LAA-Net: Localized Artifact Attention Network for Quality-Agnostic and Generalizable Deepfake Detection

Dat NGUYEN*, Nesryne MEJRI*, Inder Pal SINGH*, Polina KULESHOVA*
Marcella ASTRID*, Anis KACEM*, Enjie GHORBEL*,×, Djamila AOUADA*
CVI², SnT, University of Luxembourg*
Cristal Laboratory, National School of Computer Sciences, University of Manouba×
{dat.nguyen,nesryne.mejri,inder.singh,polina.kuleshova,
marcella.astrid,anis.kacem,enjie.ghorbel,djamila.aouada}@uni.lu

Abstract

This paper introduces a novel approach for high-quality deepfake detection called Localized Artifact Attention Network (LAA-Net). Existing methods for high-quality deepfake detection are mainly based on a supervised binary classifier coupled with an implicit attention mechanism. As a result, they do not generalize well to unseen manipulations. To handle this issue, two main contributions are made. First, an explicit attention mechanism within a multi-task learning framework is proposed. By combining heatmap-based and self-consistency attention strategies, LAA-Net is forced to focus on a few small artifact-prone vulnerable regions. Second, an Enhanced Feature Pyramid Network (E-FPN) is proposed as a simple and effective mechanism for spreading discriminative low-level features into the final feature output, with the advantage of limiting redundancy. Experiments performed on several benchmarks show the superiority of our approach in terms of Area Under the Curve (AUC) and Average Precision (AP). The code is available at https://github.com/10Ring/LAA-Net.

1. Introduction

Thanks to the development of generative models, tremendous advances in deepfake creation have been witnessed. Unfortunately, these fake visual data can be employed for malicious purposes, as shown in [4, 48]. The fact that deepfake generation techniques are rapidly gaining in realism only exacerbates this issue. *It is, therefore, crucial to design methods capable of automatically detecting deepfakes, including the most realistic ones that are commonly referred to as high-quality deepfakes*. Nonetheless, detecting high-quality deepfakes remains extremely challenging as they usually enclose subtle and localized artifacts.

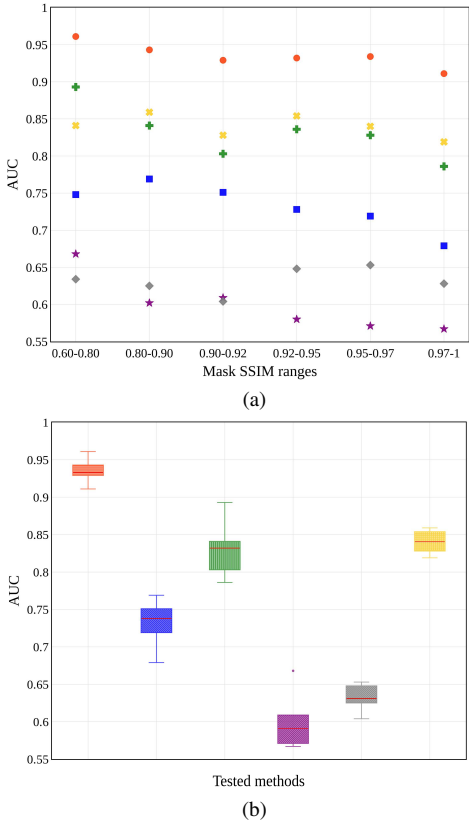


Figure 1. Comparison of LAA-Net (●) with respect to existing methods, namely, Multi-attentional (●) [55], SBI (●) [41], Xception (●) [37], RECCE (●) [6], CADDM (●) [14], using (a) the AUC performance with respect to different ranges of Mask SSIM, and (b) its associated boxplots. *The results were obtained using the official source codes pretrained on FF+ [37] and testing on Celeb-DFv2 [25]. Figure best viewed in colors.

Recent works have mostly focused on improving the generalization capabilities of deepfake detection methods by adopting multi-task learning [7, 24, 54] and/or heuris-

tic fake data generation [24, 41] strategies. However, most of these methods fail to model localized artifacts, which are critical for detecting high-quality deepfakes. This could be explained by the fact that Vanilla Deep Learning (DL) architectures are mainly used. These common architectures, such as XceptionNet [9] and EfficientNet [44], tend to learn global features, ignoring more localized cues [50, 55]. With the use of successive convolutions, localized features across layers gradually fade. Hence, proposing suitable mechanisms for capturing local and subtle artifacts turns out to be necessary.

To the best of our knowledge, only a few research works have explored this research direction [50, 55]. They mainly introduce attention modules that implicitly model subtle inconsistencies through low-level representations [50, 55]. Nevertheless, they still rely on single binary classifiers trained with real/deepfake images without considering any additional strategy for avoiding generalization issues. This considerably restricts the practical usefulness of these methods.

Hence, our goal is to address the detection of high-quality deepfakes and, at the same time, improve the generalization performance. We argue that this can be achieved by designing an attention module compatible with generic deepfake detection strategies. In particular, the solution would be to introduce an explicit fine-grained mechanism within a multi-task learning framework supported by an appropriate pseudo-fake synthesis technique. Moreover, in addition to such a learning strategy, we posit that an adequate architecture preserving low-level features could implicitly contribute to better capturing localized artifacts.

More concretely, this paper proposes a novel fine-grained approach called *Localized Artifact Attention Network (LAA-Net)* that relies on a multi-task learning framework. First, a new fine-grained mechanism that aims at focusing on small regions centered at the vulnerable pixels is introduced. By vulnerable pixels, we mean the pixels that are more likely to showcase a blending artifact[1]. This is achieved by considering two auxiliary branches, namely, *a heatmap branch* and a *self-consistency branch*. On the one hand, the heatmap branch allows localizing the set of vulnerable pixels while taking into account their neighborhood. On the other hand, the self-consistency branch estimates the similarity of pixels with respect to a randomly selected vulnerable point. To simulate fake data and generate ground-truth heatmaps and self-consistency matrices that are predicted by the additional branches, blending-based data synthesis such as [24, 41] are leveraged. Second, the proposed architecture incorporates a novel, simple, yet effective Feature Pyramid Network (FPN) [27] termed *Enhanced FPN* (E-FPN). It enables making use of multi-scale features while avoiding redundancy. In fact, it has been shown that reducing feature redundancy contributes to the regularization of Deep Neural Networks (DNNs) [2]. While the proposed attention mechanism guided by the vulnerable points helps the network to focus explicitly on artifact-prone regions, E-FPN forces the model to consider implicitly local cues. The association of these two complementary components makes LAA-Net a suitable candidate for fine-grained and generic deepfake detection. As reflected in Figure 1, our approach achieves better and more stable Area Under the Curve (AUC) performance as compared to existing methods [6, 14, 37, 41, 55] regardless of the quality of deepfakes, quantified using the Mask Structural SIMilarity (Mask-SSIM[2]). For a more comprehensive evaluation, in addition to the standard AUC, other metric is reported, namely, Average Precision (AP). We report experiments on several deepfake benchmarks and show that LAA-Net outperforms the state-of-the-art (SoA).

Contributions. In summary, the paper contributions are:

1. A novel multi-task learning method for fine-grained and generic deepfake detection called LAA-Net. It is trained using real data only.
2. An explicit attention mechanism for focusing on vulnerable points combining heatmap-based and self-consistency attention strategies.
3. A new FPN design, called E-FPN, ensures the efficient propagation of low-level features without incurring redundancy[3].
4. Extensive experiments and a comprehensive analysis reported on several benchmarks, namely, FF++ [37], CDF2 [25], DFD [15], DFDC [13], and DFW [57].

Paper Organization. The remainder of the paper is organized as follows: Section 2 reviews related works. Section 3 introduces the proposed approach, and Section 4 reports the experiments and discusses the results. Finally, Section 5 concludes this work and suggests future investigations.

