Forensic Science International: Digital Investigation 50 (2024) 301808

Contents lists available at ScienceDirect

Forensic Science International: Digital Investigation

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

DFRWS APAC 2024 - Selected Papers from the 4th Annual Digital Forensics Research Conference APAC

TAENet: Two-branch Autoencoder Network for Interpretable Deepfake Detection

Fuqiang Du [a,b], Min Yu [a,b,*], Boquan Li [c,**], Kam Pui Chow [d], Jianguo Jiang [a,b], Yixin Zhang [a,b], Yachao Liang [a,b], Min Li [a,b], Weiqing Huang [a,b]

[a] Institute of Information Engineering, Chinese Academic of Sciences, Beijing, China
[b] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[c] College of Computer Science and Technology, Harbin Engineering University, Harbin, China
[d] Department of Computer Science, The University of Hong Kong, Hong Kong

ARTICLE INFO

Keywords:
Deepfake detection
Interpretibility representation
Image forensic
Disentanglement learning

ABSTRACT

Deepfake detection attracts increasingly attention due to serious security issues caused by facial manipulation techniques. Recently, deep learning-based detectors have achieved promising performance. However, these detectors suffer severe untrustworthy due to the lack of interpretability. Thus, it is essential to work on the interpretibility of deepfake detectors to improve the reliability and traceability of digital evidence. In this work, we propose a two-branch autoencoder network named TAENet for interpretable deepfake detection. TAENet is composed of Content Feature Disentanglement (CFD), Content Map Generation (CMG), and Classification. CFD extracts latent features of real and forged content with dual encoder and feature discriminator. CMG employs a Pixel-level Content Map Generation Loss (PCMGL) to guide the dual decoder in visualizing the latent representations of real and forged contents as real-map and fake-map. In classification module, the Auxiliary Classifier (AC) serves as map amplifier to improve the accuracy of real-map image extraction. Finally, the learned model decouples the input image into two maps that have the same size as the input, providing visualized evidence for deepfake detection. Extensive experiments demonstrate that TAENet can offer interpretability in deepfake detection without compromising accuracy.

1. Introduction

Benefiting from the successful application of generative models in the field of computer vision, deepfake technologies, represented by Autoencoders (AE) (Badrinarayanan et al., 2017) and Generative Adversarial Networks (GAN) (Goodfellow et al., 2020), have rapidly developed and garnered widespread attention. Deepfakes are characterized by their low threshold for creation and high realism of forged images, which increases their risk of misuse. The abuse of fake technology can lead to the dissemination of false information, fraud online, privacy violations, and political manipulation. In recent years, many deep learning-based deepfake detection models (Rossler et al., 2019; Chollet, 2017; Afchar et al., 2018; Zhou et al., 2017; Qian et al., 2020; Zhao et al., 2021a) have been proposed, demonstrating significant detection accuracy. However, limited by the black-box nature of deep learning, these models struggle to explain their detection results, i.e., they cannot elucidate why an image is deemed fake or identify which parts is the decision-making region of the image (Wang et al., 2022a). The lack of interpretability in these models implies low trustworthiness, making practical deployment challenging. As a result, developing effective and interpretable deepfake detection algorithm is vitally essential.

Some recent works have focused on this imminent problem and attempted to provide reasonable explanations to improve the interpretability for deepfake detection. These works can be roughly categorized into two branches: 1) Saliency Map-Based methods, which highlight the most important pixels for deepfake detection algorithms (Alqaraawi et al., 2020). Typically, these approaches augment the original detection model with class activation map modules, displaying the regions of interest through heatmaps to show the areas the model focuses on when making decisions. However, the highlighted areas are not necessarily the actual forged regions, and these decision areas can

* Corresponding author. Institute of Information Engineering, Chinese Academic of Sciences, Beijing, China.
** Corresponding author.
E-mail addresses: yumin@iie.ac.cn (M. Yu), liboquan@hrbeu.edu.cn (B. Li).

https://doi.org/10.1016/j.fsidi.2024.301808

Available online 18 October 2024

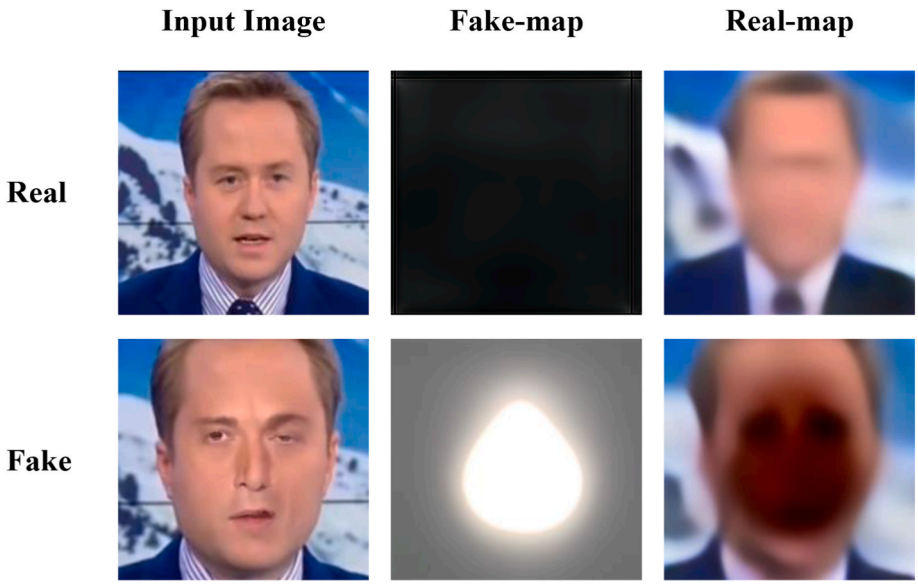

Fig. 1. Image disentanglement. The real content is visualized as real-map. The fake content is visualized as fake-map. The fake-map of real image and fake image is different.

also be highlighted in real facial images, making it impossible to distinguish them from regions in fake faces. Therefore, they cannot be used as evidence of forgery. 2) Forgery Clue-Based methods, which identify artifacts (Li et al., 2022; Hua et al., 2023; Zhao et al., 2021b), splicing traces (Li et al., 2020a), noise (Wang and Chow, 2023), and other features in images as evidence of forgery, providing a certain level of explanation for the model's decisions. However, these methods have limitations. For example, Hua et al. (2023) proposes an interpretability method for deepfake detection, but it mainly explains by visualizing the traces of forgery and cannot explain the forged content. Zhao et al. (2021b) explains forged regions by using the cue of the source feature inconsistency within the forged images. However, when faced with high-quality forged images, both detection performance and interpretability decrease significantly.

