videos. However, most of the previous methods focus on the Deepfake detection of HD videos. The problem of the Deepfake detection for compressed low definition (LD) Deepfake videos are rarely mentioned or studied for a long time. Pu et al. [22] manage to use a dual-level collaborative framework and a novel loss function for more robust Deepfake detection.

In recent years, the problem of Deepfake detection on LD videos gradually attracted attention of researchers, and some related works have been published. Hu et al. deeply investigate the algorithm of video compression, and propose to use a frame-temporality two-stream convolutional network [23] to address the problem of LD Deepfake detection. Shang et al. [24] propose a Pixel-Region Relation Network to exploit the spatial relation in face images. Their method achieves significant improvement on the LD Deepfake detection and Multi-definition Deepfake detection. Compared with these work, our paper propose a well-purpose scheme (i.e. high-level semantics reduction) to improve LD Deepfake detection and achieves significantly better performance (detailed in Section 4). We also propose a cross domain training strategy to improve multi-definition detection performance. Which is rarely motioned in previous work.

3. Methodology

The Deepfake detection can simply be considered as a binary classification task. According to the winner's solution of the Deepfake Detection Challenge (DFDC) [6], ImageNet [15] pretrained networks show superior performance on this task, hence we directly use them as our backbone network. In practice, we find that these models work well on high-definition (HD) videos, but do not perform well on low-definition (LD) and multi-definition (the mixture of HD and LD) videos in the wild. To achieve a model which can generalize well to both HD and LD videos, we propose to improve the robustness of the detector from the following two directions: (1) using Facial Structure Destruction and Adversarial Jigsaw Loss to reduce the less discriminative high-level semantics and improve the more discriminative low-level texture information; (2) applying methods of domain generalization and Spatial Attention Distillation to improve the multi-definition video detection performance. Based on (1) and (2), we propose a triple-branch model to conduct Deepfake detection. By applying (1), the LD video Deepfake detection performance can be significantly improved (detailed in 4.3.1). Moreover, by applying (1) and (2), our method can achieve promising performance on both HD and LD videos (detailed in 4.3.2).

3.1. High-level semantics reduction

Current Deepfake video synthesis methods have made great progress of improving visual effect and imitating realistic face structure. Since the synthetic faces bear strong resemblance with those real faces, the high-level semantics (e.g. facial structure, gender, race, beauty, etc.) are not very discriminative to distinguish the real and attack faces. Hence the low-level textural details are more discriminative than the high-level semantics for Deepfake detection [7].

3.1.1. Facial structure destruction

In this work, we propose a Facial Structure Destruction (FSD) mechanism to destroy the high-level semantic facial structure. As shown in Fig. 3, we creatively introduce patch shuffling operation to achieve the target: destroying global high-level semantics while preserving low-level facial textural details. Patch shuffle can destroy the high-level semantics, however, it is not clear how to conduct the patch shuffle that can achieve the greatest 'destruction'. Inspired by the concept of chaos in cryptography [8], we propose a disorder score to quantify the 'disorder' of a patch-shuffle image. Specifically, the input image I is evenly divided into $N \times N$ patches and shuffled by a random pattern

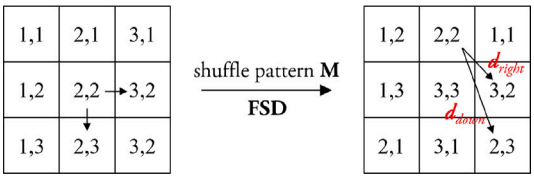


Fig. 2. FSD operation and the disorder score.

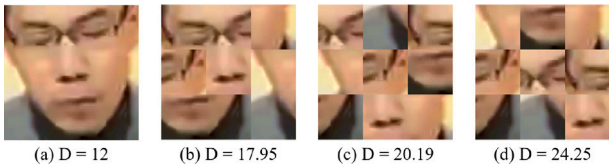


Fig. 3. Examples of the FSD operation and the corresponding disorder score D. From (a) to (d), the disorder score D increases and the facial structures are destructed to a greater degree.

$\mathbf{M} \in \{1, 2, \ldots, N\}^{2\times N\times N}$ to generate the shuffled image $\psi_N(I)$, where $\mathbf{M}_{x,y} = [i, j]^T$ means the image patch at position $[x, y]^T$ is moved to the new position $[i, j]^T$. As shown in Fig. 2, we calculate the distance of adjacent patches of each patch after shuffling, and summarize as a disorder score $D(\mathbf{M})$:

$$D(\mathbf{M}) = \sum_{x=1}^{N}\sum_{y=1}^{N} d_{right} + d_{down} \tag{1}$$

$$d_{right} = \begin{cases} \left\|\mathbf{M}_{x+1,y} - \mathbf{M}_{x,y}\right\|_2 & \text{if } x < N \\ 0 & \text{else} \end{cases} \tag{2}$$

$$d_{down} = \begin{cases} \left\|\mathbf{M}_{x,y+1} - \mathbf{M}_{x,y}\right\|_2 & \text{if } y < N \\ 0 & \text{else} \end{cases} \tag{3}$$

Note that d_{left} and d_{up} are not calculated due to the symmetry. We choose some disrupted images and calculate the corresponding disorder score as shown in Fig. 3

Obviously, the shuffled image with a larger disorder score can destroy the facial structures to a higher degree. Thus we only use the patterns with disorder score larger than a static threshold τ. In this work, if $N < 4$, τ is the average score of all the patterns; if $N >= 4$, τ is set to the average score of randomly generated 10000 patterns. However, the capacity of destruction is limited by N, and we can never completely remove the high-semantic information with the maximal N (size of the image), leading to a nearly random image (image is shuffled at the pixel level). On the other hand, a small N ($N = 1$ on the extreme) cannot reduce the high-level semantics. We quantitatively analyze the impact of N in Section 4.3.1.

3.1.2. Adversarial Jigsaw loss

As discussed, since N cannot be too large to most greatly preserve the low-level textural details, simply applying the FSD mechanism is not effective enough. Thus we propose an adversarial training strategy to further achieve high-level semantics reduction.

Following DCL [25], we take the shuffle pattern $\mathbf{M}$ of the FSD operation as the supervision label, and train a network to predict this pattern, like solving a jigsaw puzzle. In our case, we use a simple CNN, Jigsaw Solver, to complete this task. The prediction of the Jigsaw Solver can be formulated as:

$$\mathbf{P}(I, N) = S^N_{\theta_s}(D^{GAP}_{\theta_d}(\psi_N(I))) \tag{4}$$

where $D^{GAP}_{\theta_d}(\psi_N(I))$ is the extracted feature map before the Global Average Pooling (GAP) layer of the backbone detector with network weights θ_d, $S^N_{\theta_s}$ is the Jigsaw Solver network corresponding to $N \times N$ jigsaw puzzle and $\mathbf{P} \in \mathbb{R}^{2\times N\times N}$ is the predicted pattern. Based on

