Learning Forgery Signals from Synthetic Depth Targets for Generalizable Deepfake Detection

Supplementary Material

Marko Brodarič[1] Walter Scheirer[2] Deepak Kumar Jain[3] Peter Peer[4] Vitomir Štruc[1]
[1]Faculty of Electrical Engineering, University of Ljubljana, Slovenia
[2]University of Notre Dame, USA [3]Dalian University of Technology, China
[4]Faculty of Computer and Information Science, University of Ljubljana, Slovenia
{marko.brodaric,Vitomir.Struc}@fe.uni-lj.si walter.scheirer@nd.edu dkj@ieee.org
peter.peer@fri.uni-lj.si

Abstract

In the main paper, we introduced FADepth, a depth-centric deepfake detector that adapts a pretrained monocular depth foundation model through synthetic depth supervision and combines the resulting forgery-aware representation with complementary RGB evidence. We evaluated the proposed approach in a cross-dataset setting on six benchmark datasets, compared it with a broad set of state-of-the-art detectors, and analyzed the contribution of the main architectural and training components through ablation studies. In this supplementary material, we provide additional details and analyses that support the main paper. Specifically, we include: (i) *the full implementation details of the two-stage training procedure,* (ii) *an additional FF++ subset-level evaluation across the four manipulation types,* (iii) *a representation-space visualization of the original and forgery-aware DAv2 encoders,* (iv) *a description of the pseudo-deepfake generation procedure used for training,* (v) *details on the synthetic depth-target constructions used in the target-ablation study, and* (vi) *a qualitative analysis of representative failure cases. Together, these additional materials provide a more complete view of the training pipeline, experimental protocol, learned representations, and limitations of the proposed detector.*

1. Implementation Details.

We train the model in two stages. In Stage I, we adapt the DAv2 depth model using SBI-generated pseudo-deepfakes constructed from the pristine FF++ training split, following the original SBI procedure [2]. We use the official DAv2 ViT-L checkpoint [3] to initialize both teacher and student encoders, resize inputs to 518×518, and optimize the student with AdamW for 200 epochs using a learning rate of 10^{-5}, weight decay 10^{-3}, and batch size 16. Unless stated otherwise, we use the final checkpoint from the last epoch to initialize the adapted depth encoder for Stage II.

In Stage II, the depth branch is initialized from the original pretrained DAv2 ViT-L encoder and the Stage-I-adapted encoder, while the RGB branch is initialized from an EfficientNet-B4 model pretrained on SBI pseudo-deepfakes. The depth branch operates on 518×518 inputs and the RGB branch on 380×380 inputs. During the main optimization phase, both DAv2 encoders and the SBI-pretrained EfficientNet-B4 backbone are kept frozen, and only the depth classification head together with the confidence-aware fusion module are trained. We optimize the detector with AdamW for 50 epochs using a learning rate of 3×10^{-4}, weight decay 10^{-3}, and batch size 16. Supervision is applied through the Stage-II classification objective from Section 3, with label smoothing $\epsilon_{\text{ls}} = 0.02$. We then unfreeze the last layers of the depth and RGB backbones and fine-tune the full model for 5 additional epochs with the learning rate reduced to 1.5×10^{-5}. Apart from the stochastic pseudo-deepfake generation, we do not use additional training augmentations. The final Stage-II model is selected based on the fused AUC on the SBI validation split.

2. Additional FF++ Subset Evaluation

In the main paper, FaceForensics++ (FF++) is used as one of the cross-dataset benchmarks and the performance is reported on the aggregate test split. In this supplementary experiment, we further decompose the FF++ evaluation into its four manipulation subsets, i.e., DeepFakes (DF), Face2Face (F2F), FaceSwap (FS), and NeuralTextures (NT). The goal of this analysis is to verify whether the

Table 1. Comparison on FaceForensics++ manipulation subsets. Reported are video-level AUC scores (%). All methods are trained without real deepfake manipulations, using pseudo/synthetic forged samples generated from pristine data. Avg. denotes the mean over the four FF++ manipulation subsets.