Li et al. (2020a) cannot detect or explain forged images that do not involve blending operations. Li et al. (2022) and Wang and Chow (2023) are not capable of explaining high-quality forged images. Although the above explanation methods can, to some extent, indicate the presence of forgery, they cannot pinpoint the specific forged regions. We believe that, while ensuring accuracy, being able to distinguish between forged and non-forged contents and answer the question of where the forgery occurred would provide a better interpretability.

To address the aforementioned issues, we aim to construct a deepfake detection framework that, without sacrificing accuracy, can split an image into forged and non-forged parts, which represent the forgery-related and forgery-irrelevant content respectively, thereby providing interpretability for the detection. Our approach is inspired by disentangled representation learning (Wang et al., 2022b), which can decouple an image and extract the target contents. Specifically, any image can be disentangled into real and forged contents. Particularly, the forged content of a real image is empty, which can be considered as a zero-image. Thus, the forged content of deepfake and real images exhibit significant differences. If these differences can be extracted and visualized, it would allow us to distinguish between real and fake images while simultaneously providing interpretability. Fig. 1 illustrates the differences in forged content between real and deepfakes.

However, separating the real and forged content of a face is not a trivial task. Lacking the labels of the forged regions makes it difficult to extract features of forgery and non-forgery content. Additionally, visualizing the contents to provide interpretability for the detection poses another challenge.

In this paper, the real and forged contents are visualized as real-map and fake-map respectively. These maps have the same size as the input image. Ideally, the real-map of a real image is the image itself, and the fake-map is a zero image. For a forged image, the real-map and fake-map are two non-zero maps that, when combined, represent the original forged image. In order to learn the two maps of an image, we propose a Two-branch Autoencoder Network (TAENet) to decouple real and forgery content features and visualize these features as real-map and fake-map. TAENet is composed of Content Feature Disentanglement (CFD), Content Map Generation (CMG), and Classification (C). (1) CFD learns hidden representations of real and forgery content features from an input image with dual encoder. A discriminator is employed to distinguish these two features, achieving the disentanglement of real and forged content features. (2) CMG is implemented using dual decoder. To obtain the real-map and fake-map, we design a Pixel-level Content Map Generation Loss (PCMGL) to guide the dual decoder in generating accurate real-map and fake-map. (3) Classification is composed of an Auxiliary Classifier (AC) and a Prediction Classifier (C). The difference between real and fake images exists not only in fake maps, but also in real maps. Therefore, we introduce an auxiliary classifier to distinguish real latent features extracted from real and fake images, which can further improve the accuracy of detection and map estimation. Therefore, TAENet maintains high accuracy while predicting forged and real contents, providing visual and interpretable evidence for deepfake detection.

Our contributions can be summarized as follows.

- We propose a novel interpretable deepfake detection framework named Two-branch Autoencoder Network (TAENet), which can disentangle the real and forged contents from an input image, to gain better results for deepfake detection and provide convincing evidence through visualizing real and fake contents.
- The proposed Pixel-level Content Map Generation Loss (PCMGL) is designed for efficient pixel-level supervised training to obtain accurate estimates of fake and real contents.
- Extensive experiments demonstrate the effectiveness of our approach in interpreting face forgery detection with accuracy guarantee.

The remainder of this paper is organized as follows: In section 2, we provide a brief review of related works. Section 3 presents the details of the proposed Two-branch Autoencoder Network (TAENet) framework.

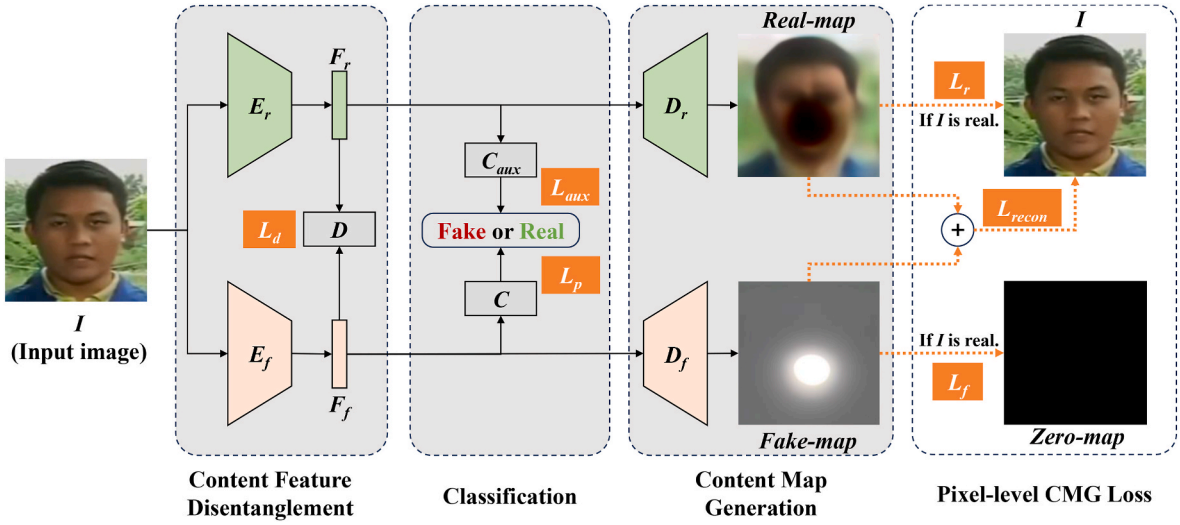

Fig. 2. The overview framework of our proposed method. The framework consists of Content Feature Disentanglement (CFD), Content Map Generation (CMG) and Classification. The CFD extracts latent features of real and forged content with dual encoder and feature discriminator. In the CMG, the dual decoder generate interpretable real-map and fake-map for deepfake detection. A Pixel-level Content Map Generation Loss (PCMGL) is designed to facilitate the learning of CMG. Finally, the prediction of the model is created by Classifier (C).