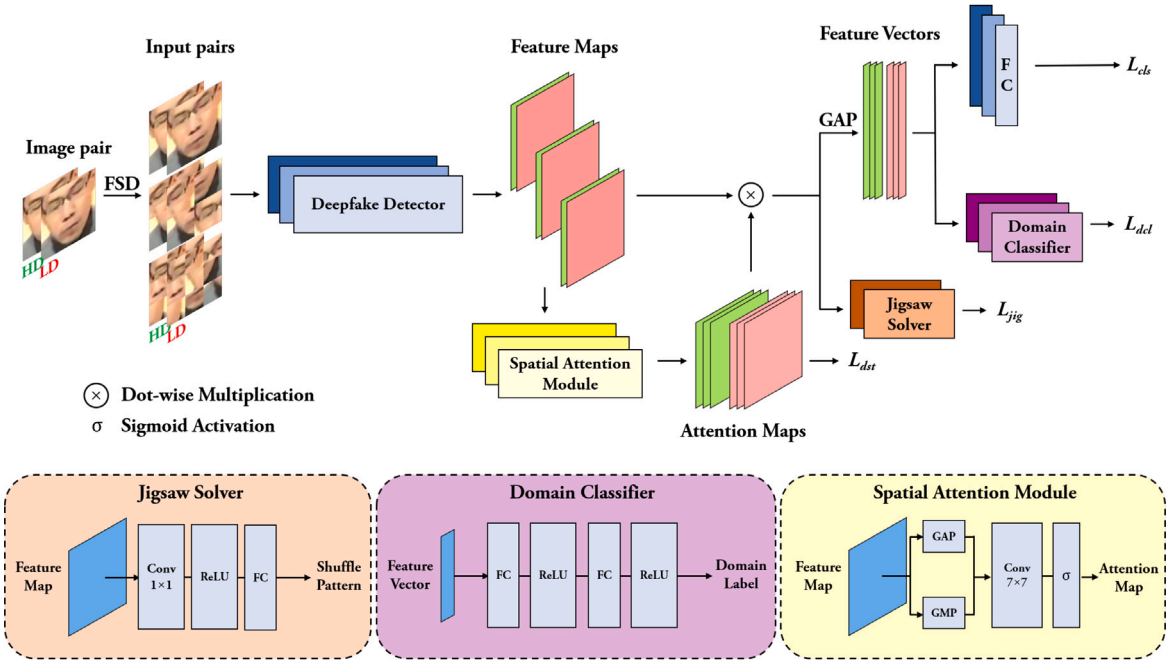


Fig. 4. The pipeline of the proposed method. N of FSD is set to $1, 2, 3$. (1) FSD (Facial Structure Destruction) shuffles the input images; (2) Deepfake Detector consists of 3 branches; (3) Jigsaw Solver is actually a discriminator from the perspective of GAN, aiming to solve the shuffled image; (4) Domain Classifier is another discriminator which helps to learn domain-robust feature. (5) Spatial Attention Module is a convolutional network to indicate the importance of different area. (6) FC: a fully-connected network to conduct binary classification (pristine or fake). The detailed architecture of the Jigsaw Solver, Domain Classifier, and Spatial Attention Module is shown at the bottom. The image pair is only required for training, and only one image (either LD or HD) is required for inference.

the prediction $\mathbf{P}$ and groundtruth $\mathbf{M}$, the jigsaw solving loss can be formulated as:

$$\mathcal{L}_{jig} = \frac{1}{N^2}\sum_{x=1}^{N}\sum_{y=1}^{N}\left\|\mathbf{P}_{x,y} - \mathbf{M}_{x,y}\right\|_2 \tag{5}$$

However, predicting these random shuffle patterns is based on high-level semantic spatial information, which is relatively less discriminative for Deepfake detection. Since our aim is to reduce high-level semantics from the learned representations of the backbone network, we set $\mathcal{L}_{jig}$ as an adversarial loss to the backbone network. It means it is a negative term in the final loss function, thus by the jigsaw loss $\mathcal{L}_{jig}$, the weights θ_s and θ_d are optimized under adversarial learning:

$$\underset{\theta_d}{min}\underset{\theta_s}{max}\mathbb{E}_{I\sim p_{data}(I)} - \lambda_{jig}\cdot\mathcal{L}_{jig} \tag{6}$$

where $p_{data}(I)$ denotes the data distribution of the training set and λ_{jig} is a positive weight for the jigsaw loss.

Same as the adversarial alternating optimization of GAN, our training is also adversarial: (i) the weights of the Deepfake detector are optimized by maximizing the jigsaw solving loss (i.e. preventing solving the jigsaw) (ii) the weights of the Jigsaw Solver are optimized by minimizing the jigsaw solving loss. In this way, we can reduce our Deepfake detector to learn the high-level semantics.

3.2. Cross-domain training

In previous sections, we have proposed two methods to improve the performance of Deepfake detection on the single definition videos. But in the real applications, the videos can be of various resolutions and definitions, which can be regarded as different domains. To achieve the cross-domain robustness of model training, in this work, we propose to apply domain generalization and knowledge distillation for that.

3.2.1. Domain generalization

The domain gap caused by various resolutions and definitions can greatly decrease the model performance. The work [26] verifies this performance drop under (1) training on HD videos and test on LD ones and (2) training on LD videos and test on HD ones. We also have the same empirical observations. We also empirically observe that simply mixing HD and LD data on training does not lead to promising results detailed in Section 4. It inspires us to turn to a smart way to train a cross-definition network.

Motivated by DCL [25], we propose a domain generalization discriminator, Domain classifier, to classify the domains (i.e. video definition) that the training samples are from. Specifically, we randomly sample the HD and LD face images for training and use one-hot labels $\mathbf{c} \in \{0, 1\}^2$ to indicate the domains, our Domain Classifier will predict whether the feature vector $D^{fc}_{\theta_d}(\psi_N(I))$ is extracted from HD or LD video, thus the domain classification loss can be formulated as:

$$\mathcal{L}_{dcl} = -\mathbf{c}\cdot\log\left[C_{\theta_c}(D^{fc}_{\theta_d}(\psi_N(I)))\right] \tag{7}$$

where $C(\cdot)$ denotes the definition prediction and θ_c denotes the network weights of the Domain Classifier.

Since we expect the backbone network to learn domain-invariant representations, the learned representations should be more difficult to classify by the Domain Classifier. Thus the domain classification loss is also an adversarial loss to the weights θ_d, meaning by $\mathcal{L}_{dcl}$, the weights θ_c and θ_d are optimized as follows:

$$\underset{\theta_d}{min}\underset{\theta_c}{max}\mathbb{E}_{I\sim p_{data}(I)} - \lambda_{dcl}\cdot\mathcal{L}_{dcl} \tag{8}$$

where λ_{dcl} is a positive weight of the domain classification loss.

3.2.2. Spatial attention distillation

When videos are been compressed, its visual information are highly disrupted and cause the Deepfake detectors more difficult to examine whether the video is manipulated. As discussed before, the reason is that the Deepfake detectors are poor to find the discriminative textures in LD videos without any constraint. To solve this problem, we devise the Spatial Attention Distillation mechanism, which includes a Spatial Attention Module and an Attention Distillation Loss. Our Spatial Attention Module is a simple Convolutional net borrowed from [27], which

generates attention maps to indicate the important features for Deepfake detection, as shown in Fig. 4. In addition, to learn a domain robust attention map, we are inspired by the pretext-invariant representation learning [28], which is one way of self-supervised learning. [28] aligns the features from original and transformed (with data augmentation) images to be close, leading to a pretext-invariant feature learning. In our Deepfake detection, the HD videos contain rich information, while LD ones miss many useful information, hence the attention maps of the HD videos are more precise than the LD ones. Inspired by [28], we push the attention maps of LD to be close to the HD ones, aiming to make the LD attention map as good as the HD maps. We conduct the knowledge distillation [29] to achieve this. Specifically, given an HD and LD image pair $\langle I_H, I_L \rangle$, we distill the knowledge from the attention maps of HD to LD via minimizing the knowledge distillation loss $\mathcal{L}_{dst}$, formulated as:

$$\mathcal{L}_{dst} = \|\mathbf{Attn}_H - \mathbf{Attn}_L\|_2 \tag{9}$$

where $\mathbf{Attn}_Q$, $Q \in \{H, L\}$ indicates the attention maps of the HD inputs and the LD inputs generated by the Spatial Attention Module, formulated as:

$$\mathbf{Attn}_Q = A^N_{\theta_a}(D^{Mul}_{\theta_d}(\psi_N(I_Q))) \tag{10}$$

where $D^{Mul}_{\theta_d}(\psi_N(I_Q))$ is the extracted feature map of HD or LD inputs before attention multiplication, $A^N_{\theta_a}$ are the three attention modules with weights θ_a to be optimized.