2. Related Works: Attention-based Deepfake Detection

Prior works are diverse in the way they approach the problem of deepfake detection [1, 31, 32, 35, 36, 42, 49]. Earlier methods generally formulate it as a purely binary classification [10, 37], leading to poor generalization capabilities. As a solution, two main strategies have been investigated by the research community, namely, multi-task learning [5–7, 19, 24, 54] and/or pseudo-fake generation [3, 24, 33, 41, 52].

[1] A more formal definition is given in Section 3.2

[2] The Mask-SSIM [25, 29] has been proposed as a metric for quantifying the quality of deepfakes [25]. The Mask-SSIM is computed by computing the similarity in the head region between the fake image and its original version using the SSIM score introduced in [51]. Hence, a higher Mask-SSIM score corresponds to a deepfake of higher quality.

[3] E-FPN is generic and can be used in conjunction with any traditional encoder-decoder architecture.

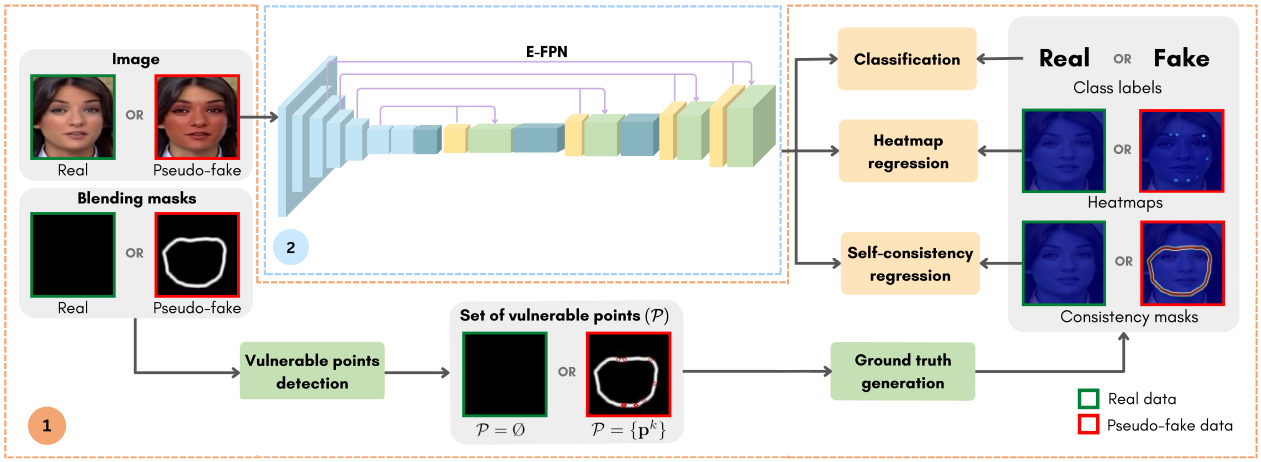


Figure 2. Overview of the proposed LAA-Net approach: it is formed by two components, namely, (1) an *explicit attention mechanism* based on a multi-task learning framework composed of three branches, i.e., the binary classification branch, the heatmap branch, and the self-consistency branch. The heatmap and self-consistency ground-truth data are generated based on the detected vulnerable points, and (2) an *Enhanced Feature Pyramid Networks (E-FPN)* that aggregates multi-scale features.

Despite their great potential, the aforementioned models are less robust when considering high-quality deepfakes. Indeed, these SoA methods mainly employ traditional DNN backbones such as XceptionNet [9] and EfficientNet [44]. Hence, through their successive convolution layers, they implicitly generate global features. As a result, low-level cues, that can be very informative, might be unintentionally ignored, leading to poor detection performance of high-quality deepfakes. It is, therefore, crucial to design adequate strategies for modeling more localized artifacts.

Alternatively, some attention-based methods such as [50, 55] have been proposed. Specifically, they have made attempts to integrate attention modules for implicitly focusing on low-level artifacts [50, 55]. Unfortunately, the two aforementioned methods make use of a unique binary classifier solely trained with real and deepfake images. This means that they do not consider any pseudo-fake generation technique or multi-task learning strategy. Consequently, as demonstrated experimentally, they do not generalize well to unseen datasets in comparison to other recent techniques [3, 41, 52].

3. Localized Artifact Attention Network (LAA-Net)

Our goal is to introduce a method that is robust to high-quality deepfakes yet capable of handling unseen manipulations. Accordingly, we introduce a fine-grained method called Localized Artifact Attention Network (LAA-Net) illustrated in Figure 2. LAA-Net incorporates: (1) an *explicit attention mechanism* and (2) a new architecture based on an *enhanced FPN*, called *E-FPN*.

First, the proposed attention mechanism aims at explicitly focusing on blending artifact-prone pixels referred to as vulnerable points (a formal definition is given in Section 3.1). For that purpose, a hand-free annotation of vulnerable points is proposed by leveraging a blending-based data synthesis. Specifically, a multi-task learning framework composed of three simultaneously optimized branches, namely (a) classification, (b) heatmap regression, and (c) self-consistency regression, is introduced, as depicted in Figure 2. The classification branch predicts whether the input image is fake or real, while the two other branches aim at giving attention to vulnerable pixels. Second, E-FPN allows extracting multi-scale features without injecting redundancy. This enables modeling low-level features, which can better discriminate subtle inconsistencies.

3.1. Explicit Attention to Vulnerable Points

3.1.1 Blending-based Data Synthesis

We start by recalling blending-based data synthesis methods such as [24, 41]. In fact, the proposed method relies on this kind of pseudo-fake generation and, therefore avoids using actual deepfakes and manually annotating data to train the proposed multi-task learning framework. Let us consider a manipulated face image denoted by $\mathbf{I}_\mathrm{M}$. The image $\mathbf{I}_\mathrm{M}$ can be obtained by combining (e.g., blending) two images denoted by $\mathbf{I}_\mathrm{F}$ and $\mathbf{I}_\mathrm{B}$ as follows,

$$\mathbf{I}_\mathrm{M} = \mathbf{M} \odot \mathbf{I}_\mathrm{F} + (1 - \mathbf{M}) \odot \mathbf{I}_\mathrm{B} \,, \tag{1}$$

where $\mathbf{I}_\mathrm{F}$ refers to the foreground image enclosing the desired facial attributes, $\mathbf{I}_\mathrm{B}$ indicates a background image, $\mathbf{M}$ is the deformed Convex Hull mask with values varying between 0 and 1, and $\odot$ denotes the element-wise multiplication operator.

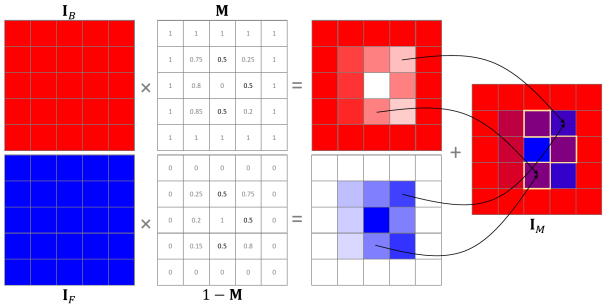


Figure 3. Extraction of the vulnerable points.

3.1.2 Proposed Multi-task Learning Framework

In addition to the deepfake classification branch, the network learns to focus on specific regions by taking advantage of the parallel *Heatmap* and *Self-consistency* branches. Our hypothesis is that deepfake detection can be formulated as a fine-grained classification. Therefore, giving more attention to the vulnerable points should be an effective solution for detecting high-quality deepfakes. For the sake of clarity, we start by formally defining the notion of *'vulnerable points'*.