To assess the effectiveness of our approach, we perform experiments in section 4. Finally, section 5 conclude this work.

2. Related work

2.1. Deepfake detection

Deepfake detection has become a crucial topic of research due to the increasing prevalence and sophistication of deepfake technologies. To address this problem, various methods have been researched to identify manipulated faces. Early work focused on extracting handcrafted features (Bai et al., 2023), such as blinking (Jung et al., 2020), head inconsistencies (Yang et al., 2019), and visual artifacts (Bappy et al., 2019; Li and Lyu, 2018). As deepfake technology has advanced, these features have become increasingly difficult to detect. Researchers have applied deep learning techniques to deepfake detection tasks, achieving notable results. These methods are mainly divided into detection methods based on spatial artifacts (Afchar et al., 2018; Zhao et al., 2021a) and those based on frequency domain artifacts (Qian et al., 2020; Frank et al., 2020; Li et al., 2021). However, these methods fail when encountering high-quality forged images. More importantly, deep learning has a black-box nature, making it impossible to know how the model makes decisions, leading to a lack of interpretability in deepfake detection models and making them difficult to apply in practice. This paper proposes an interpretable deepfake detection method to address this shortcoming.

2.2. Model interpretability

The interpretability of the model aims to describe the internal working mechanism of deep neural networks in understandable terms to humans (Hua et al., 2023). To achieve this goal, numerous works have been proposed (Cheng et al., 2020; Zhang et al., 2019, 2020). However, these works cannot be directly applied to the deepfake detection task (Hua et al., 2023). The interpretability of deepfake detection is challenging (Wang et al., 2022a). The main aspects of interpretability in deepfake detection involve answering why an image is judged as fake and identifying which parts of the image. Only by understanding the decision mechanism of deepfake detection can we deploy these models effectively. Therefore, the interpretability of deepfake detection is crucial. Some methods have addressed this issue, such as Saliency Map-based methods (Alqaraawi et al., 2020; Hua et al., 2023) and Forgery Clue-Based methods (Li et al., 2020a, 2022; Wang and Chow, 2023). However, saliency maps only indicate the regions the model focuses on, which may not necessarily relate to the forgery. Forgery Clue-Based methods rely on image artifacts or splicing traces, which are challenging to identify in high-quality forged images. Thus, this paper proposes a method to decouple the real and fake content in images and visualize these content to achieve interpretable deepfake detection.

3. The proposed approach

In this section, we present our approach of Two-branch Autoencoder Network for Interpretable Deepfake Detection. Here, we first give a brief overview of the problem formulation and then provide a detailed description of the approach.

3.1. Problem formulation

We define an image as composed of real content and fake content, formalized as:

$$I = I_r + I_f \tag{1}$$

where I is an image, I_r is real content of the image, and I_f is fake content of the image. Specifically, the fake content of an real image is considered as empty, denoted as a zero-image. Additionally, we define implicit representation of the image into real features and fake features. Real features are the implicit representation of real content, while fake features are the implicit representation of fake content. For a fogery image, real features are related to the parts that are irrelevant to the forgery, such as the image background. We define the image domain as $I \subseteq X^{n\times n\times 3}$, where n represents the image size. Our goal is to learn an interpretable deepfake detection model $F(I; w)$ that, while maintaining high accuracy, decouples the real part I_r and the fake part I_f of the image. This is formalized as:

$$F(I, w) = M_r, M_f, \widehat{y}, \quad I, M_r, M_f \subseteq X \tag{2}$$

where w is the model parameters, M_r is the visualized image of I_r (real-map), M_f is the visualized image of I_f (fake-map), and $\widehat{y}$ represents the predicted result of the input image.

To achieve the above objectives, we propose a two-branch

autoencoder network (TAENet). TAENet consists of Content Feature Disentanglement (CFD), Content Map Generation (CMG), and Classification (C). The learning process of the model is shown in Fig. 2. CFD extracts real content features and fake content features. CMG visual results of real content and fake content, offering interpretable detection. The prediction result of the image is given by Classification.

3.2. Content Feature Disentanglement

The purpose of Content Feature Disentanglement (CFD) is to extract and separate real and fake features of the image, solving the problem of highly coupled features between real and fake content. As shown in Fig. 2, CFD consists of dual-branch encoder and feature discriminator. For an input image, the dual-branch encoder extracts real content features and fake content features, respectively. Then, the feature discriminator is used to distinguish between these two features.

Dual Encoder. The dual-branch encoder E_r and E_f employ the same backbone based on CNN, and is used to extract the real content features F_r and the fake content features F_f of an input image, respectively. Thus, F_r represents the implicit representation of real content, while F_f represents the implicit representation of fake content. This can be formalized as:

$$F_r = E_r(I) \tag{3}$$

$$F_f = E_f(I) \tag{4}$$

where I is the input image.

Feature Discriminator. After the dual encoder, we obtain real content features F_r and fake content features F_f. Since the real content features and fake content features are different, we label all the features extracted from the E_r encoder with the label "1", and label the features extracted from the E_f encoder with the label "0". In order to supervise these two types of features, we introduce a feature discriminator D to determine the classes of F_r and F_f. The discriminator is a fully connected classifier. We use binary cross-entropy loss L_d to calculate the cross-entropy, which encourages the model to decouple the real content features and fake content features. L_d is formalized as:

$$L_d = L_{ce}^{r}(D_r(F_r), 1) + L_{ce}^{f}(D_f(F_f), 0) \tag{5}$$

where L_{ce}^{r} represents the cross-entropy loss of real content features, and L_{ce}^{f} represents the cross-entropy loss of fake content features.