3.3. Triple-branch network architecture

As discussed, the number of patches $N \times N$ of FSD and the corresponding adversarial learning step are important. Too small or too big N will lead to degraded performance. In fact, with different N settings during training, the learned model has its strengths and weaknesses during testing. Thus, we propose a Triple-branch architecture to capture the complementary information. Our experiments in Section 4.3 show that this architecture can significantly improves the performance through the complementarity of the network branches. To implement the Spatial Attention Distillation proposed in Section 3.2.2, we take both the HD and LD versions of a chosen video frame to form a input pair $\langle I_H, I_L \rangle$, and use the same shuffle pattern for FSD before feeding them to the three branches respectively. The frame-level classification loss $\mathcal{L}_{cls}$ can be formulated as:

$$\mathcal{L}_{cls} = -\mathbf{l} \cdot \log \left[\prod_{i=1}^{3} \prod_{Q} D_{\theta^i_d}(\psi_{N_i}(I_Q)) \right] \tag{11}$$

where $\mathbf{l}$ is the video label (pristine or fake), $D_{\theta^i_d}(\psi_{N_i}(I_Q))$ denotes the predicted label of the ith branch for the input with image quality $Q \in \{H, L\}$ and FSD parameter N_i, where N_1 is usually set to 1 to let the first branch takes the original images as inputs, θ_{d_i} is the parameters of the ith Deepfake detector. When testing on a video in the wild with a single resolution version, the prediction l_{pred} of each frame I is:

$$l_{pred} = argmax \left[\sum_{i=1}^{3} D_{\theta^i_d}(I) \right] \tag{12}$$

In addition, we utilize and combine the mechanism proposed in Sections 3.1.2 and 3.2.1 as the adversarial network. Since we use the image pairs as inputs, loss function $\mathcal{L}_{jig}$ and $\mathcal{L}_{dcl}$ are calculated both on the outputs of the HD and LD images, where the ground truth of the Jigsaw Solver will not change because the shuffle pattern of each pair is the same, yet the labels of domain (i.e video definition) are opposite for the pairs. Finally, the total loss of the model is:

$$\mathcal{L}_{total} = \mathcal{L}_{cls} + \lambda_{dst}\mathcal{L}_{dst} - \lambda_{jig}\mathcal{L}_{jig} - \lambda_{dcl}\mathcal{L}_{dcl} \tag{13}$$

where λ_{dst}, λ_{jig} and λ_{dcl} are the corresponding positive weights of each loss. The model parameters are optimized as follows:

$$\min_{\theta_d, \theta_a \theta_s, \theta_c} \max \mathbb{E}_{I \sim p_{data}(I)} \mathcal{L}_{total} \tag{14}$$

where $\theta_d = \{\theta^1_d, \theta^2_d, \theta^3_d\}$ are the parameters of the Deepfake detectors; $\theta_a = \{\theta^1_a, \theta^2_a, \theta^3_a\}$ are the parameters of the Spatial Attention Modules; $\theta_s = \{\theta^1_s, \theta^2_s\}$ are the parameters of the Jigsaw Solvers; $\theta_c = \{\theta^1_c, \theta^2_c, \theta^3_c\}$ are the parameters of the Domain Classifiers.

4. Experiments

4.1. Datasets and metrics

Datasets. We use FaceForensics++ (FF++) [4] and Celeb-DF v2 (CDF) [5] for the Deepfake detection experiments, OULU-NPU [9] for the Face Anti-Spoofing experiments. FaceForensics++ contains 1000 pristine videos collected from *YouTube* and their corresponding manipulated videos created by Deepfakes, FaceSwap, Face2Face [3], and NeuralTextures [16]. Three definition levels of the videos are provided including raw, c23 compressed version (HD), and c40 compressed version (LD). Celeb-DF v2 is a challenging Deepfake dataset which contains 590 pristine videos and 5639 high visual quality Deepfake videos created by improved Deepfake synthesis methods. OULU-NPU is a high-resolution face attack database including photo print attack and video display attack. It has four protocols to evaluate the generalization performance of models. Specifically, Protocol I is designed to evaluate the generalization capacity of various environmental conditions; Protocol II use unseen print and video-replay attack in the test set to evaluate the robustness of the FAS methods; Protocol III examine the generalization of the models on videos recorded with different smartphones; and Protocol IV is the most challenging one with all above three factors are considered (i.e. unseen environmental conditions, attacks and input sensors). These datasets are chosen for the following reasons: (1) They are all large-scale dataset, with millions of well generated fake faces by various methods, hence the generalization capacity of our method can be well tested; (2) Most of the previous works are based on these datasets, hence we can conduct fair comparison with them under same settings.

Metrics. In Deepfake detection experiments, we use video-level detection accuracy, frame-level detection accuracy and average detection accuracy of HD and LD frames as the evaluation metric. In Face-Anti Spoof experiments, following [9], we use the Attack Presentation Classification Error Rate (APCER), Bona Fide Presentation Classification Error Rate (BPCER), and Average Classification Error Rate (ACER) for fair comparisons.

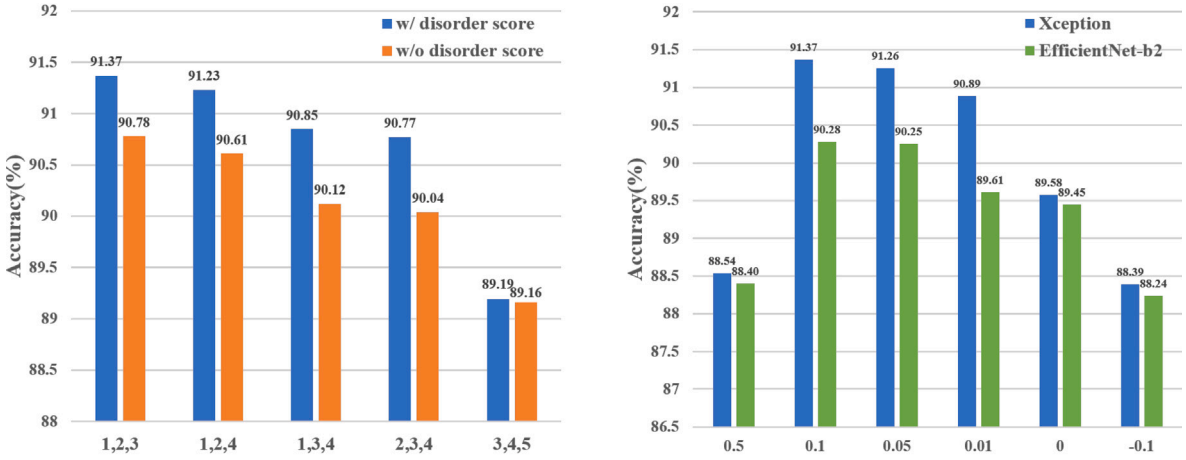
4.2. Settings

Data Preprocessing. We use *face_recognition* python library for face detection and landmark detection. If the facial landmark detection fails, we simply remove that frame to prevent wrong detection. If multiple faces are detected in a single frame, we only choose the biggest one. All the face images are resized to 300 × 300 before training and testing. For FaceForensics++ dataset, we extract 270 faces from each video and use the official splits (720 videos for training, 140 for validation, and 140 for testing) to keep the same with the previous works for fair comparison; For the Celeb-DF v2 dataset, we use the officially provided training and testing split and extract 100 face images from each video. Since Celeb-DF v2 dataset have no official LD version provided, we compress the videos in Celeb-DF v2 dataset to LD version (quality factor is set to 40) by utilizing the codes in [4]. And we regard the raw videos as the HD version.