Definition 1 - *Vulnerable points in a deepfake image are the pixels that are more likely to carry blending artifacts.*

As discussed in Section 3.1.1, any deepfake generation approach involves a blending operation for mixing the background and the foreground of two different images $\mathbf{I}_B$ and $\mathbf{I}_F$, respectively. This implies the presence of blending artifacts regardless of the used generation approach. Thus, we posit that the vulnerable points can be seen as the pixels belonging to the blending regions with the most equivalent contributions from both $\mathbf{I}_B$ and $\mathbf{I}_F$.
In this paper, we assume that we work under a realistic setting where we only have access to real data during training. A blending-based augmentation is, therefore, considered and leveraged for defining vulnerable pixels. Specifically, inspired from [24], a blending boundary mask $\mathbf{B} = (b_{ij})_{i,j\in[\![1,D]\!]}$ is firstly computed as follows,

$$\mathbf{B} = 4\,.\,\mathbf{M} \odot (\mathbf{1} - \mathbf{M})\,, \tag{2}$$

with $\mathbf{1}$ being an all-one matrix. Note that $\mathbf{M}$ is defined in Eq. (1). The variable D is the height and width of $\mathbf{B}$, and b_{ij} its value at the position (i,j). A higher value of b_{ij} indicates that the position (i,j) is more impacted by the blending. Hence, if an input image is real, $\mathbf{B}$ should be set to $\mathbf{0}$. Then, the set of vulnerable pixels denoted by $\mathcal{P}$ is defined as follows,

$$\mathcal{P} = \underset{(i,j)\in[\![1,D]\!]^2}{\operatorname{argmax}}\,(\mathbf{B}), \tag{3}$$

where $[\![\,]\!]$ defines an integer interval. Figure 3 illustrates the extraction of vulnerable points. In the following, we describe how the notion of vulnerable points is used within the heatmap and self-consistency branches.

Heatmap Branch. In general, forgery artifacts not only appear in a single pixel but also affect its surroundings. Hence, considering vulnerable points as well as their neighborhood is more appropriate for effectively discriminating deepfakes, especially in the presence of images with local irregularities caused by noise or illumination changes. To model that, we propose to use a heatmap representation that encodes at the same time the information of both vulnerable points as well as their neighbor points.

More specifically, ground-truth heatmaps are generated by fitting an *Unnormalized Gaussian Distribution* for each pixel $\mathbf{p}^k \in \mathcal{P}$. The pixel $\mathbf{p}^k$ is considered as the center of the Gaussian Mask $\mathbf{G}^k$. To take into account the neighborhood information of $\mathbf{p}^k$, the standard deviation of $\mathbf{G}^k$ is adaptively computed. In particular, inspired from the work of [23], the standard deviation σ_k of $\mathbf{p}^k$ is computed based on the width and the height of the blending boundary mask $\mathbf{B}$ with respect to the point $\mathbf{p}^k$. Similar to [23], a radius r_k is computed based on the size of the set of virtual objects that overlap the mask centered at $\mathbf{p}^k$ with an Intersection over Union (IoU) greater than a threshold t. In all our experiments, we set t to 0.7 and we assume that $\sigma_k = \frac{1}{3}r_k$. Hence, $\mathbf{G}^k = (g^k_{ij})_{i,j\in[\![1,D]\!]}$ is computed as follows,

$$g^k_{ij} = e^{-\frac{i^2+j^2}{2\sigma_k^2}}\,, \tag{4}$$

where i and j refer to the pixel position. The ground-truth heatmap $\mathbf{H}$ is finally constructed by superimposing the set $\mathcal{G} = \{\mathbf{G}^k\}_{k\in[\![1,\text{card}(\mathcal{P})]\!]}$. A figure depicting the heatmap generation process is provided in supplementary materials.

For optimizing the heatmap branch, the following focal loss [26] is used,

$$L_{\text{H}} = \sum_{i,j}^{D} -(1-\tilde{h}_{ij})^{\gamma}\log\tilde{h}_{ij}\,, \tag{5}$$

such that,

$$\tilde{h}_{ij} = \begin{cases} \hat{h}_{ij} & \text{if } h_{ij} = 1\,, \\ 1-\hat{h}_{ij} & \text{otherwise}\,, \end{cases} \tag{6}$$

with $\hat{h}_{ij}$ and h_{ij} being the value of the predicted heatmap $\hat{\mathbf{H}}$ and the ground-truth $\mathbf{H}$ at the pixel location (i,j), respectively. The hyperparameter γ is used to stabilize the adaptive loss weights.

Self-consistency Branch. To enhance the proposed attention mechanism, the idea of learning self-consistency proposed in [54] is revisited to fit our context. Instead of computing the consistency values for each pixel of the mask,

we consider only the vulnerable location. Since the set $\mathcal{P}$ might include more than one pixel (the blending mask can include several pixels with equal values), we randomly choose one of them that we denote by $\mathbf{p}^s$ for generating the self-consistency ground-truth matrix. Hence, the generated matrices denoted by $\mathbf{C}$ are 2-dimensional and not 4-dimensional as in the original method. Given the randomly selected vulnerable point $\mathbf{p}^s = (u, v)$, the self-consistency $\mathbf{C}$ matrix is computed as,

$$\mathbf{C} = \mathbf{1} - |b_{uv}.\mathbf{1} - \mathbf{B}| \,, \tag{7}$$

where $|.|$ refers to the element wise modulus and $\mathbf{1}$ is an all-one matrix.

This refinement allows for reducing the model size and, consequently, the computational cost. It can also be noted that even though our method is inspired by [54], our self-consistency branch is inherently different. In [54], the consistency is calculated between the foreground and background, whereas we measure the consistency between the vulnerable point and the other pixels of the blended mask. The self-consistency loss L_{C} is then computed as a binary cross entropy loss between $\mathbf{C}$ and the predicted self-consistency $\hat{\mathbf{C}}$.

Training Strategy. The network is optimized using the following loss,

$$L = L_{\mathrm{BCE}} + \lambda_1 L_{\mathrm{H}} + \lambda_2 L_{\mathrm{C}} \,, \tag{8}$$

where L_{BCE} denotes the binary cross-entropy classification loss. L_{H} and L_{C} are weighted by the hyperparameters λ_1 and λ_2, respectively. Note that only real and pseudo-fakes are used during training.

3.2. Enhanced Feature Pyramid Network (E-FPN)

Feature Pyramid Networks (FPN) are widely adopted feature extractors capable of complementing global representations with multi-scale low-level features captured at different resolutions [27]. This makes them ideal candidates for implicitly supporting the heatmap and self-consistency branches towards fine-grained deepfake detection. Although some attempts have been made to exploit multi-scale features [14], no previous works have considered FPN in the context of deepfake detection.

Over the last years, several FPN variants have been proposed for numerous computer vision tasks [26, 27, 38, 47]. Nevertheless, these FPN-based methods usually lead to the generation of redundant features, which might, in turn, lead to the overfitting of the model [2]. Moreover, as described in Section 1, small discrepancies are gradually eliminated through the successive convolution blocks [55], going from high-resolution low-level to low-resolution high-level features. Consequently, the last block outputs usually contain global features where local artifact-sensitive features might be discarded. To overcome this issue, we introduce a new alternative referred to as Enhanced Feature Pyramid Network (E-FPN) that is integrated in the proposed LAA-Net architecture. The E-FPN goal is to propagate relevant information from high to low-resolution feature representations.