Method	DF	F2F	FS	NT	Avg. AUC
Face X-ray + BI [1]	99.17	98.57	98.21	98.13	98.52
PCL + I2G [4]	**100.00**	98.97	99.86	97.63	99.11
FADepth (Ours)	**100.00**	**99.97**	**99.96**	**98.56**	**99.62**

strong FF++ performance of FADepth is consistent across different manipulation mechanisms, rather than being dominated by one particular forgery type. This is particularly relevant in our setting, since FADepth is not trained on real FF++ manipulations, but only on pseudo-manipulated samples generated from pristine data.

We compare FADepth with representative pseudo-forgery-based detectors that follow the same training principle and do not rely on real deepfake samples during learning. Specifically, we include Face X-ray with blending images [1] and PCL+I2G [4]. For all methods, we report video-level AUC scores on the individual FF++ manipulation subsets and compute the average over the four subsets.

The results in Table 1 show that FADepth performs consistently across all four FF++ manipulation subsets. The proposed detector matches the best result on DF and achieves the highest AUC on F2F, FS, and NT, leading to the best average performance of 99.62%. This confirms that the gains observed on the aggregate FF++ benchmark in the main paper are not caused by a single manipulation category, but arise from robust behavior across different forgery-generation mechanisms.

The improvement is especially meaningful for the NT subset, where both competing pseudo-forgery-based methods obtain their lowest scores. In contrast, FADepth reaches the strongest NT performance while also maintaining near-saturated results on DF, F2F, and FS. These results further support the main claim of the paper: learning forgery-aware signals in the depth representation space provides complementary evidence that generalizes well beyond the specific pseudo-manipulations used during training.

3. Representation-Space Visualization

The proposed Stage-I adaptation is designed to reshape a pretrained monocular depth model into a forgery-aware representation extractor, rather than to improve metric depth estimation. To provide additional evidence for this behavior, we visualize the latent representation spaces of the original pretrained DAv2 encoder and the forgery-aware adapted DAv2 encoder in Fig. 1. For both encoders, we extract latent DAv2 features for the same set of 100 pristine and 100 manipulated FF++ images, pool the resulting token/spatial features into one descriptor per image, and project the descriptors to two dimensions using t-SNE. The same samples and visualization protocol are used for both encoders, so the comparison isolates the effect of the Stage-I forgery-aware adaptation.

As shown in Fig. 1, the original pretrained DAv2 encoder produces substantially overlapping embeddings for pristine and manipulated samples. This is expected, since the original model is optimized for generic monocular depth prediction and is not trained to expose manipulation-related facial inconsistencies. In contrast, the adapted encoder yields a more discriminative organization of the feature space, with manipulated and pristine samples becoming more separable. This indicates that the synthetic depth targets do not merely alter the final rendered depth response, but also steer the internal DAv2 features toward forgery-sensitive representations.

Importantly, this visualization is not used as a training objective and should be interpreted as qualitative representation-level evidence. It complements the quantitative results in the main paper and supports the central motivation of FADepth: adapting a depth foundation model with localized synthetic depth supervision encourages the encoder to retain useful geometric priors while becoming sensitive to manipulation-induced deviations.

4. Pseudo-Deepfake Generation

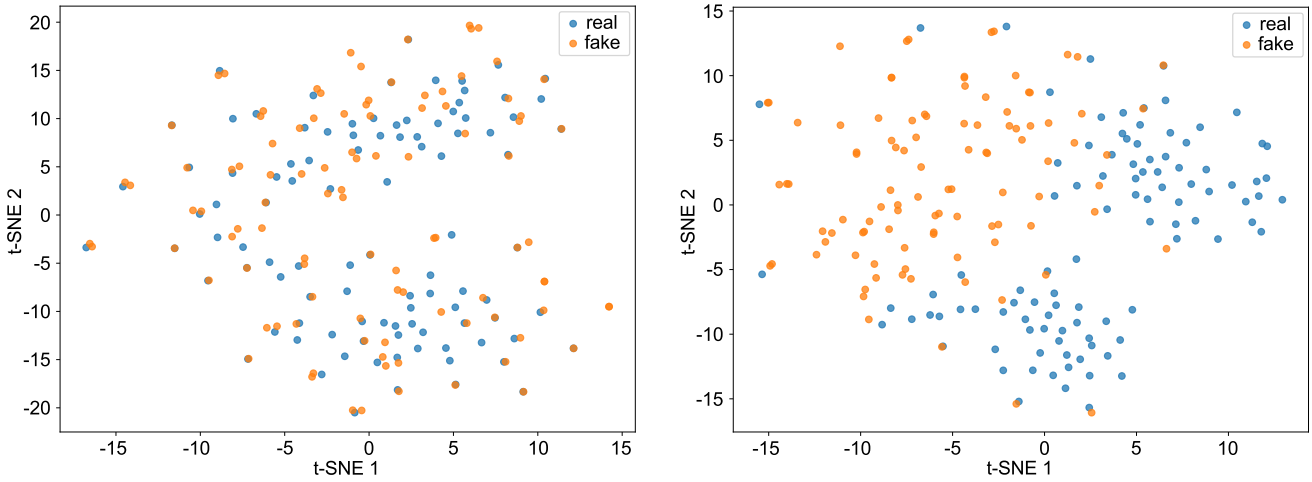
In this section, we describe the pseudo-deepfake generation procedure used throughout our training pipeline. The procedure follows the self-blending principle of SBI [2]. The generation process corresponds to the pseudo-manipulation step shown in Fig. 2 of the main paper.