Implementation Details. We choose the ImageNet pretrained Xception [14] and as our backbone detector. Unless otherwise specified,

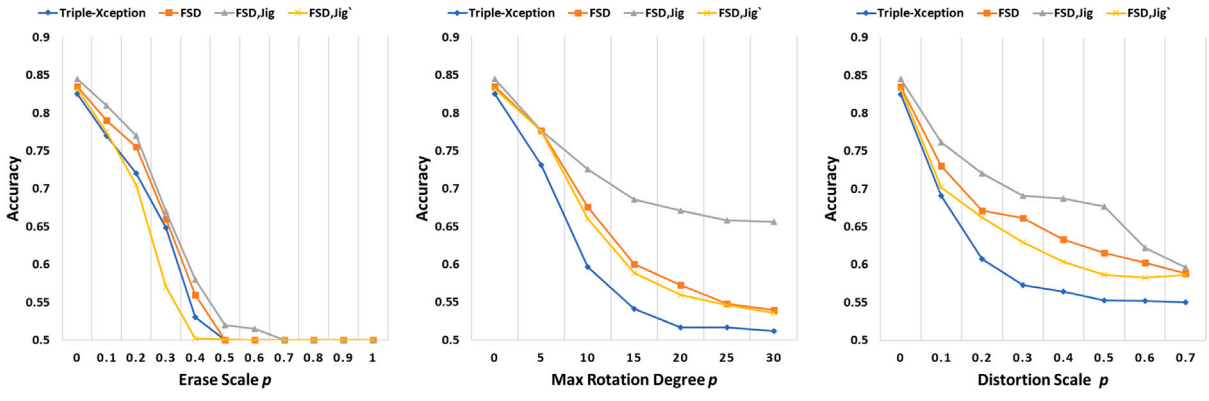
Table 1
The detection accuracy (%) of Single-branch and Triple-branch networks. Triple-branch is the ensemble of three networks. FSD $N = x$: Facial Structure Destruction with $N = x$ is applied. Jig: Adversarial Jigsaw Loss is applied.

Method	Xception				EfficientNet-b2			
	FF++		CDF		FF++		CDF	
	HD	LD	HD	LD	HD	LD	HD	LD
Single-branch	95.73	86.86	96.24	82.76	94.52	86.76	96.07	81.92
+ FSD $N = 2$	95.79	86.94	96.33	83.04	94.75	86.92	96.18	82.21
+ FSD $N = 3$	95.86	86.97	96.39	83.11	94.77	86.98	96.13	82.18
+ FSD $N = 3$, Jig	95.98	87.05	96.84	83.30	94.96	87.02	96.39	82.54
Triple-branch	96.08	87.11	97.30	84.14	95.86	87.09	96.88	83.90
+ FSD $N = \{1,2,3\}$	97.16	89.58	98.06	84.89	96.43	89.45	97.04	84.46
+ FSD $N = \{1,2,3\}$, Jig	**97.79**	**91.37**	**98.62**	**85.72**	**97.52**	**90.28**	**97.35**	**85.64**



(a) Different N settings (b) Different λ_{jig} settings

Fig. 5. The impact of different N settings and different λ_{jig} settings on the LD version of Faceforensics++ dataset. For (a), experiments are conducted on Xception network with λ_{jig} set to 0.1. w/ disorder score means that shuffle patterns are generated guided by the disorder scores and w/o disorder score means that shuffle patterns are randomly generated. For (b), N is set to $\{1,2,3\}$ for all the models.



(a) random erasing (b) random rotation (c) random perspective

Fig. 6. Test accuracy on different augmented data. FSD, Jig': Model trained with $\lambda_{jig} = -0.1$.

the weights of each loss function (i.e. λ_{dst}, λ_{jig}, and λ_{dcl}) are set to 1, 0.1, and 0.2. We use the Adam optimizer to train our models, the learning rates are set to 3e-5, 3e-4, and 3e-3 for the Deepfake detector (including the Spatial Attention Module), Domain classifier, and Jigsaw solver respectively. The Domain Classifier is a 4-layer fully-connected network and the Jigsaw Solver is a CNN including a 1×1 convolution layer and a linear layer. We apply ReLU activation function to these two modules. Our models are trained on the RTX 3090 GPU with batch size set to 32 for up to 80 epochs. We choose the best models based on the validation accuracy.

4.3. Ablation study

The ablation studies in this subsection are conducted on the FaceForensics++ dataset and Celeb-DF v2 dataset. To verify the effectiveness of our methods on different Deepfake detectors, we also conduct experiments on the EfficientNet-b2 [30] network in this section.

4.3.1. Impact of high-level semantics reduction methods

Facial Structure Destruction and Adversarial Jigsaw Loss introduced in Section 3.1 are the proposed high-level semantics reduction methods

in our paper. We apply these mechanisms to the single detector and triple-branch detectors respectively for our ablation study. All the results are reported in Table 1. The results show that both mechanisms can improve the performance on LD and HD videos, and the improvement on LD videos is more significant than on HD videos. This means our high-level semantics reduction methods are very suitable for LD Deepfake detection. Also, our methods are effective on both datasets and both backbone networks, this result demonstrates the universality of our high-level semantics reduction methods.

Since the setting of N is very important for our triple-branch network, we test the impact of N on the performance as shown in Fig. 5(a). The experiments are conducted on the LD version of Faceforensics++ dataset using the Xception networks. The results show that setting N to 1, 2, and 3 for each branch performs the best. Thus, our following experiments will use this setting. Moreover, we test the performance of using randomly generated shuffle patterns (i.e. w/o disorder score in Fig. 5(a)) and compare with that using the patterns generated by the proposed disorder score (i.e. w/ disorder score in Fig. 5(a)). The results show that selecting shuffle patterns based on the disorder score is obviously better than selecting randomly for the FSD operation. We also notice that a too large N setting (e.g. 3, 4, and 5) extremely reduces the performance improvement, this is because the excessive splitting and shuffling may cause the destruction of the critical textures (i.e. Deepfake artifacts), making the detector hard to discriminate the images.

We also test the impact of λ_{jig} on the performance as shown in Fig. 5(b). The results show $\lambda_{jig} = 0.1$ is the best setting. To further show the relationship between Deepfake detection and high-level semantics, we set $\lambda_{jig} = -0.1$ to let backbone detectors focus more on the spatial information. In this setting, the optimization problem becomes:

$$\min_{\theta_d,\theta_a}\max_{\theta_c}\mathbb{E}_{I\sim p_{data}(I)}\mathcal{L}_{total} \tag{15}$$

$$\min_{\theta_s}\mathbb{E}_{I\sim p_{data}(I)}\mathcal{L}_{total} \tag{16}$$

this ensures the jigsaw solver always tries to minimize $\mathcal{L}_{jig}$. In this case, the performance of the detector drops compared to the baseline (88.39% vs. 89.58% for Xception and 88.24% vs. 89.45% for EfficientNet-b2), proves that focusing on the high-level semantics is inappropriate for Deepfake detection. Besides, we notice that when λ_{jig} is too large (i.e. $\lambda_{jig} = 0.5$ in Fig. 5((b)), the performance also drops compared with the baseline. We experimentally find in this scenario, it will be extremely hard for the jigsaw solver to predict the shuffle pattern (i.e. the jigsaw solving loss of the jigsaw solver is always high). In This situation, the jigsaw loss generates unstable gradients, leading to a decline in the performance.