As shown in Figure 4, we denote the output shape of the $N-1$ latest layers by $(n^{(l)}, D^{(l)}, D^{(l)})$ with $l \in [\![2, N]\!]$. For the sake of simplicity, we assume that the shape of the feature maps is square. For a given layer l, $n^{(l)}$, $D^{(l)}$ and $\mathbf{F}^{(l)}$ correspond, respectively, to its feature dimension, its height and width, and its output features. For strengthening the textural information in the ultimate layer $\mathbf{F}^{(N)}$, we propose to take advantage of the features generated by previous layers $\mathbf{F}^{(l)}$ with $l \in [\![2, N-1]\!]$. Concretely, for each layer l, a convolution followed by a transpose convolution is applied to $\mathbf{F}^{(l+1)}$. The obtained features are denoted by $\mathbf{E}^{(l)}$ and have the same shape as $\mathbf{F}^{(l)}$. Then, a sigmoid function is applied to $\mathbf{E}^{(l)}$ returning probabilities. The latter indicates the pixels that contributed to the final decision. For enriching $\mathbf{F}^{(l+1)}$ while avoiding redundancy related to the most contributing pixels, the features $\mathbf{F}^{(l)}$ are filtered by computing $(1 - \mathrm{sigmoid}(\mathbf{E}^{(l)}))^{\gamma_w}$ resulting in a weighted mask. The latter is concatenated along the same axis with $\mathbf{E}^{(l)}$ for obtaining the final features. This operation is iterated for all the layers $l \in [\![2, N-1]\!]$. In summary, the final representation $\mathbf{F}'^{(l)}$ is obtained as follows,

$$\mathbf{F}'^{(l)} = (\mathbf{F}^{(l)} \odot (1 - \mathrm{sigmoid}(\mathbf{E}^{(l)}))^{\gamma_w} \oplus \mathbf{E}^{(l)}) \,, \tag{9}$$

where $\mathbf{E}^{(l)} = \mathfrak{T}(f(\mathbf{F}'^{(l+1)})$ with $\mathbf{F}'^{(l+1)} = \mathbf{F}^{(l+1)}$ if $l = N-1$, such that f and $\mathfrak{T}$, are respectively the convolution and transpose convolution operators, and $\oplus$ refer to the concatenation operator. The hyper-parameter γ_w is set to 1 in all our experiments. The relevance of E-FPN in the context of deepfake detection is experimentally demonstrated in Section 4, as compared to the traditional FPN.

4. Experiments

In this section, we start by presenting the experimental settings. Then, we compare the performance of LAA-Net to SoA methods, both qualitatively and quantitatively. Finally, we conduct an ablation study to validate the different components of LAA-Net.

4.1. Experimental Settings

Datasets. The FF++ [37] dataset is used for training and validation. In our experiments, we follow the standard splitting protocol of [37]. This dataset contains 1000 original videos and 4000 fake videos generated by four different manipulation methods, namely, Deepfakes (DF) [11], Face2Face (F2F) [46], FaceSwap (FS) [22], and NeuralTextures (NT) [45]. In the training process, we utilize real

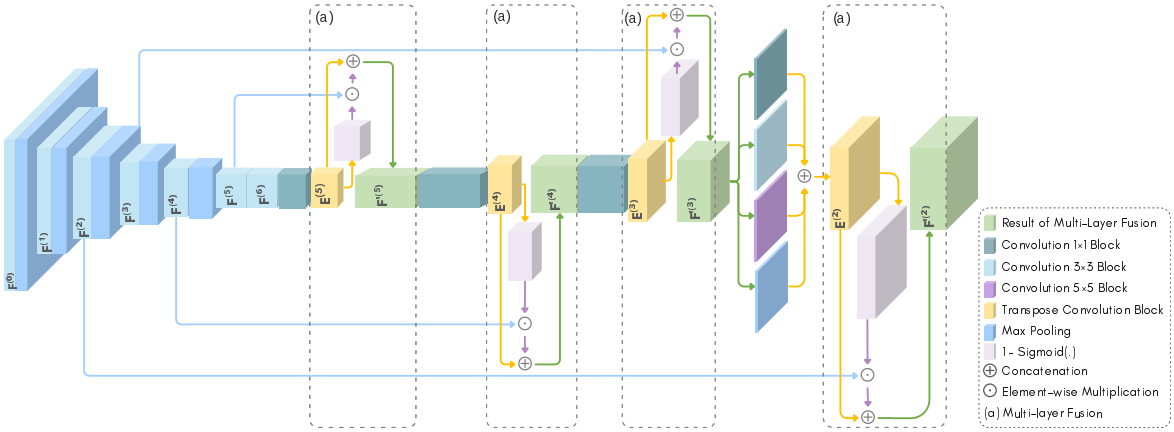


Figure 4. Architecture of the proposed Enhanced Feature Pyramid Network (E-FPN).

Method	Training set		Test set							
			In-dataset	Cross-dataset						
	Real	Fake	FF++	CDF2		DFW		DFD		DFDC	
			AUC (%)	AUC (%)	AP (%)	AUC (%)	AP (%)	AUC (%)	AP (%)	AUC (%)	AP (%)
Xception [37]	✓	✓	99.09	61.18	66.93	65.29	55.37	89.75	85.48	69.90	91.98
FaceXRay+BI [24]	✓	✓	99.20	79.5	-	-	-	95.40	93.34	65.5	-
LRNet [43]	✓	✓	-	53.20	-	-	-	52.29	-	-	-
LocalRL [8]	✓	✓	99.92	78.26	-	-	-	89.24	-	76.53	-
TI^{2}Net [28]	✓	✓	-	68.22	-	-	-	72.03	-	-	-
Multi-attentional [55]	✓	✓	-	68.26	75.25	73.56	73.79	92.95	96.51	63.02	-
RECCE [6]	✓	✓	-	70.93	70.35	68.16	54.41	98.26	79.42	-	-
SFDG [50]	✓	✓	99.53	75.83	-	69.27	-	88.00	-	73.63	-
EIC+IIE [20]	✓	✓	99.32	83.80	-	-	-	93.92	-	81.23	-
AltFreezing [52]	✓	✓	98.6	89.50	-	-	-	98.50	-	-	-
CADDM [14]	✓	✓	99.79	<u>93.88</u>	91.12	<u>74.48</u>	<u>75.23</u>	99.03	<u>99.59</u>	-	-
UCF [53]	✓	✓	-	82.4	-	-	-	94.5	-	80.5	-
Controllable GS [17]	✓	✓	-	84.97	-	-	-	-	-	81.65	-
PCL+I2G [54]	✓		99.11	90.03	-	-	-	99.07	-	74.27	-
SBI [41]	✓		99.64	93.18	85.16	67.47	55.87	97.56	92.79	86.15	93.24
AUNet [3]	✓		99.46	92.77	-	-	-	<u>99.22</u>	-	<u>86.16</u>	-
Ours (w/ BI)	✓		<u>99.95</u>	86.28	<u>91.93</u>	57.13	56.89	**99.51**	**99.80**	69.69	<u>93.67</u>
Ours (w/ SBI)	✓		**99.96**	**95.40**	**97.64**	**80.03**	**81.08**	98.43	99.40	**86.94**	**97.70**

Table 1. In-dataset and Cross-dataset evaluation in terms of AUC and AP on multiple deepfake datasets. **Bold** and <u>Underlined</u> highlight the best and the second-best performance, respectively.

images only to dynamically generate pseudo-fakes, as discussed in Section 3. To evaluate the generalization capability of the proposed approach as well as its robustness to high-quality deepfakes, we test the trained model on four datasets incorporating different quality of deepfakes, namely, Celeb-DFv2 [25] (CDF2), DeepFake Detection [15] (DFD), DeepFake Detection Challenge [13] (DFDC) and Wild Deepfake [57] (DFW). To assess the quality of the considered datasets, we compute the Mask-SSIM[2] for each benchmark. In particular, CDF2 [25] is formed by the most realistic deepfakes with an average Mask-SSIM [29] value of 0.92, followed by DFD and DFDC with an average Mask-SSIM of 0.88 and 0.84, respectively. We note that computing the Mask-SSIM [25] for DFW was not possible since real and fake images are not paired.

Evaluation Metrics. To compare the performance of LAA-Net with the state-of-the-art, we report the common Area Under the Curve (AUC) metric at the video-level and the Average Precision (AP) as in [14, 24, 41, 54]. More metrics, namely, Average Recall (AR) and mean F1-score (mF1) are provided in supplementary materials.