3.3. Content Map Generation

We utilize the decoupled real content features and forged content features from the input image to generate a real content map and a forged content map. This is a key aspect of interpretability in our approach. In this module, we consider employing dual decoder to generate maps.

Dual Decoder. The dual decoder consist of two same decoders, namely the Real Content Map Decoder D_r and the Fake Content Map Decoder D_f. The input of dual decoder is the real content features F_r and the fake content features F_f obtained from the previous module. Through upsampling and convolutional layers, D_r and D_f respectively generate the real content map (real-map) and the fake content map (fake-map), with the same size as the input image. It can be formalized as:

$$M_r = D_r(F_r) \tag{6}$$

$$M_f = D_f(F_f) \tag{7}$$

where M_r is real-map and M_f is fake-map.

To generate interpretable fake-map that reflect the differences between real and forged images, we designed a Pixel-level Content Map Generation Loss (PCMGL) to facilitate the generation of interpretable real and forged content maps by the dual decoder. According to Equation (1), we divide an input image into real content and forged content. Through our model, the real content is visualized as real-map, and the forged content is visualized as fake-map. For an real image, we know clearly that the real-map should be the image itself, while the fake-map should be a zero map. For a forged image, since both the real and forged contents are unknown, constraints cannot be set for real-map and fake-map. Additionally, for any image, whether real or forged, the sum of real-map and fake-map should be the image itself. Therefore, the PCMGL is formalized as:

$$L_g = L_f + L_r + L_{recon} \tag{8}$$

Here, L_f and L_r are pixel-level L1 loss functions, which constrain the real-map and fake-map of an real image, formalized as:

$$L_f = \|M_r - I_{real}\|_1 \tag{9}$$

$$L_r = \|M_f\|_1 \tag{10}$$

where I_{real} is an input real image, M_r is real-map of the real image, and M_f is fake-map of the real image. L_{recon} is an L2 loss function, which constrains the fake-map and real-map for any image, formalized as:

$$L_{recon} = \|M_r + M_f - I\|_2 \tag{11}$$

where I is the input image, and M_r and M_f are the real-map and fake-map of the image, respectively.

3.4. Classification

Previous work has shown that it is possible to decouple the real content and fake content for an input image. However, we also need to consider how to predict the authenticity of the input image. Considering that there are significant differences in fake-maps between real and forged image, the corresponding forged content features have characteristics that can distinguish between real and forged images. Therefore, we introduce a classifier to distinguish these differences and detect the authenticity of images. Similarly, the real content features of real and forged images are also different. Leveraging this characteristic, we introduce an auxiliary classifier.

Auxiliary Classifier. The auxiliary classifier (C_{aux}) is a binary classifier implemented by fully connected layers, aiming to improve the accuracy of predicting feature maps during the training phase. The input is the real content features, and the output is the prediction result for the input image. We use binary cross-entropy loss in the training phase, as follows:

$$L_{aux} = L_{ce}(C_{aux}(F_r), y) \tag{12}$$

where F_r is real content features of an input image, y is ground truth, and L_{ce} is the cross-entropy loss function.

Prediction Classifier. As mentioned earlier, we draw inspiration from the differences in forged content between real and fake images as a basis for our model to judge the authenticity of images. We use the same fully connected layers structure as the auxiliary classifier to build a prediction classifier, which serves as the final prediction of the model for the authenticity of the input image. The loss function for this classifier is:

$$L_p = L_{ce}(C(F_f), y) \tag{13}$$

where F_f is forged content features of an input image, y is the ground truth, and L_{ce} is the cross-entropy loss function.

3.5. Total loss

The final loss function of the training phase is the weighted sum of the above loss functions.

$$L = \lambda_1 L_d + \lambda_2 L_g + \lambda_3 L_{aux} + \lambda_4 L_p$$

Table 1
Detection results (%) on DeepFakes, FaceSwap, Face2Face, and NeuralTextures.

Train Set	Method	DeepFakes		FaceSwap		Face2Face		NeuralTextures	
		ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
DeepFakes	ResNet18	**95.85**	**98.49**	48.64	36.02	54.12	67.54	54.82	70.97
	Ours	95.63	98.41	48.87	39.55	53.61	69.27	53.58	69.23
FaceSwap	ResNet18	50.75	50.85	95.00	98.71	51.39	61.30	49.65	51.51
	Ours	50.70	52.79	**95.99**	**98.86**	51.31	62.69	50.19	53.74
Face2Face	ResNet18	54.43	71.22	50.73	52.31	**96.01**	98.29	51.63	63.47
	Ours	54.92	73.06	50.82	46.78	95.79	**98.40**	51.11	60.98
NeuralTextures	ResNet18	63.26	77.59	50.25	51.42	55.94	67.94	**89.45**	**95.90**
	Ours	61.49	75.21	50.27	50.25	55.58	66.89	89.38	95.64

Here, λ_1, λ_2, λ_3, λ_4 are hyperparameters that balance training losses. Empirically, we set $\lambda_1 = 1$, $\lambda_2 = 1$, $\lambda_3 = 1$, $\lambda_4 = 1$ during experiments.