To further verify the robustness of our methods in complex situations (e.g. incomplete faces or warped frames), we apply random erasing, random rotation and random perspective to the testing images (training images are not augmented) in the *NeuralTextures* dataset to simulate these situation. Other subsets of FaceForensics++ dataset are not introduced to reduce the influence of other factors. We choose Xception network as the backbone. The degree of the augmentation is controlled by a parameter p (for erasing, p is the erase scale; for rotation, p is the maximum rotation degree; for perspective, p is the distortion scale) and record the changes of detection accuracy under different p settings. The results shown in Fig. 6 demonstrate that our methods, including FSD and Adversarial Jigsaw Loss, can effectively improve the robustness comparing to the baseline (Triple-Xception). Results also show that the robustness degrades when the model is focused on the high-level semantics by minimizing the jigsaw loss (FSD, Jig', i.e. $\mathcal{L}_{jig}$ is set to -0.1, same as in Fig. 5(b)).

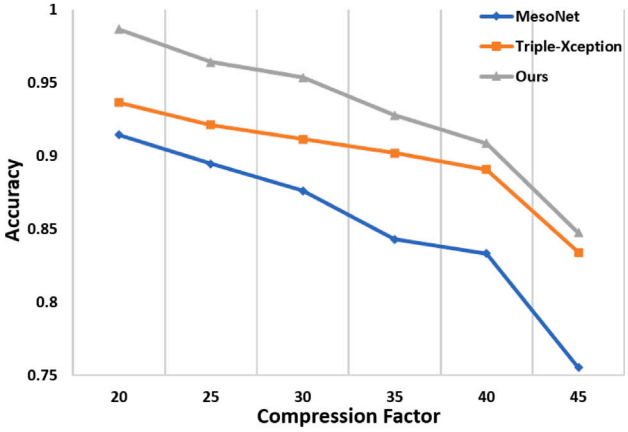


Fig. 7. Detection accuracy on the FaceForensics++ dataset with different compression factors.

4.3.2. Impact of cross-domain training methods

This section investigates the impact of our cross-domain learning strategies: Domain Generalization and Spatial Attention Distillation. We use the triple-branch models trained with semantics reduction methods as baselines and conduct the ablation study on the mixing dataset with both HD and LD version data are included. The models are trained on different domains (i.e HD, LD and Mixed) and tested on each domain respectively. The results are reported in Table 2. The results show that Deepfake detectors trained only on LD videos can be applied to the HD video detection with minor performance drop. The same model trained only on HD videos can achieve superior detection accuracy on HD videos, but perform very poorly on LD videos, leading to a low average accuracy. Simply mixing the HD and LD videos for training can lead to better average accuracy, but the HD and LD accuracy are lower than those trained on data with a single domain due to the domain gap. The results demonstrate our domain generalization method effectively prevents the accuracy degradation of LD videos and thus improves the average accuracy. And our Spatial Attention Distillation further improves the performance on multi-definition videos. Besides, the significant improvement on both datasets (i.e. FaceForensics++ and Celeb-DF) and both backbones (i.e. Xception and EfficientNet-b2) proves that our Cross-Domain Training Methods can be generalized to different practical application scenarios.

4.4. Performance comparison

Deepfake detection on LD videos. In this section, we compare our method with previous Deepfake detection methods on the LD version of each subset of the FaceForensics++ [4] dataset. Each experiment is conducted on one of the subsets. Following [35], we use 720 videos with 270 frames sampled each for network training. Since most of the previous methods only report results trained and tested on a single version of the FaceForensics++ data (HD or LD), in this section, we remove the cross-domain training methods (i.e. Domain Generalization and Spatial Attention Distillation) from our model to avoid introducing extra data. The frame-level and video-level results are shown in Table 3. For video-level detection, we simply use the average result of each frame as the video Deepfake detection result. Our method is competitive with the previous state-of-the-art methods and achieves most significant improvement on the most challenging *NeuralTextures* [16] subset.

Results in Table 3 also show that method using frequency-based features [35] also performs well on the LD videos, this is because: (1) extracting frequency-based features can also force the network focus on the discriminative low-level textures; (2) these frequency-based

Table 2
Cross-domain performance. Mixed videos: the model trained on the mixture of HD and LD videos. DG: adversarial domain generalization loss is utilized. SAD: Spatial Attention Distillation. Best results are bold.

Method	Xception						EfficientNet-b2					
	FaceForensics++			Celeb-DF			FaceForensics++			Celeb-DF		
	HD	LD	Avg	HD	LD	Avg	HD	LD	Avg	HD	LD	Avg
on LD videos	85.01	91.37	88.19	80.48	85.72	83.10	84.80	90.28	87.54	80.16	85.64	82.90
on HD videos	97.79	63.46	80.63	98.62	68.39	83.51	97.52	62.95	80.24	97.35	67.72	82.54
on Mixed Videos	93.26	89.06	91.16	93.55	83.93	88.74	93.11	88.79	90.95	93.40	83.88	88.64
+ DG	96.07	90.16	93.12	97.76	84.62	91.19	96.03	88.90	92.47	96.76	84.22	90.49
+ SAD	96.62	90.33	93.48	98.01	84.41	91.21	96.44	88.93	92.69	96.64	84.49	90.57
+ DG, SAD	96.88	90.86	**93.87**	98.29	85.04	**91.67**	96.78	89.27	**93.03**	96.98	85.17	**91.08**

Table 3
Frame-level and video-level detection accuracy (%) comparing with the previous methods on the LD (c40 compressed) version subsets of the FaceForensics++ dataset and Celeb-DF v2 dataset. Best results are bold.

Method	DeepFakes		Face2Face		FaceSwap		NeuralTextures		Celeb-DF	
	Frame	Video	Frame	Video	Frame	Video	Frame	Video	Frame	Video
Steg.Features [31]	67.00	–	48.00	–	49.00	–	56.00	–	–	–
LD-CNN [32]	75.00	–	56.00	–	51.00	–	62.00	–	–	–
Cons. Conv [33]	87.00	–	82.00	–	74.00	–	74.00	–	–	–
Cus. Pool. [34]	80.00	–	62.00	–	59.00	–	59.00	–	–	–
MesoNet [11]	90.00	–	83.00	–	83.00	–	75.00	–	72.62	73.36
Xception [14]	96.01	97.14	93.29	94.50	94.71	96.07	79.14	86.07	81.92	83.98
F^3-Net [35]	–	98.62	–	**95.84**	–	97.23	–	86.01	–	–
SlowFast [36]	–	97.53	–	94.93	–	95.01	–	82.55	–	–
S-MIL [19]	–	97.14	–	91.43	–	94.64	–	86.79	–	–
S-MIL-T [19]	–	96.79	–	91.07	–	96.07	–	88.57	–	–
STIL [20]	–	98.21	–	92.14	–	97.14	–	91.78	–	–
Frame-Temporality [23]	–	94.64	–	85.27	–	86.48	–	80.05	–	80.74
PRRNet [24]	95.63	–	90.15	–	94.93	–	80.01	–	–	–
Triple-Xception	96.01	97.86	93.42	94.64	96.20	96.43	81.54	87.86	82.26	85.14
Ours	**96.99**	**99.29**	**93.56**	95.00	**96.79**	**97.86**	**84.47**	**92.14**	**85.72**	**90.32**

Table 4
Frame-level detection accuracy comparing with the previous methods on the complete FaceForensics++ dataset. For a fair comparison, we conducted a horizontal comparison among methods based on the same backbone. Best results are bold.