Implementation Details. To train our model, 128 training and 32 validation frames are used. RetinaNet [26] is used to crop faces with a conservative enlargement (by a factor of 1.25) around the face center. Note that all the cropped images are then resized to 384×384. In addition, 68 facial landmarks are extracted per frame using Dlib [21]. We adopt the EFNB4 variant of the EfficientNet [44] pretrained on ImageNet [12]. For each training epoch, 8 frames are dynamically selected and used for online pseudo-fake gen-

Method	Fake	Saturation	Contrast	Block	Noise	Blur	Pixel
Xception [9]	✓	99.3	98.6	99.7	53.8	60.2	74.2
FaceXray [24]	✓	97.6	88.5	99.1	49.8	63.8	88.6
LipForensics [18]	✓	99.9	99.6	87.4	73.8	96.1	95.6
CADDM [14]	✓	99.6	99.8	99.8	**87.4**	**99.0**	98.8
Ours		**99.96**	**99.96**	**99.96**	53.9	98.22	**99.80**

Table 2. Robustness inspection on the FF++ with different types of perturbation. **Bold** and Underline highlight the best and the second-best performance, respectively.

C	H	E-FPN	Test set AUC (%)				
			CDF2	DFD	DFDC	DFW	Avg.
×	×	×	74.54	92.24	70.81	59.81	74.35
×	✓	×	80.89	94.53	77.93	67.12	80.12(↑5.77)
×	×	✓	84.21	95.03	80.68	65.47	81.35(↑7.00)
×	✓	✓	**95.56**	**98.54**	82.21	74.98	87.82(↑13.47)
✓	×	✓	79.87	94.60	71.70	72.47	79.66(↑5.31)
✓	✓	×	91.56	98.27	78.35	73.02	85.30(↑10.95)
✓	✓	✓	95.40	98.43	**86.94**	**80.03**	**90.20(↑15.85)**

Table 3. Ablation study under the cross-dataset setup of the Consistency branch (C), Heatmap branch (H), and E-FPN.

eration. The model is trained for 100 epochs with the SAM optimizer [16], a weight decay of 10^{-4}, and a batch size of 16. We apply a learning rate scheduler that increases from 5.10^{-5} to 2.10^{-4} in the first quarter of the training and then decays to zero in the remaining quarters. We freeze the backbone at the first 6 epochs and only train the remaining layers. For data augmentation, we apply horizontal flipping, random cropping, random scaling, random erasing [56], color jittering, Gaussian noise, blurring, and JPEG compression. The parameters λ_1 and λ_2, defined in Eq. (8), are set to 10 and 100. Furthermore, label smoothing [34] is utilized as a regularizer. To generate pseudo-fakes, two blending synthesis techniques are considered, namely, Blended Images (BI) [24] and Self-Blended Images (SBI) [41]. All experiments are carried out using a GPU Tesla V-100.

4.2. Comparison with State-of-the-art

In-dataset Evaluation. We compare the performance of LAA-Net to existing methods under the in-dataset protocol of [3, 14, 41, 50, 52, 54]. The first column in Table 1 reports the obtained results on the testing set of FF++. It can be seen that all methods achieve competitive performance on the forgeries of the FF++ dataset. Our method combined with SBI outperforms all methods with an AUC of 99.96%, while using only real data for training.

Cross-dataset Evaluation. We evaluate LAA-Net under the challenging cross-dataset setup [6, 50]. Table 1 reports the obtained results on CDF2, DFW, DFD, and DFDC, respectively. It can be noted that LAA-Net achieves state-of-the-art results on the four considered benchmarks, thereby demonstrating its robustness to different quality of deepfakes. The best performance is reached using SBI as a data synthesis, confirming the importance of modeling subtle artifacts. The performance of LAA-Net (w/BI) is slightly superior to LAA-Net (w/SBI) only on DFD, with an improvement of 1.08% and 0.4% of AUC and AP, respectively. A plausible explanation could be the fact that deepfake detection in DFD is less challenging. In fact, numerous methods report AUC and AP scores exceeding 98%.

Furthermore, LAA-Net clearly outperforms attention-based approaches such as Multi-attentional [55] and SFDG [50] by a margin of 27.14% and 19.57% in terms of AUC and AP on CDF2, respectively. This confirms the superior generalization capabilities of LAA-Net as compared to [50, 55]. These results are further supported by high AR and mF1, which are provided in the supplementary materials.

Robustness to Perturbations. Since deepfake videos are easily altered on various social platforms, the robustness of LAA-Net against some common perturbations is investigated. Following the same settings of [14, 18], we evaluate the performance of LAA-Net on FF++ [37] by applying different corruptions. The results are reported in Table 2. As our method focuses on vulnerable points, it can be seen that color-related changes such as saturation and contrast do not impact the performance. However, the proposed method is extremely sensitive to structural perturbations such as Gaussian Noise. In future work, strategies for ensuring more robustness to structural perturbations will be investigated. For instance, denoising methods [30, 40] will be considered for solving this issue.

Qualitative Results. We provide Grad-CAMs [39] in Figure 5, to visualize the image regions in deepfakes that are activated by LAA-Net, SBI [41], Xception [37], and Multi-attentional (MAT) [55] on FF++ [37]. Generally, attention-based methods such as MAT [55] and LAA-Net focus more on localized regions. However, in some cases, MAT [55] concentrates on irrelevant regions such as the background or the inner face areas. Conversely, LAA-Net consistently identifies blending artifacts and shows interesting capabilities on mouth-rendered Neural Textures (NT).

4.3. Ablation Study

Table 3 reports the cross-dataset performance of LAA-Net when discarding the following components: E-FPN, the consistency branch denoted by C and the heatmap branch denoted by H. The best performance is reached when all the components are integrated. It can be seen that the proposed explicit attention mechanism through the heatmap branch contributes more to improving the result. A qualitative example visualizing Grad-CAMs [39] with different components of LAA-Net is also given in Figure 6. The illustration clearly shows that by combining the three components, the network activates more precisely the blending region.

	EFNB4					Test Set AUC (%)							
		E-FPN Integration				CDF2		DFD		DFW		DFDC	
	$\mathbf{F}^{(6)}$	$\mathbf{F}^{(5)}$	$\mathbf{F}^{(4)}$	$\mathbf{F}^{(3)}$	$\mathbf{F}^{(2)}$	FPN	E-FPN	FPN	E-FPN	FPN	E-FPN	FPN	E-FPN
(a)	√					91.56		98.27		73.02		78.35	
(b)	√	√				93.42	91.79	98.59	97.12	73.78	71.39	78.40	75.80
(c)	√	√	√			88.72	92.86	97.96	**98.95**	69.40	74.93	71.91	83.97
(d)	√	√	√	√		88.35	**95.40**	98.89	98.43	70.94	**80.03**	79.02	**86.94**
(e)	√	√	√	√	√	92.16	94.22	96.58	97.31	65.17	72.54	74.31	82.90
Avg						90.84	*93.16*	*98.06*	98.02	70.46	*74.38*	76.40	*81.59*

Table 4. Traditional FPN versus E-FPN, using the SBI-based data synthesis under the cross-dataset protocol. **Bold** and Underline indicate the best and the second-best performance, respectively. We report the results when integrating features $\mathbf{F}^{(i)}$ from different layers.

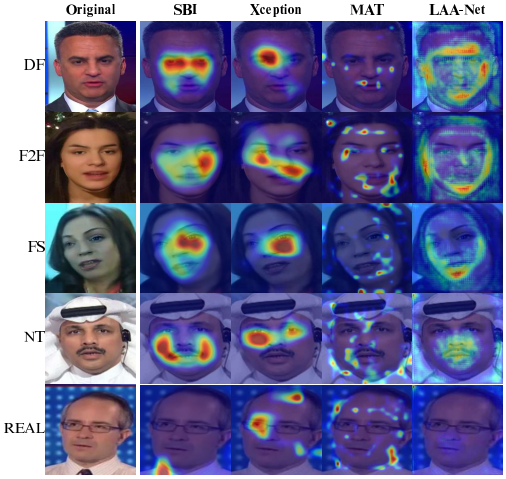


Figure 5. Grad-CAM [39] visualization on different types of manipulation from FF++ [37]. LAA-Net is compared to SBI [41], Xception [37], and MAT [55].