We generate pseudo-deepfakes from pristine FF++ training frames. For each selected frame, we load the corresponding facial landmarks and RetinaFace detections. To ensure that the crop is aligned with the annotated facial region, we first compute a landmark-based bounding box and then select the RetinaFace box with the highest IoU with this landmark box. During training, horizontal flipping is applied with probability 0.5, jointly to the image, landmarks, and bounding box. The face region is then cropped with a margin, yielding an aligned facial image x and the corresponding landmarks ℓ.

Given the cropped face image $x \in \mathbb{R}^{H\times W\times 3}$ and its landmarks ℓ, we first construct a binary facial mask. With probability 0.25, only the standard 68 landmarks are used, while otherwise the extended landmark set is retained. This introduces variability in the spatial support of the pseudo-manipulation. The initial mask is obtained as the convex hull of the selected landmarks:

$$M_0 = \mathrm{Hull}(\ell), \tag{1}$$

where $M_0 \in \{0,1\}^{H\times W}$ indicates the facial region selected for blending.



(a) Original pretrained DAv2 **(b) Forgery-aware adapted DAv2**

Figure 1. t-SNE visualization of latent DAv2 representations before and after forgery-aware adaptation. The left plot shows the representation space of the original pretrained DAv2 encoder, while the right plot shows the representation space of the adapted forgery-aware encoder. Each plot contains 100 pristine and 100 manipulated FF++ images. Compared with the original encoder, where authentic and manipulated samples remain strongly overlapped, the adapted encoder produces a more discriminative representation space with clearer real/fake separation.

The pseudo-deepfake is generated by blending two versions of the same face image. We denote the target image by x_t and the source image by x_s. Initially, both are copied from the same pristine face image,

$$x_t = x, \qquad x_s = x. \tag{2}$$

To introduce appearance discrepancies between the two versions, source-style perturbations are applied to either x_s or x_t with equal probability. These perturbations consist of color and image-quality transformations, including RGB shifts, hue/saturation/value changes, brightness and contrast changes, and either random downscaling or sharpening. This step creates two slightly different versions of the same identity, thereby simulating the local inconsistencies commonly introduced by face manipulation pipelines without requiring real deepfake samples.

Next, the source image and the blending mask are spatially perturbed. We first apply a random affine transformation to both x_s and M_0, with small translation and scaling ranges:

$$x'_s, M_1 = \mathcal{A}(x_s, M_0), \tag{3}$$

where $\mathcal{A}(\cdot)$ denotes an affine transformation with horizontal translation sampled from $[-0.03, 0.03]$, vertical translation sampled from $[-0.015, 0.015]$, and scale sampled from $[0.95, 1/0.95]$. The mask is then further perturbed with an elastic deformation,

$$M = \mathcal{E}(M_1), \tag{4}$$

where $\mathcal{E}(\cdot)$ denotes a smooth non-rigid deformation field applied to the mask. This deformation locally bends and stretches the mask boundary while preserving its overall facial support, which increases shape variability and avoids overly regular blending regions. These transformations introduce spatial misalignment between the source and target faces and diversify the shape of the manipulated region.