Method	Publication	LD (c40)	HD (c23)
Transformer-based Methods			
TALL-Swin [37]	ICCV'23	**92.82**	98.65
F^2-Trans-S [38]	TIFS'23	90.14	98.14
F^2-Trans-B [38]	TIFS'23	90.57	**98.71**
CNN-based Methods			
Steg.Features [31]	TIFS'12	55.98	70.97
LD-CNN [32]	IH&MMSec'17	58.69	78.45
Cons. Conv [33]	IH&MMSec'16	66.84	82.97
Cus. Pool. [34]	WIFS'17	61.18	79.08
MesoNet [11]	WIFS'18	70.47	83.10
Xception [14]	CVPR'17	86.86	95.73
Xception-ELA [39]	AAAI'21	79.63	93.86
F^3-Net [35]	ECCV'20	90.43	97.52
PRRNet [24]	PR'21	96.15	86.13
Multi-Attention [7]	CVPR'21	86.95	96.37
Triple-Xception	–	87.11	96.08
Ours	–	**91.37**	**97.79**

features are robust to the video compression. However, comparing with [35], our method has following advantages: (1) By High-level Semantics Reduction methods, our method can also achieve promising performance on the LD videos with less time consumption (detailed in Section 6); (2) Our method performs better on the cutting-edge Deepfake manipulation methods (e.g. *NeuralTextures*) by focus more on the fine artifacts hiding in low-level textures; (3) By utilizing Cross-Domain Training methods, our method can be applied to cross-definition videos and running at a real-time speed, thus our method is more suitable for practical usage.

To compare with more state-of-the-art methods and test the detection capacity of our method on various manipulation methods, we also train ad test our method on the complete FaceForensics++ dataset, the results in accuracy on the HD (c23 compressed version) and LD (c40 compressed version) version are reported in Table 4 respectively. The results show our method is also promising on the complete FaceForensics++ dataset, and the improvement is more significant on the LD videos than on the HD videos.

Deepfake detection on cross-definition videos. In practice, the definition and manipulation type of the videos are diverse. Hence, we use the FaceForensics++ benchmark [4] to evaluate our model. With 1000 additional video frames that are randomly compressed and manipulated by different methods, FaceForensics++ benchmark is a publicly available automated benchmark that provides a standardized comparison of different approaches on cross-definition and cross-manipulation videos. We trained our model on the full HD and LD version FaceForensics++ Dataset by $\mathcal{L}_{total}$ and choose the best model on the validation set to run the benchmark. Compared with the published work, our result is reported in Table 5. We achieve the great performance improvement compared with the baseline methods.

To further verify the robustness of our method on unseen definition levels and compare it with other baseline methods, we conduct experiments on the FaceForensics++ dataset that was compressed by different compression factors. The model used for experiments is trained on the training set mixed by c23 (HD) and c40 (LD) videos, the results are shown in Fig. 7. The results show that all the tested methods have promising detection performance when the compression factor is lower than the highest factor used during training (i.e. c40). And our method achieves significant performance improvement compared with the baseline methods. However, the performance significantly drops when the compression factor is higher than the highest factor used during training. To address this, we can use the videos compressed by the highest compression factor (e.g. c50 for the h.264 standard) to ensure that the proposed method can effectively detect Deepfake videos on all the possible video definitions.We also conduct experiments on the

Table 5
Results of the FaceForensics++ automated benchmark.

Method	Accuracy (%)					
	DeepFakes	Face2Face	FaceSwap	NeuralTextures	Real	Total
Steg. Features [31]	73.64	73.72	68.93	63.33	34.00	51.80
LD-CNN [32]	85.45	67.88	73.79	78.00	34.40	55.20
Cus. Pool. CNN [34]	85.45	64.23	56.31	60.07	50.00	58.10
Cons. Conv [33]	84.55	73.72	82.52	70.67	46.20	61.60
MesoNet [11]	87.27	56.20	61.17	40.67	72.60	66.00
Xception [14]	96.36	86.86	90.29	80.67	52.40	70.10
LTW [40]	43.60	48.90	61.20	26.00	82.40	62.90
Triple-Xception	93.60	79.60	81.60	84.00	52.80	68.60
Ours	97.30	91.20	99.00	90.70	52.40	**73.20**

Table 6
Comparison of the time consumption.

Methods	Time consumption		
	Data preprocessing	Model reasoning	Total
Xception [14]	–	7.2 ms/frame	7.2 ms/frame (138.89FPS)
F^3-Net [35]	1586 ms/frame	17.6 ms/frame	1603.6 ms/frame (0.62FPS)
Ours	2.4 ms/frame	23.5 ms/frame	25.9 ms/frame (38.61FPS)

Table 7
Detection accuracy comparing with the previous methods on the Deeperforensics dataset..

Method	Accuracy (%)
Xception [14]	84.5
Multi-task [41]	77.7
FWA [42]	50.2
Face X-ray [43]	89.3
Ours	89.3

Table 8
The results of testing on four protocols of OULU-NPU.

Prot.	Method	APCER(%)	BPCER(%)	ACER(%)
1	GRADIANT [44]	1.3	12.5	6.9
	Auxiliary [45]	1.6	1.6	1.6
	FaceDs [46]	1.2	1.7	1.5
	FAS-TD [47]	2.5	0.0	1.3
	DeepPixBiS [48]	0.8	0.0	0.4
	CDCN++ [49]	0.4	0.0	**0.2**
	Ours	3.7	1.4	2.5
2	DeepPixBiS [48]	11.4	0.6	6.0
	FaceDs [46]	4.2	4.4	4.3
	Auxiliary [45]	2.7	2.7	2.7
	GRADIANT [44]	3.1	1.9	2.5
	FAS-TD [47]	1.7	2.0	1.9
	CDCN++ [49]	1.8	0.8	**1.3**
	Ours	3.4	1.3	2.4
3	DeepPixBiS [48]	11.7 ± 19.6	10.6 ± 14.1	11.1 ± 9.4
	FAS-TD [47]	5.9 ± 1.9	5.9 ± 3.0	5.9 ± 1.0
	GRADIANT [44]	2.6 ± 3.9	5.0 ± 5.3	3.8 ± 2.4
	FaceDs [46]	4.0 ± 1.8	3.8 ± 1.2	3.6 ± 1.6
	Auxiliary [45]	2.7 ± 1.3	3.1 ± 1.7	2.9±1.5
	CDCN++ [49]	1.7 ± 1.5	2.0 ± 1.2	1.8 ± 0.7
	Ours	1.6 ± 1.6	1.5 ± 1.7	**1.5 ± 1.7**
4	DeepPixBiS [48]	36.7 ± 29.7	13.3 ± 14.1	25.0 ± 12.7
	GRADIANT [44]	5.0 ± 4.5	15.0 ± 7.1	10.0 ± 5.0
	Auxiliary [45]	9.3 ± 5.6	10.4 ± 6.0	9.5 ± 6.0
	FAS-TD [47]	14.2 ± 8.7	4.2 ± 3.8	9.2 ± 3.4
	FaceDs [46]	1.2 ± 6.3	6.1 ± 5.1	5.6 ± 5.7
	CDCN++ [49]	4.2 ± 3.4	5.8 ± 4.9	5.0 ± 2.9
	Ours	5.6 ± 3.6	3.2 ± 8.5	**4.4 ± 1.9**

Deeperforensics dataset [52] to test the performance of our methods on compressed cross-definition videos. We report the accuracy and compare it with previous methods in Table 7. The results demonstrate that our method is also effective on this real-world dataset.

Table 9
Cross-method evaluation: Generalization performance on Celeb-DF (AUROC%) after training on FaceForensics++ dataset.

Method	FF++ [4]	Celeb-DF [5]
Two-stream [50]	70.10	53.80
Meso4 [11]	84.70	54.80
MesoInception4 [11]	83.00	53.60
FWA [42]	80.10	56.90
Xception-raw [5]	99.70	48.20
Xception-c23 [5]	99.70	65.30
Xception-c40 [5]	95.50	65.50
Multi-task [41]	76.30	54.30
Capsule [13]	96.60	57.50
DSP-FWA [42]	93.00	64.60
Two Branch [51]	93.18	73.41
F^3-Net [35]	98.10	65.17
EfficientNet-B4 [30]	99.70	64.29
Multi-Attention [7]	99.80	67.44
Ours	99.82	67.26

4.5. Running speed

To prove that our proposed method is suitable for the practical usage, we test the running speed, including the data preprocessing step (i.e. FSD operation in our method) and the model reasoning step of our proposed method, on our experimental platform (equipped with an i9-10900k CPU and a RTX 3090 GPU) and compare with the baseline model (i.e. Xception [14]) and the F^3-Net [35] (a state-of-the-art). The results are shown in Table 6. Results show that our proposed method can run in a real-time speed so is competent for practical usage. And comparing with other texture focused methods based on frequency-domain features (e.g. FAD and LFS feature used in F^3-Net), our FSD preprocessing is much more rational in time consumption.