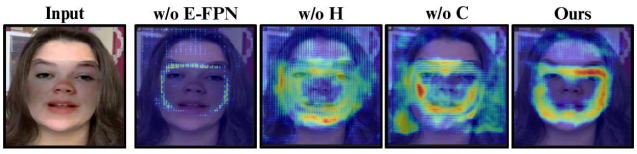


Figure 6. GradCAM [39] visualization of different components in LAA-Net. w/o E-FPN, w/o H, and w/o C refer to ablating E-FPN, heatmap branch, and self-consistency branch, respectively.

4.4. E-FPN versus Traditional FPN

To assess the effectiveness of the low-level features injected by E-FPN into the final feature representation, we combine different feature levels and compare the results of E-FPN and traditional FPN [26, 27] in Table 4. It can be seen that in general E-FPN outperforms FPN except for $\mathbf{F}^{(5)}$. This confirms the relevance of employing multi-scale features and the need for reducing their redundancy in the context of deepfake detection.

4.5. Sensitivity Analysis

In this subsection, we analyze the impact of the two hyperparameters, λ_1 and λ_2 given in Eq. (8). Table 5 shows the experimental results for different values of λ_1 and λ_2. It can be noted that our model is robust to different hyperparameter values, with the best average performance obtained with $\lambda_1 = 10$ and $\lambda_2 = 100$.

λ_1	λ_2	Test Set AUC (%)			
		CDF2	DFDC	DFW	Avg
1	1	90.69	78.12	70.98	79.93
10	10	**95.73**	85.87	73.56	85.05
100	100	93.72	78.60	75.25	82.52
100	10	93.05	83.86	76.72	84.54
10	100	95.40	**86.94**	**80.03**	**87.46**

Table 5. Sensitivity analysis: The impact of the hyper-parameters λ_1 and λ_2 using the cross-dataset protocol on three datasets in terms of AUC.

5. Conclusion

In this paper, a fine-grained deepfake detection method called LAA-Net is introduced with the aim of detecting high-quality deepfakes while remaining generic to unseen manipulations. For that purpose, two different components are proposed. On the one hand, we argue that by making the network focus on the most vulnerable points, we can detect both global and subtle artifacts. To this end, an explicit attention mechanism within a multi-task learning framework is used. In addition to the binary classification branch, heatmap and self-consistency branches are defined with respect to the vulnerable points. On the other hand, a novel E-FPN module for aggregating multi-scale features is proposed; hence enabling the integration of more localized features. The results reported on several benchmarks show the superiority of LAA-Net as compared to the state-of-the-art, including attention-based methods. In future works, strategies for improving the robustness to noise will be investigated. In addition, an attempt to extend this idea by taking into account the temporal dimension will be explored.

Acknowledgment

This work is supported by the Luxembourg National Research Fund, under the BRIDGES2021/IS/16353350/FaKeDeTeR and UNFAKE, ref.16763798 projects, and by POST Luxembourg.

Overview

This document provides supplementary material complementing the main manuscript. It is structured as follows. First, the computation of the self-consistency loss and the ground truth generation of heatmaps are described. Second, more quantitative and qualitative results are provided. In particular, additional metrics are reported for both in-dataset and cross-dataset settings. Moreover, qualitative results comparing E-FPN and FPN are shown.

6. Self-Consistency Loss

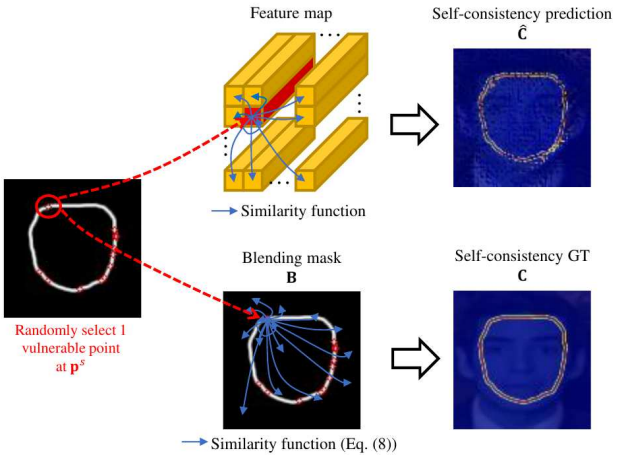


Figure 7. In order to generate the consistency map prediction $\hat{\mathbf{C}}$ as well as the associated ground truth $\mathbf{C}$, we first randomly select a vulnerable point located at $\mathbf{p}^s$. For computing $\hat{\mathbf{C}}$, we measure the similarity between the feature at $\mathbf{p}^s$ (red block) and the features generated from every point. Namely, we use the similarity function in [54]. As for $\mathbf{C}$, we measure the consistency values between the pixel at the $\mathbf{p}^s$ and all pixels in $\mathbf{B}$, as also described in Eq. (7) of the manuscript.

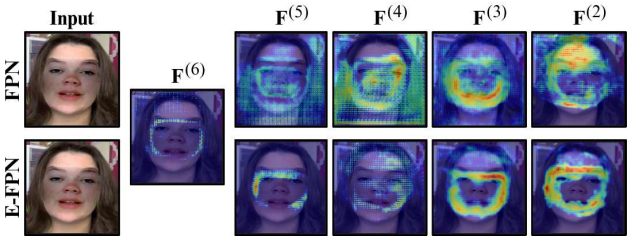


Figure 8. Feature visualization by gradCAM [39] between *E-FPN* and *FPN* with different integration of multi-scale layers. It shows that E-FPN can focus better on artifacts as compared to FPN. The setup details are provided in Table 4 as shown in the manuscript.

To clarify the calculation of the self-consistency loss, we show Figure 7, which illustrates the generation process of the predicted and the ground-truth, $\hat{\mathbf{C}}$ and $\mathbf{C}$, respectively.

Method	Training Set		FF++ [37]				
	Real	Fake	ACC	AUC	AP	AR	mF1
Ours w/ BI [24]	✓		99.03	99.95	99.99	99.21	99.60
Ours w/ SBI [41]	✓		99.04	99.96	99.99	99.29	99.64

Table 6. In-dataset evaluation on FF++ [37] reported by ACC, AUC, AP, AR, and mF1.

The self-consistency loss is a binary cross entropy loss between $\hat{\mathbf{C}}$ and $\mathbf{C}$.

7. Ground Truth Generation of Heatmaps

In this section, we provide more details regarding the generation of ground-truth heatmaps, described in Section 3.1.2. Firstly, a k-th vulnerable point, denoted as $\mathbf{p}^k$, is selected, as shown in Figure 9 (i). Secondly, we measure the height and the width of the blending mask $\mathbf{B}$ at the point $\mathbf{p}^k$ shown as orange lines in Figure 9 (ii). Using the calculated distances, a virtual bounding box is created, indicated by the blue box in Figure 9 (iii). Then, we identify overlapping boxes, illustrated by dashed-line green boxes in Figure 9 (iv), with the Intersection over Union (IoU) greater than a threshold ($t = 0.7$) compared to the virtual bounding box. A radius r_k (solid purple line in Figure 9 (v)) is calculated by forming a tight circle encompassing all these boxes. Finally, an *Unnormalized Gaussian Distribution*, shown as a red circle in Figure 9 (vi), is generated with a standard deviation $\sigma_k = \frac{1}{3} r_k$ (Eq. (4) of the manuscript). The steps are repeated for every vulnerable point $k \in [\![1, \text{card}(\mathcal{P})]\!]$. The final $\mathbf{H}$ is the superimposition of all g_{ij}^k.

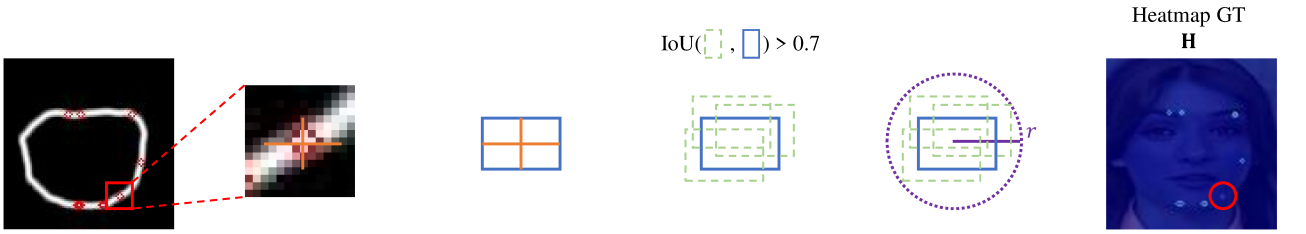
8. Additional Results

In addition to AUC, we provide results using additional metrics, namely, Average Precision (AP), Average Recall (AR), Accuracy (ACC), and mean F1-score (mF1).