4. Experiments

4.1. Experimental settings

Datasets. To evaluate the effectiveness of our proposed method, we conducted experiments on three large-scale mainstream benchmark datasets: FaceForensics++ (FF++) (Rossler et al., 2019), Celeb-DF-v2 (Celeb-DF) (Li et al., 2020b), and DeepFake Detection Challenge (DFDC) (Dolhansky et al., 2019). FF++ comprises 1000 real videos and 4000 forged videos. The forged videos are generated using four different methods: DeepFakes (DF) (Korshunov and Marcel, 2018), FaceSwap (FS) (Rossler et al., 2019), Face2Face (F2F) (Thies et al., 2016), and NeuralTextures (NT) (Thies et al., 2019). Each method corresponds to 1000 fake videos. We used the HQ version of the C23 compression from FF++, and extracted 30 frames from each video. The training, validation and testing sets are divided according to the official guidelines. Celeb-DF consists of 590 real videos, with 390 used for training, 115 for validation, and 115 for testing. There are 5639 forged videos. Each real video randomly sampled for 5 frames, and each forged video randomly sampled for 50 frames to balance the dataset labels. DFDC contains nearly 119,146 videos, with 19,154 real and 99,992 forged videos, which is divided into training, validation, and testing sets in 6:2:2. To balance the labels, each real video is randomly sampled for 5 frames, and each forged video is randomly sampled for 1 frame. These datasets provide a diverse range of real and forged videos, allowing us to thoroughly evaluate the performance of our method across different scenarios and challenges in deepfake detection.

Experimental Details. We use ResNet18 as the backbone (He et al., 2016). The backbone was trained on ImageNet. Face extraction and alignment are performed using DLIB (Sagonas et al., 2016). The aligned faces are resized to 224 × 224 for both training and testing. We use the Adam (Kingma and Ba, 2014) for optimization with the learning rate of 0.001, and the batch size is 128.

Evaluation Metrics. To evaluate the effectiveness of our approach, we check both the detection performance and the interpretability performance with comprehensive metrics. We use area under curve (AUC) and accuracy (ACC) as detection evaluation metrics, which is consistent with the evaluation approach adopted in previous works (Cao et al., 2022; Liu et al., 2021). The interpretability evaluation is implemented through the visualization analysis of maps.

4.2. Detection performance

Main Objective Accuracy. We proposed a framework (TAENet), where different backbone networks of base models can serve as encoders within this framework. Consequently, we initially evaluated the accuracy of the baseline model (ResNet18) and its corresponding TAENet. To achieve this goal, all models are trained on DeepFakes, FaceSwap, Face2Face and NeuralTextures with pretrained ResNet18 on ImageNet. The results are presented in Table 1. We can see that our interpretable models are comparable to the baseline models on ACC and AUC. It can be observed that our proposed TAENet maintains high accuracy compared to the baseline models. This indicates that our model does not compromise the accuracy of the original baseline models and can provide interpretability.

Table 2
Comparison of accuracy (%) with competing methods on FF++ and Celeb-DF.

Method	FF++		Celeb-DF	
	AUC	ACC	AUC	ACC
Meso4 (Afchar et al., 2018)	82.32	72.12	91.24	83.67
MesoInception4 (Afchar et al., 2018)	86.45	77.30	92.02	84.53
Xception (Rossler et al., 2019)	91.80	82.54	96.20	90.23
SPSL (Liu et al., 2021)	**96.25**	**89.53**	**98.26**	**93.24**
Ours	95.35	87.20	97.63	91.94

Comparison of Accuracy with Competing Methods. To further assess the comprehensive detection capabilities of our framework, we reproduced four state-of-the-art methods, including Meso4 (Afchar et al., 2018), MesoInception4 (Afchar et al., 2018), Xception (Rossler et al., 2019), and SPSL (Liu et al., 2021), under the same conditions. We trained these models on FF++, Celeb-DF and tested in-dataset, evaluating by AUC and ACC. The experimental results are presented in Table 2. Specifically, our method is significantly better than Meso4, MesoInception4, and Xception, achieving the second highest AUC. Compared to the SOTA method (Liu et al., 2021), our method is competitive, showcasing superior performance in deepfake detection. It is evident that the proposed TAENet leads to excellent performance compared to other models in most cases.

4.3. Interpretability performance

Accuracy and interpretability are two crucial capabilities of our proposed method. The above experiments demonstrate that our method ensures accuracy. Next, we will evaluate the interpretability by visualizing the real content and the fake content.

Visualization of Maps. We analyzed the interpretability of our model on the Deepfakes, FaceSwap, Face2Face and NeuralTextures datasets. We visualized the real content and forged content of input images as real-maps and fake-maps. As shown in Fig. 3, in the real-map, the non-black regions represent the real content of the image. In the fake-map, the non-black regions represent the forged content of the image. From Fig. 3, it can be observed that the real map and the fake map have a complementary relationship. The forged content of the input image is decoupled into the fake-map and visualized, while the real content is decoupled into the real-map and visualized. For real images, the fake map is a zero-map, and the real map is similar to the original input image, which is consistent with reality. This indicates that the model decouples the real and forged content from real images. To verify