4.6. The extension of our method to anti-spoofing

Our high-level semantics reduction method is effective in encouraging the backbone networks to focus on texture information. Thus it can be applied to other computer vision tasks where low-level texture information is also important, like Face Anti-Spoofing (FAS). Similar to the Deepfake detection, FAS also needs detectors to detect fake faces from the videos, yet the fake faces here are not generated by neural networks, but by manual committed presentation attacks like photo prints, video display and 3D masks. For the print attack and display attack, high-level semantics including facial structure is useless like

Deepfake detection, and all the attack clues are hidden in the texture information. Thus our method can be perfectly utilized. We train our proposed model without $\mathcal{L}_{dcl}$ and $\mathcal{L}_{dst}$ on all four protocols of the OULU-NPU [9] dataset. The test results are reported and compared with previous FAS methods in Table 8. The results show our method is competitive with these previous work on FAS, and we even achieve the state-of-the-art results on the challenging protocol 3 and 4, which proves our proposed model can extract consistent texture features between different sessions and cameras.

4.7. Cross-method evaluation

In Table 9, we conduct a cross-method evaluation by training the model on the FaceForensics++ dataset and testing it on the Celeb-DF dataset. This setup assesses the model's ability to generalize across different datasets and methods, effectively addressing the cross-method testing concerns. The results demonstrate the model's robustness in handling variations in data from different sources, validating its applicability in diverse deepfake detection scenarios.

5. Conclusions and future work

5.1. Advantage

In this paper, we propose four mechanisms that improve the robustness to the video compression of existing Deepfake detectors to improve their performance on low-definition videos and multi-definition videos. It means our method can work better on the videos in the wild than the competitors. Extensive experiments demonstrate our high-level semantic reduction method is effective in forcing the backbone networks to focus on the low-level texture information, thus it can be well applied to other similar tasks like Face-Anti Spoofing. We achieve state-of-the-art performance on the most challenging LD *NeuralTextures* subset of the FaceForensics++ [4] dataset and the FaceForensics++ automated benchmark.

5.2. Limitation and future work

Our main focus in this paper is to enhance the Deepfake detection performance on low definition and cross-definition videos, which is great practical significance for the general video forensics on the in-the-wild internet videos. However, the Deepfake detection on these videos faces another critical issue, that is the Deepfake detectors trained on specific types of manipulated videos cannot generalize to other manipulation types well [43]. To quantitatively analyze the generalization capacity of our method, we train a model on the complete FaceForensics++ dataset [4] and test it on the Celeb-DF dataset [5]. As the results shown in Table 9, though our work achieves significant improvement on the generalization capacity compared with the baseline Xception model (67.26% vs. 65.50%), the performance is still not competent for practical application. Incorporating adversarial losses, specifically Adversarial Jigsaw Loss and Domain Classification Loss, introduces additional complexity into the training process of our model. The minimax optimization problem inherent in adversarial training can lead to challenges such as training instability and oscillations, especially if the adversarial components do not converge smoothly. To address these issues, we implemented several strategies: careful balancing of the adversarial losses relative to the primary classification loss, alternating optimization akin to techniques used in GANs, and a gradual warm-up phase to introduce adversarial losses progressively. Despite these measures, the training process remains sensitive to hyperparameter tuning. Our experiments, conducted over approximately 20 h on an RTX 3090 GPU (with a computational cost of 246.24 × 109 FLOPs), indicate that while the adversarial losses add complexity, they significantly enhance the model's robustness and generalization capabilities. Hence our future work will focusing on the Deepfake detection of various manipulation methods, and develop detector which is not only robust on the definition of the videos but also is available on different manipulation methods.

CRediT authorship contribution statement

Cairong Zhao: Supervision, Project administration, Methodology, Investigation. **Chutian Wang:** Writing – review & editing, Writing – original draft, Software, Methodology, Investigation. **Zifan Song:** Writing – review & editing, Software. **Guosheng Hu:** Supervision, Investigation. **Liang Wang:** Supervision, Resources. **Duoqian Miao:** Funding acquisition.

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.

Data availability

Data will be made available on request.

References

[1] D.P. Kingma, M. Welling, Auto-encoding variational Bayes, 2014, arXiv:1312.6114.

[2] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, NIPS '14, MIT Press, Cambridge, MA, USA, 2014, pp. 2672–2680.

[3] 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.

[4] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, M. Nießner, Faceforensics++: Learning to detect manipulated facial images, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 1–11.

[5] 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.

[6] B. Dolhansky, R. Howes, B. Pflaum, N. Baram, C.C. Ferrer, The deepfake detection challenge (dfdc) preview dataset, 2019, arXiv preprint arXiv:1910.08854.

[7] H. Zhao, W. Zhou, D. Chen, T. Wei, W. Zhang, N. Yu, Multi-attentional deepfake detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2021.

[8] J. Amigo, L. Kocarev, J. Szczepanski, Theory and practice of chaotic cryptography, Phys. Lett. A 366 (3) (2007) 211–216.

[9] Z. Boulkenafet, J. Komulainen, L. Li, X. Feng, A. Hadid, Oulu-npu: A mobile face presentation attack database with real-world variations, in: 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), IEEE, 2017, pp. 612–618.

[10] C. Wang, C. Zhao, G. Hu, Multi-definition video deepfake detection via semantics reduction and cross-domain training, in: 2022 IEEE International Conference on Multimedia and Expo (ICME) (To Be Published), IEEE, 2022.

[11] 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, IEEE, 2018, pp. 1–7.

[12] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, M. Nießner, Faceforensics: A large-scale video dataset for forgery detection in human faces, 2018, arXiv preprint arXiv:1803.09179.

[13] 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, IEEE, 2019, pp. 2307–2311.

[14] 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.

[15] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, Ieee, 2009, pp. 248–255.

[16] J. Thies, M. Zollhöfer, M. Nießner, Deferred neural rendering: Image synthesis using neural textures, ACM Trans. Graph. 38 (4) (2019) 1–12.

[17] H. Dang, F. Liu, J. Stehouwer, X. Liu, A.K. Jain, On the detection of digital face manipulation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 5781–5790.

[18] H. Qi, Q. Guo, F. Juefei-Xu, X. Xie, L. Ma, W. Feng, Y. Liu, J. Zhao, DeepRhythm: Exposing DeepFakes with attentional visual heartbeat rhythms, in: Proceedings of the 28th ACM International Conference on Multimedia, 2020, pp. 4318–4327.