Table 6 and Table 7 report the results under the in-dataset and the cross-dataset settings, respectively. Overall, it can be seen that LAA-Net achieves better performances than other state-of-the-art methods.

8.1. Qualitative Results: E-FPN versus FPN

A qualitative comparison between the proposed E-FPN and the traditional FPN with different fusion settings is reported in Figure 8. Using EfficientNet-B4 [44] (EFNB4) as our backbone, the $\mathbf{F}^{(6)}$ refers to the features extracted from the last convolution block in the backbone. In other words, this means that no FPN design is integrated. By gradually aggregating features from lower to higher resolution layers, we can observe the improvement of the forgery localization ability for both E-FPN and FPN. More notably, E-FPN produces more precise activations on the blending boundaries as compared to FPN. This can be explained by the fact that the E-FPN integrates a filtering mechanism for learning less



(i) Given k-th vulnerable point | (ii) Getting virtual object size | (iii) Make a virtual bounding box | (iv) Find all other overlapping boxes of same size | (v) Find radius r_k | (vi) Unnormalized Gaussian with $\sigma_k = \frac{1}{3} r_k$

Figure 9. The generation process of ground truth heatmaps by producing using an *Unnormalized Gaussian Distribution* given a selected vulnerable point.

Method	Fake	Test set (%)															
		CDF2				DFW				DFD				DFDC			
		AUC	AP	AR	mF1	AUC	AP	AR	mF1	AUC	AP	AR	mF1	AUC	AP	AR	mF1
Xception [37]	✓	61.18	66.93	52.40	58.78	65.29	55.37	57.99	56.65	89.75	85.48	79.34	82.29	69.90	91.98	67.07	77.57
FaceXRay+BI [24]	✓	79.5	-	-	-	-	-	-	-	95.40	93.34	-	-	65.5	-	-	-
LRNet [43]	✓	53.20	-	-	-	-	-	-	-	52.29	-	-	-	-	-	-	-
LocalRL [8]	✓	78.26	-	-	-	-	-	-	-	89.24	-	-	-	76.53	-	-	-
TI^{2}Net [28]	✓	68.22	-	-	-	-	-	-	-	72.03	-	-	-	-	-	-	-
Multi-attentional [55]	✓	68.26	75.25	52.40	61.78	73.56	73.79	63.38	68.19	92.95	96.51	60.76	74.57	63.02	-	-	-
RECCE [6]	✓	70.93	70.35	59.48	64.46	68.16	54.41	56.59	55.48	98.26	79.42	69.57	74.17	-	-	-	-
SFDG [50]	✓	75.83	-	-	-	69.27	-	-	-	88.00	-	-	-	73.63	-	-	-
EIC+IIE [20]	✓	83.80	-	-	-	-	-	-	-	93.92	-	-	-	81.23	-	-	-
AltFreezing [52]	✓	89.50	-	-	-	-	-	-	-	98.50	-	-	-	-	-	-	-
CADDM [14]	✓	93.88	91.12	77.00	83.46	74.48	75.23	65.26	69.89	99.03	99.59	82.17	90.04	-	-	-	-
UCF [53]	✓	82.4	-	-	-	-	-	-	-	94.5	-	-	-	80.5	-	-	-
Controllable GS [17]	✓	84.97	-	-	-	-	-	-	-	-	-	-	-	81.65	-	-	-
PCL+I2G [54]		90.03	-	-	-	-	-	-	-	99.07	-	-	-	74.27	-	-	-
SBI [41]		93.18	85.16	82.68	83.90	67.47	55.87	55.82	55.85	97.56	92.79	89.49	91.11	86.15	93.24	71.58	80.99
AUNet [3]		92.77	-	-	-	-	-	-	-	99.22	-	-	-	86.16	-	-	-
Ours (w/ BI)		86.28	91.93	50.01	64.78	57.13	56.89	50.12	53.29	**99.51**	**99.80**	**95.47**	**97.59**	69.69	93.67	50.12	65.30
Ours (w/ SBI)		**95.40**	**97.64**	**87.71**	**92.41**	**80.03**	**81.08**	**65.66**	**72.56**	98.43	99.40	88.55	93.64	**86.94**	**97.70**	**73.37**	**83.81**

Table 7. Cross-dataset evaluation in terms of AUC, AP, AR, and mF1 (%) on CDF2 [25], DFW [57], DFD [15], and DFDC [13]. **Bold** and underlined highlight the best and the second-best performance, respectively. ✓symbol is used to depict methods that utilized both Real data and Fake data for training.

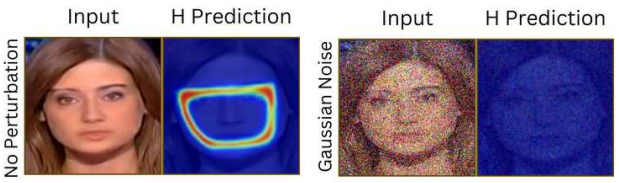


Figure 10. Detection of vulnerable points w/o and w/ Gaussian noise.

noise. In contrast, FPN seems to consider regions outside the blending boundary, which results in lower performance as previously shown in Table 4 - Section 4.4 of the main manuscript.

8.2. Qualitative Results: Gaussian Noise

In Table 2 of the main manuscript, the performance of LAA-Net declined significantly when encountering Gaussian Noise perturbations. One possible reason is that the introduction of noise elevates the difficulty of detecting the vulnerable points. To confirm that, we report the inference of the heatmap before and after applying a Gaussian Noise on a facial image in Figure 10. As it can be observed, the detection of vulnerable points is highly impacted with the introduction of a Gaussian noise.

8.3. Robustness to Compression

To assess the robustness of LAA-Net to compression, we test LAA-Net on the c23 version of FF++, and the overall AUC is equal to 89.30%.

References

[1] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Mesonet: a compact facial video forgery detection network. *CoRR*, abs/1809.00888, 2018. 2

[2] Babajide O Ayinde, Tamer Inanc, and Jacek M Zurada. Reg-

ularizing deep neural networks by enhancing diversity in feature extraction. *IEEE transactions on neural networks and learning systems*, 30(9):2650–2661, 2019. 2, 5

[3] Weiming Bai, Yufan Liu, Zhipeng Zhang, Bing Li, and Weiming Hu. Aunet: Learning relations between action units for face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24709–24719, 2023. 2, 3, 6, 7, 10

[4] Sarah Cahlan. How misinformation helped spark an attempted coup in Gabon. https://wapo.st/3KZARDF, 2020. [Online; accessed 7-March-2023]. 1

[5] Zhixi Cai, Shreya Ghosh, Kalin Stefanov, Abhinav Dhall, Jianfei Cai, Hamid Rezatofighi, Reza Haffari, and Munawar Hayat. Marlin: Masked autoencoder for facial video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1493–1504, 2023. 2

[6] Junyi Cao, Chao Ma, Taiping Yao, Shen Chen, Shouhong Ding, and Xiaokang Yang. End-to-end reconstruction-classification learning for face forgery detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4103–4112, 2022. 1, 2, 6, 7, 10

[7] Liang Chen, Yong Zhang, Yibing Song, Lingqiao Liu, and Jue Wang. Self-supervised learning of adversarial example: Towards good generalizations for deepfake detection, 2022. 1, 2

[8] Shen Chen, Taiping Yao, Yang Chen, Shouhong Ding, Jilin Li, and R. Ji. Local relation learning for face forgery detection. In *AAAI Conference on Artificial Intelligence*, 2021. 6, 10