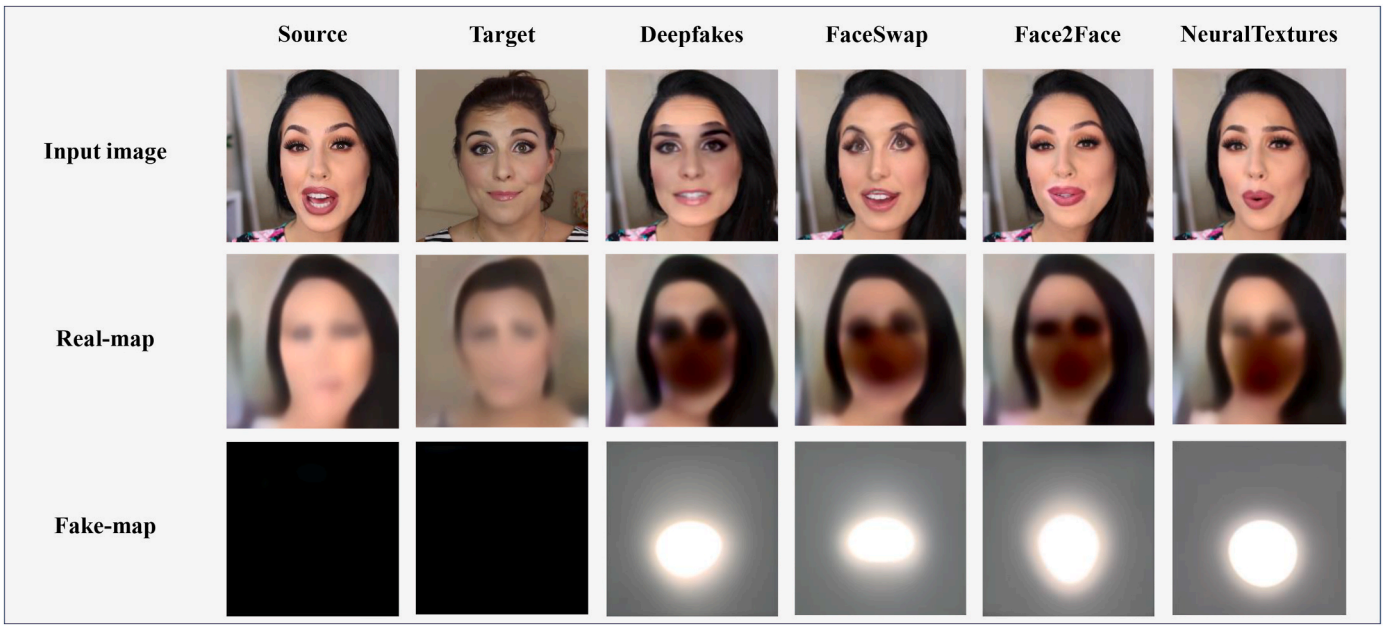

Fig. 3. The real content and forged content of input images are visualized as real-maps and fake-maps. The non-black regions represent forged content in real-map. The non-black regions represent forged content in fake-map. These maps provide interpretability for deepfake detection.

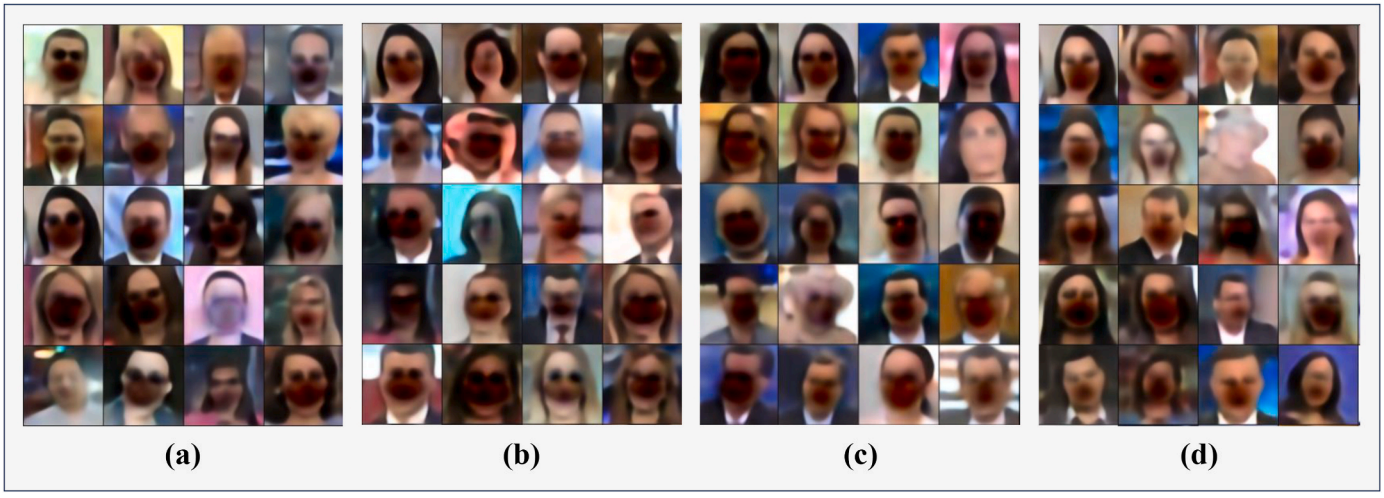

Fig. 4. Fake-maps of fake images from four forgery methods. (a) Deepfakes, (b) FaceSwap, (c) Face2Face, (d) NeuralTextures. Black regions represent forged content. The forged contents we detected matched what was actually known.

the accuracy of the decoupling capability in fake images, we performed a qualitative visual analysis.

We know that Face2Face is a facial reenactment technique that targets the entire face, so its forged content includes nearly the whole face. NeuralTextures is a facial reenactment technique that targets the mouth area, so its forged content is the mouth. FaceSwap is a face-swapping technique, and its forged content covers most of the facial area. Deepfakes is also a face-swapping technique, with forged content covering most of the facial area, typically in a rectangular region from the eyebrows to the chin.

From Fig. 4, it can be seen that for images forged using the Face2Face method, the extracted forged content is concentrated in the facial area, while the real content includes non-facial regions such as hair and background. For images forged using the NeuralTextures method, the extracted forged content is located in the mouth area, with the area outside the mouth being real content, which aligns closely with the actual forged content. For images forged using the FaceSwap method, the extracted forged content covers most of the facial area, the remaining areas being real content. For images forged using the Deepfakes method, the extracted forged content is mainly concentrated in the region between the eyebrows and the chin, closely matching the actual forged content.

Therefore, for the input images, the real and forged content extracted by our model generally corresponds to the actual content.