[19] 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.
[20] Z. Gu, Spatiotemporal inconsistency learning for DeepFake video detection, in: Proceedings of the 29th ACM International Conference on Multimedia, 2021.
[21] U.A. Ciftci, I. Demir, L. Yin, Fakecatcher: Detection of synthetic portrait videos using biological signals, IEEE Trans. Pattern Anal. Mach. Intell. (2020).
[22] W. Pu, J. Hu, X. Wang, Y. Li, S. Hu, B. Zhu, Q. Song, X. Wu, S. Lyu, Learning a deep dual-level network for robust DeepFake detection, Pattern Recognit. (2022) 108832.
[23] J. Hu, X. Liao, W. Wang, Z. Qin, Detecting compressed deepfake videos in social networks using frame-temporality two-stream convolutional network, IEEE Trans. Circuits Syst. Video Technol. (2021).
[24] 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.
[25] Y. Chen, Y. Bai, W. Zhang, T. Mei, Destruction and construction learning for fine-grained image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 5157–5166.
[26] P. Kumar, M. Vatsa, R. Singh, Detecting face2face facial reenactment in videos, in: The IEEE Winter Conference on Applications of Computer Vision, 2020, pp. 2589–2597.
[27] S. Woo, J. Park, J.-Y. Lee, I.S. Kweon, Cbam: Convolutional block attention module, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 3–19.
[28] I. Misra, L.v.d. Maaten, Self-supervised learning of pretext-invariant representations, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 6707–6717.
[29] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, Stat 1050 (2015) 9.
[30] M. Tan, Q. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, in: International Conference on Machine Learning, PMLR, 2019, pp. 6105–6114.
[31] J. Fridrich, J. Kodovsky, Rich models for steganalysis of digital images, IEEE Trans. Inf. Forensics Secur. 7 (3) (2012) 868–882.
[32] D. Cozzolino, G. Poggi, L. Verdoliva, Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection, in: Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security, 2017, pp. 159–164.
[33] B. Bayar, M.C. Stamm, A deep learning approach to universal image manipulation detection using a new convolutional layer, in: Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security, 2016, pp. 5–10.
[34] N. Rahmouni, V. Nozick, J. Yamagishi, I. Echizen, Distinguishing computer graphics from natural images using convolution neural networks, in: 2017 IEEE Workshop on Information Forensics and Security, WIFS, IEEE, 2017, pp. 1–6.
[35] Y. Qian, G. Yin, L. Sheng, Z. Chen, J. Shao, Thinking in frequency: Face forgery detection by mining frequency-aware clues, in: European Conference on Computer Vision, Springer, 2020, pp. 86–103.
[36] C. Feichtenhofer, H. Fan, J. Malik, K. He, Slowfast networks for video recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 6202–6211.
[37] Y. Xu, J. Liang, G. Jia, Z. Yang, Y. Zhang, R. He, Tall: Thumbnail layout for deepfake video detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 22658–22668.
[38] C. Miao, Z. Tan, Q. Chu, H. Liu, H. Hu, N. Yu, F 2 trans: High-frequency fine-grained transformer for face forgery detection, IEEE Trans. Inf. Forensics Secur. 18 (2023) 1039–1051.
[39] T.S. Gunawan, S.A.M. Hanafiah, M. Kartiwi, N. Ismail, N.F. Za'bah, A.N. Nordin, Development of photo forensics algorithm by detecting photoshop manipulation using error level analysis, Indones. J. Electr. Eng. Comput. Sci. 7 (1) (2017) 131–137.
[40] K. Sun, H. Liu, Q. Ye, J. Liu, Y. Gao, L. Shao, R. Ji, Domain general face forgery detection by learning to weight, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, 2021, pp. 2638–2646.
[41] H.H. Nguyen, F. Fang, J. Yamagishi, I. Echizen, Multi-task learning for detecting and segmenting manipulated facial images and videos, in: 2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems, BTAS, IEEE, 2019, pp. 1–8.
[42] Y. Li, Exposing deepfake videos by detecting face warping artif acts, 2018, arXiv preprint arXiv:1811.00656.
[43] 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.
[44] Z. Boulkenafet, J. Komulainen, Z. Akhtar, A. Benlamoudi, D. Samai, S.E. Bekhouche, A. Ouafi, F. Dornaika, A. Taleb-Ahmed, L. Qin, et al., A competition on generalized software-based face presentation attack detection in mobile scenarios, in: 2017 IEEE International Joint Conference on Biometrics, IJCB, IEEE, 2017, pp. 688–696.
[45] Y. Liu, A. Jourabloo, X. Liu, Learning deep models for face anti-spoofing: Binary or auxiliary supervision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 389–398.
[46] A. Jourabloo, Y. Liu, X. Liu, Face de-spoofing: Anti-spoofing via noise modeling, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 290–306.
[47] Z. Wang, C. Zhao, Y. Qin, Q. Zhou, G. Qi, J. Wan, Z. Lei, Exploiting temporal and depth information for multi-frame face anti-spoofing, 2018, arXiv preprint arXiv:1811.05118.
[48] A. George, S. Marcel, Deep pixel-wise binary supervision for face presentation attack detection, in: 2019 International Conference on Biometrics, ICB, IEEE, 2019, pp. 1–8.
[49] Z. Yu, C. Zhao, Z. Wang, Y. Qin, Z. Su, X. Li, F. Zhou, G. Zhao, Searching central difference convolutional networks for face anti-spoofing, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 5295–5305.
[50] P. Zhou, X. Han, V.I. Morariu, L.S. Davis, Two-stream neural networks for tampered face detection, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPRW, IEEE, 2017, pp. 1831–1839.
[51] 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.
[52] L. Jiang, R. Li, W. Wu, C. Qian, C.C. Loy, Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 2889–2898.

Cairong Zhao is currently a professor at Tongji University. He received a Ph.D. degree from Nanjing University of Science and Technology, an M.Sc. degree from Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, and a B.Sc. degree from Jilin University, in 2011, 2006, and 2003, respectively. He is the author of more than 30 scientific papers in pattern recognition, computer vision, and related areas. His research interests include computer vision, pattern recognition, and visual surveillance.

Chutian Wang is currently a graduate student at Tongji University. He received a B.Sc. degree from Department of Computer Science and Technology, Tongji University, in 2020. His research interests include computer vision and pattern recognition.

Zifan Song is currently pursuing the Ph.D. degree with the College of Electronics and Information Engineering, Tongji University, Shanghai, China. His research interests include deep learning, computer vision, and multimodal learning.

Guosheng Hu is a Senior Researcher of Oosto. He was a postdoctoral researcher in the LEAR team, INRIA Grenoble Rhone-Alpes, France from May 2015 to May 2016. He finished his Ph.D. in Centre for Vision, Speech and Signal Processing, University of Surrey, UK in June, 2015. His research interests include deep learning, pattern recognition, and biometrics (mainly face recognition).

Liang Wang (Fellow, IEEE) received the B.Eng. and M.Eng. degrees from Anhui University in 1997 and 2000, respectively, and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences (CASIA), in 2004. From 2004 to 2010, he was a Research Assistant with Imperial College London, U.K., and Monash University, Australia, also a Research Fellow with the University of Melbourne, Australia, and also a Lecturer with the University of Bath, U.K.. He is currently a Full Professor of the Hundred Talents Program with the National Laboratory of Pattern

Recognition, CASIA. His current research interests include machine learning, pattern recognition, and computer vision. He is an IAPR fellow.

Duoqian Miao was born in 1964. He is a professor and a Ph.D. tutor at the College of Electronics and Information Engineering of Tongji University, and he serves as Vice President of the International Rough Set Society (IRSS), Executive Manager of the Chinese Association for Artificial Intelligence (CAAI), Chair of the CAAI Granular Computing Knowledge Discovery Technical Committee, a distinguished member of Chinese Computer Federation (CCF), Vice President of the Shanghai Computer Federation, and Vice President of the Shanghai Association for Artificial Intelligence. He serves as Associate Editor for the International Journal of Approximate Reasoning and Editor of the Journal of Computer Research and Development (in Chinese). His interests includes machine learning, data mining, big data analysis, granular computing, artificial intelligence, and text image processing. He has published more than 200 papers in IEEE Transactions on Cybernetics, IEEE Transactions on Information Forensics and Security, IEEE Transactions on Knowledge and Data Mining, IEEE Transactions on Fuzzy Systems, Pattern Recognition, Information Sciences, Knowledge-Based Systems, Chinese Journal of Computers, Journal of Software (in Chinese), Journal of Computer Research and Development (in Chinese), Automatica Sinica (in Chinese), and ACTA Electronica Sinica (in Chinese). He won the second prize at Wuwenjun AI Science and Technology (2018).