[9] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 2, 3, 7

[10] Davide Coccomini, Nicola Messina, Claudio Gennaro, and Fabrizio Falchi. Combining efficientnet and vision transformers for video deepfake detection. *CoRR*, abs/2107.02612, 2021. 2

[11] Deepfakes. Faceswapdevs. https://github.com/deepfakes/faceswap, 2019. 5

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 6

[13] Brian Dolhansky, Russ Howes, Ben Pflaum, Nicole Baram, and Cristian Canton-Ferrer. The deepfake detection challenge (DFDC) preview dataset. *CoRR*, abs/1910.08854, 2019. 2, 6, 10

[14] Shichao Dong, Jin Wang, Renhe Ji, Jiajun Liang, Haoqiang Fan, and Zheng Ge. Implicit identity leakage: The stumbling block to improving deepfake detection generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3994–4004, 2023. 1, 2, 5, 6, 7, 10

[15] Nick Dufour and Andrew Gully. Contributing data to deepfake detection research. https://ai.googleblog.com/2019/09/contributing-data-t 2019. 2, 6, 10

[16] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *CoRR*, abs/2010.01412, 2020. 7

[17] Ying Guo, Cheng Zhen, and Pengfei Yan. Controllable guide-space for generalizable face forgery detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20818–20827, 2023. 6, 10

[18] Alexandros Haliassos, Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. Lips don't lie: A generalisable and robust approach to face forgery detection. *CoRR*, abs/2012.07657, 2020. 7

[19] Alexandros Haliassos, Rodrigo Mira, Stavros Petridis, and Maja Pantic. Leveraging real talking faces via self-supervision for robust forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14950–14962, 2022. 2

[20] Baojin Huang, Zhongyuan Wang, Jifan Yang, Jiaxin Ai, Qin Zou, Qian Wang, and Dengpan Ye. Implicit identity driven deepfake face swapping detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4490–4499, 2023. 6, 10

[21] Davis E. King. Dlib-ml: A machine learning toolkit. *J. Mach. Learn. Res.*, 10:1755–1758, 2009. 6

[22] Marek Kowalski. Faceswap. https://github.com/MarekKowalski/FaceSwap, 2018. 5

[23] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. *International Journal of Computer Vision*, 128:642–656, 2018. 4

[24] Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face x-ray for more general face forgery detection. *CoRR*, abs/1912.13458, 2019. 1, 2, 3, 4, 6, 7, 9, 10

[25] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A new dataset for deepfake forensics. *CoRR*, abs/1909.12962, 2019. 1, 2, 6, 10

[26] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *CoRR*, abs/1708.02002, 2017. 4, 5, 6, 8

[27] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 2, 5, 8

[28] Baoping Liu, Bo Liu, Ming Ding, Tianqing Zhu, and Xin Yu. Ti2net: Temporal identity inconsistency network for deepfake detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4691–4700, 2023. 6, 10

[29] Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. Pose guided person image generation. *Advances in neural information processing systems*, 30, 2017. 2, 6

[30] Youssef Mansour and Reinhard Heckel. Zero-shot noise2noise: Efficient image denoising without any data. In

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 14018–14027, 2023. 7

[31] Falko Matern, Christian Riess, and Marc Stamminger. Exploiting visual artifacts to expose deepfakes and face manipulations. In *2019 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pages 83–92, 2019. 2

[32] Nesryne Mejri, Konstantinos Papadopoulos, and Djamila Aouada. Leveraging high-frequency components for deepfake detection. In *2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6, 2021. 2

[33] Nesryne Mejri, Enjie Ghorbel, and Djamila Aouada. Untag: Learning generic features for unsupervised type-agnostic deepfake detection. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023. 2

[34] Rafael Müller, Simon Kornblith, and Geoffrey E. Hinton. When does label smoothing help? *CoRR*, abs/1906.02629, 2019. 7

[35] Huy H. Nguyen, Junichi Yamagishi, and Isao Echizen. Capsule-forensics: Using capsule networks to detect forged images and videos. *CoRR*, abs/1810.11215, 2018. 2

[36] Thanh Thi Nguyen, Cuong M. Nguyen, Dung Tien Nguyen, Duc Thanh Nguyen, and Saeid Nahavandi. Deep learning for deepfakes creation and detection. *CoRR*, abs/1909.11573, 2019. 2

[37] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. FaceForensics++: Learning to detect manipulated facial images. In *International Conference on Computer Vision (ICCV)*, 2019. 1, 2, 5, 6, 7, 8, 9, 10

[38] Selim S. Seferbekov, Vladimir I. Iglovikov, Alexander V. Buslaev, and Alexey A. Shvets. Feature pyramid network for multi-class land segmentation. *CoRR*, abs/1806.03510, 2018. 5

[39] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016. 7, 8, 9

[40] Zehua Sheng, Zhu Yu, Xiongwei Liu, Si-Yuan Cao, Yuqi Liu, Hui-Liang Shen, and Huaqi Zhang. Structure aggregation for cross-spectral stereo image guided denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13997–14006, 2023. 7

[41] Kaede Shiohara and Toshihiko Yamasaki. Detecting deepfakes with self-blended images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18720–18729, 2022. 1, 2, 3, 6, 7, 8, 9, 10

[42] Inder Pal Singh, Nesryne Mejri, van Dat Nguyen, Enjie Ghorbel, and Djamila Aouada. Multi-label deepfake classification. *IEEE Workshop on Multimedia Signal Processing*, 2023. 2

[43] Zekun Sun, Yujie Han, Zeyu Hua, Na Ruan, and Weijia Jia. Improving the efficiency and robustness of deepfakes detection through precise geometric features. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3608–3617, 2021. 6, 10

[44] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019. 2, 3, 6, 9

[45] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *CoRR*, abs/1904.12356, 2019. 5

[46] Justus Thies, Michael Zollhöfer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of RGB videos. *CoRR*, abs/2007.14808, 2020. 5

[47] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: fully convolutional one-stage object detection. *CoRR*, abs/1904.01355, 2019. 5

[48] Jane Wakefield. Deepfake presidents used in Russia-Ukraine war. https://www.bbc.com/news/technology-60780142, 2022. [Online; accessed 7-March-2023]. 1

[49] Run Wang, Lei Ma, Felix Juefei-Xu, Xiaofei Xie, Jian Wang, and Yang Liu. Fakespotter: A simple baseline for spotting ai-synthesized fake faces. *CoRR*, abs/1909.06122, 2019. 2

[50] Yuan Wang, Kun Yu, Chen Chen, Xiyuan Hu, and Silong Peng. Dynamic graph learning with content-guided spatial-frequency relation reasoning for deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7278–7287, 2023. 2, 3, 6, 7, 10

[51] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. 2

[52] Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, and Houqiang Li. Altfreezing for more general video face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4129–4138, 2023. 2, 3, 6, 7, 10

[53] Zhiyuan Yan, Yong Zhang, Yanbo Fan, and Baoyuan Wu. Ucf: Uncovering common features for generalizable deepfake detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22412–22423, 2023. 6, 10

[54] Eric Zhao, Xiang Xu, Mingze Xu, Hui Ding, Yuanjun Xiong, and Wei Xia. Learning self-consistency for deepfake detection. In *ICCV 2021*, 2021. 1, 2, 4, 5, 6, 7, 9, 10

[55] Hanqing Zhao, Wenbo Zhou, Dongdong Chen, Tianyi Wei, Weiming Zhang, and Nenghai Yu. Multi-attentional deepfake detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2185–2194, 2021. 1, 2, 3, 5, 6, 7, 8, 10

[56] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. *CoRR*, abs/1708.04896, 2017. 7

[57] Bojia Zi, Minghao Chang, Jingjing Chen, Xingjun Ma, and Yu-Gang Jiang. Wilddeepfake: A challenging real-world

dataset for deepfake detection. *Proceedings of the 28th ACM International Conference on Multimedia*, 2020. 2, 6, 10