Explanations. From the above analysis, it is evident that the model accurately decouples the forged and real content of input images. The forged content map and real content map are generated from the real content features and forged content features, indicating that the dual encoder in the model successfully extract these features. Our model makes authenticity predictions based on the features of the forged content, providing interpretability. Furthermore, in addition to the significant differences in the fake content maps of real and fake images, the real content maps (real-map) also exhibit notable differences. This further provides evidence for deepfake detection. This demonstrates that

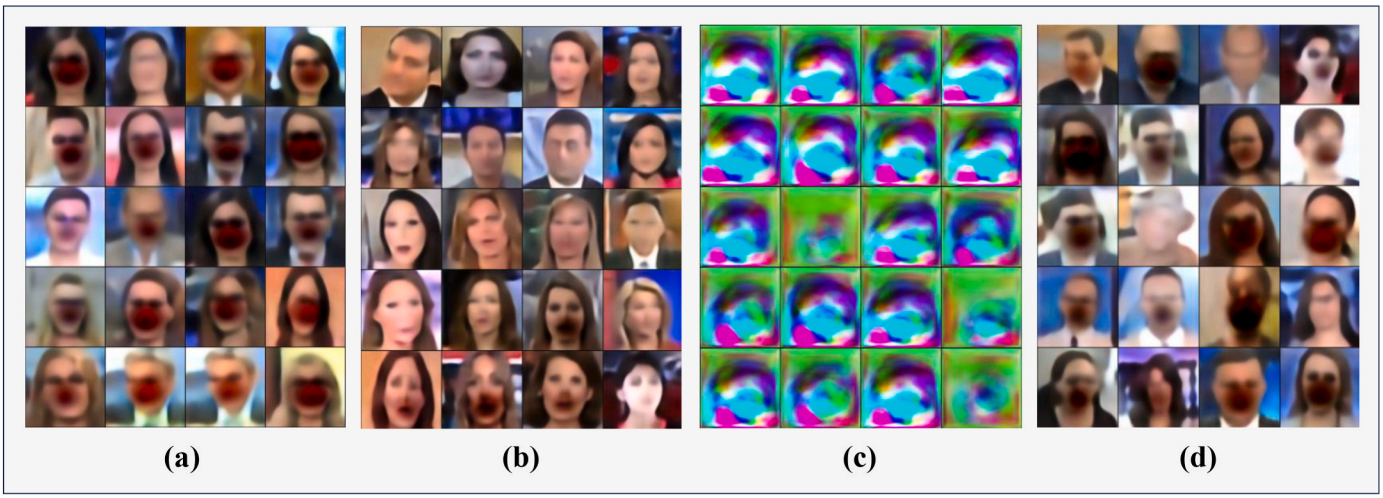

Fig. 5. Ablation study on the NeuralTextures dataset, examining the impact of removing different components from the proposed method on interpretability. (a) without D, (b) without AC, (c) without PCMGL, (d) is our proposed method. The mouth is manipulated by NeuralTextures in face image.

Table 3
Cross-dataset evaluation accuracy (%) on FF++, Celeb-DF, and DFDC.

Train Set	FF++		Celeb-DF		DFDC		Avg.	
	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC
FF++	95.35	87.20	81.00	62.36	68.48	61.67	81.61	74.89
Celeb-DF	58.67	54.67	97.63	91.94	62.81	57.67	75.18	69.79
DFDC	60.22	57.48	77.62	65.95	85.59	76.12	76.04	68.85

our model is an easily interpretable deepfake detection method. The generated real-map and fake-map visually assist in identifying forged images, offering discernible evidence for deepfake detection.

4.4. Ablation study

We conducted ablation experiments on the NeuralTextures to evaluate the impact of the real-fake feature Discriminator (D), the Auxiliary Classifier (AC), and Pixel-level Content Map Generation Loss (PCMGL) on the detection capability and interpretability.

Effects of Detection Performance. We first compared the impact of different components on detection performance. The experimental results are presented in Table 4. The highest AUC (95.70) is observed when the PCMGL component is removed. However, the differences are minor and the AUC of all methods is very close, indicating that the impact on the accuracy of each component is relatively small. It is indicate that the accuracy of our method is determined by the performance of backbone.

Effects of Interpretability Performance. Then, we evaluated the impact of each component on the interpretive performance. Fig. 5 illustrates the real content maps generated by the models which removing some component. Visually, after removing the Discriminator (D), the forgery area is too large, indicating that the model does not completely separate the real and fake contents. When the Auxiliary Classifier (AC) is removed, the forgery regions are not obvious. The real content represented by real-map showed deviations. Specifically, when our proposed Pixel-level Content Map Generation Loss was removed, we cannot understand the contents in real-map, which shows the interpretability significantly decreased.

Based on the above experiments and analysis, the auxiliary classifier, real-fake feature discriminator, and Pixel-level Content Map Generation Loss ensure the interpretive capabilities. This demonstrates the effectiveness of these components.

Table 4
The ablation study of proposed method on NeuralTextures.

Method	NeuralTextures	
	AUC	ACC
Proposed w/o D	95.66	89.33
Proposed w/o AC	95.60	88.41
Proposed w/o PMGL	**95.70**	**89.60**
Proposed (full method)	95.64	89.38

4.5. Discussion

The above experiments prove that the proposed method can provide interpretability for deepfake detection while ensuring accuracy. Meanwhile, the generalization of the model should also be appreciated. In order to evaluate the generalization, we conducted cross-dataset experiments. The results are shown in Table 3. It can be seen that our model has a certain generalization, with an average AUC of more than 75%, which is not enough for deepfake detection. In the future, we will improve the generalization of the model on this interpretable method.

5. Conclusion

In this paper, we propose an interpretable deepfake detection network based on a two-branch autoencoder. The real and forged content features of an image are decoupled by dual encoder. Then, guided by a Pixel-level Content Map Generation Loss, dual decoder generate fake content map and real content map of the same size as the input image. To evaluate the effectiveness of our proposed method, we conduct extensive experiments on several benchmark datasets and compare to existing SOTA methods. Experiments demonstrate that our approach can provide visually discriminative evidence to understand face forgery detection while maintaining high accuracy. In the future, we will further investigate the generalization of deepfake detection

models while maintaining interpretability.

Acknowledgment

This work was supported by the National Key R&D Program of China (Grant No. 2021YFF0602104).

References

Afchar, D., Nozick, V., Yamagishi, J., Echizen, I., 2018. Mesonet: a compact facial video forgery detection network. In: 2018 IEEE International Workshop on Information Forensics and Security (WIFS). IEEE, pp. 1–7.

Alqaraawi, A., Schuessler, M., Weiß, P., Costanza, E., Berthouze, N., 2020. Evaluating saliency map explanations for convolutional neural networks: a user study. In: Proceedings of the 25th International Conference on Intelligent User Interfaces, pp. 275–285.

Badrinarayanan, V., Kendall, A., Cipolla, R., 2017. Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39, 2481–2495.

Bai, W., Liu, Y., Zhang, Z., Li, B., Hu, W., 2023. Aunet: learning relations between action units for face forgery detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 24709–24719.

Bappy, J.H., Simons, C., Nataraj, L., Manjunath, B., Roy-Chowdhury, A.K., 2019. Hybrid lstm and encoder–decoder architecture for detection of image forgeries. IEEE Trans. Image Process. 28, 3286–3300.

Cao, J., Ma, C., Yao, T., Chen, S., Ding, S., Yang, X., 2022. End-to-end reconstruction-classification learning for face forgery detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4113–4122.

Cheng, K., Wang, N., Li, M., 2020. Interpretability of deep learning: a survey. In: The International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery. Springer, pp. 475–486.

Chollet, F., 2017. Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1251–1258.

Dolhansky, B., Howes, R., Pflaum, B., Baram, N., Ferrer, C.C., 2019. The Deepfake Detection Challenge (Dfdc) Preview Dataset arXiv preprint arXiv:1910.08854.

Frank, J., Eisenhofer, T., Schönherr, L., Fischer, A., Kolossa, D., Holz, T., 2020. Leveraging frequency analysis for deep fake image recognition. In: International Conference on Machine Learning. PMLR, pp. 3247–3258.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2020. Generative adversarial networks. Commun. ACM 63, 139–144.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.

Hua, Y., Shi, R., Wang, P., Ge, S., 2023. Learning patch-channel correspondence for interpretable face forgery detection. IEEE Trans. Image Process. 32, 1668–1680.

Jung, T., Kim, S., Kim, K., 2020. Deepvision: deepfakes detection using human eye blinking pattern. IEEE Access 8, 83144–83154.

Kingma, D.P., Ba, J., 2014. Adam: A Method for Stochastic Optimization arXiv preprint arXiv:1412.6980.

Korshunov, P., Marcel, S., 2018. Deepfakes: a New Threat to Face Recognition? Assessment and Detection arXiv preprint arXiv:1812.08685.

Li, J., Xie, H., Li, J., Wang, Z., Zhang, Y., 2021. Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6458–6467.

Li, L., Bao, J., Zhang, T., Yang, H., Chen, D., Wen, F., Guo, B., 2020a. Face x-ray for more general face forgery detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5001–5010.

Li, X., Ni, R., Yang, P., Fu, Z., Zhao, Y., 2022. Artifacts-disentangled adversarial learning for deepfake detection. IEEE Trans. Circ. Syst. Video Technol. 33, 1658–1670.

Li, Y., Lyu, S., 2018. Exposing Deepfake Videos by Detecting Face Warping Artifacts arXiv preprint arXiv:1811.00656.

Li, Y., Yang, X., Sun, P., Qi, H., Lyu, S., 2020b. Celeb-df: a large-scale challenging dataset for deepfake forensics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3207–3216.

Liu, H., Li, X., Zhou, W., Chen, Y., He, Y., Xue, H., Zhang, W., Yu, N., 2021. Spatial-phase shallow learning: rethinking face forgery detection in frequency domain. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 772–781.

Qian, Y., Yin, G., Sheng, L., Chen, Z., Shao, J., 2020. Thinking in frequency: face forgery detection by mining frequency-aware clues. In: European Conference on Computer Vision. Springer, pp. 86–103.

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

Sagonas, C., Antonakos, E., Tzimiropoulos, G., Zafeiriou, S., Pantic, M., 2016. 300 faces in-the-wild challenge: Database and results. Image Vis Comput. 47, 3–18.

Thies, J., Zollhöfer, M., Nießner, M., 2019. Deferred neural rendering: image synthesis using neural textures. ACM Trans. Graph. 38, 1–12.

Thies, J., Zollhofer, M., Stamminger, M., Theobalt, C., Nießner, M., 2016. Face2face: real-time face capture and reenactment of rgb videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2387–2395.

Wang, T., Chow, K.P., 2023. Noise based deepfake detection via multi-head relative-interaction. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 14548–14556.

Wang, T., Liao, X., Chow, K.P., Lin, X., Wang, Y., 2022a. Deepfake Detection: A Comprehensive Study from the Reliability Perspective arXiv preprint arXiv:2211.10881.

Wang, Y.C., Wang, C.Y., Lai, S.H., 2022b. Disentangled representation with dual-stage feature learning for face anti-spoofing. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1955–1964.

Yang, X., Li, Y., Lyu, S., 2019. Exposing deep fakes using inconsistent head poses. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 8261–8265.

Zhang, C., Liu, A., Liu, X., Xu, Y., Yu, H., Ma, Y., Li, T., 2020. Interpreting and improving adversarial robustness of deep neural networks with neuron sensitivity. IEEE Trans. Image Process. 30, 1291–1304.

Zhang, Q., Yang, Y., Ma, H., Wu, Y.N., 2019. Interpreting cnns via decision trees. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6261–6270.

Zhao, H., Zhou, W., Chen, D., Wei, T., Zhang, W., Yu, N., 2021a. Multi-attentional deepfake detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2185–2194.

Zhao, T., Xu, X., Xu, M., Ding, H., Xiong, Y., Xia, W., 2021b. Learning self-consistency for deepfake detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15023–15033.

Zhou, P., Han, X., Morariu, V.I., Davis, L.S., 2017. Two-stream neural networks for tampered face detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, pp. 1831–1839.