The final pseudo-deepfake image is obtained using the dynamic blending operator. Given the perturbed source image x'_s, target image x_t, and mask M, the operator first converts the binary mask into a soft blending mask by smoothing its boundaries and then combines source and target pixels according to the resulting spatially varying weights. This produces subtle transition artifacts around the manipulated region, rather than a hard copy-paste boundary. Formally, the operator produces a blended image x^f and the corresponding soft manipulation mask:

$$x^f, m = \mathrm{Blend}(x'_s, x_t, M). \tag{5}$$

The original target image is retained as the real counterpart,

$$x^r = x_t. \tag{6}$$

Thus, each training sample consists of a paired real and pseudo-manipulated image (x^r, x^f), together with the manipulation mask m. This paired construction is important for our Stage-I depth adaptation, since the same facial identity and crop are observed in both authentic and manipulated form, while the mask provides the spatial support where the synthetic depth target is altered.

During training, we apply additional image-level augmentations jointly to the real and pseudo-manipulated images. These include RGB shifts, hue/saturation/value changes, brightness and contrast changes, and JPEG compression. The same sampled transformation is applied to both x^r and x^f, which prevents the detector from relying on augmentation-induced differences between the two branches of the pair.

The described generation process creates pseudo-deepfakes by introducing controlled appearance and geometric inconsistencies into pristine facial images. Since both the source and target originate from the same identity, the generated samples avoid identity-level shortcuts and instead emphasize local blending artifacts, color/quality discrepancies, and spatial misalignment. This makes the procedure well suited for generalizable deepfake detection and, more importantly for our method, provides an explicit manipulation mask that allows us to translate RGB-space pseudo-manipulations into localized synthetic depth supervision.

5. Details on Synthetic Depth-Target Ablations

Fig. 5 of the main paper compares different strategies for constructing the synthetic supervisory depth targets used during the Stage-I adaptation of DAv2. Here, we provide additional implementation details for this ablation. All compared variants follow the same training protocol and differ only in the way the target depth map is modified inside the pseudo-manipulated region.

Given a real image x^r, a pseudo-manipulated image x^f, and the corresponding manipulation mask m, we first compute reference depth predictions using the frozen pretrained DAv2 model $T(\cdot)$:

$$D^r = T(x^r), \qquad D^f = T(x^f). \tag{7}$$

The mask is normalized as

$$P = \frac{m}{\max(m)}, \tag{8}$$

where $P \in [0,1]^{H\times W}$. For authentic samples, all variants use the same unchanged target,

$$\widetilde{D}^r = D^r. \tag{9}$$

Thus, the ablation concerns only the construction of $\widetilde{D}^f$ for pseudo-manipulated samples.

Mean-anchored plateau. The first strategy corresponds to the proposed construction, shown in Fig. 5(a) of the main paper. For completeness, we briefly restate it here. The masked region is moved toward the image-specific mean depth value,

$$\mu(D^f) = \frac{1}{HW}\sum_{u=1}^{H}\sum_{v=1}^{W} D^f_{u,v}, \tag{10}$$

resulting in the target

$$\begin{aligned}\widetilde{D}^f_{\text{mean}} &= D^f + P \odot \big(\mu(D^f) - D^f\big) \\ &= (1-P)\odot D^f + P \odot \mu(D^f).\end{aligned} \tag{11}$$

This design preserves the original prediction outside the manipulated region and replaces the masked region with an image-adaptive plateau that remains within the natural range of the depth prediction.

Out-of-range plateau with a large margin. The second strategy, shown in Fig. 5(b) of the main paper, evaluates whether a stronger supervisory signal can be obtained by forcing the manipulated region to an out-of-range constant plateau. In general, such a plateau can be defined either below the minimum predicted depth or above the maximum predicted depth. In our implementation, we instantiate the below-range variant by setting

$$\tau^{0.3}_{\text{out}} = \min(D^f) - 0.3\cdot\max(D^f), \tag{12}$$

and constructing the target as

$$\begin{aligned}\widetilde{D}^f_{\text{out},0.3} &= D^f + P \odot \big(\tau^{0.3}_{\text{out}} - D^f\big) \\ &= (1-P)\odot D^f + P \odot \tau^{0.3}_{\text{out}}.\end{aligned} \tag{13}$$

The purpose of this variant is to test an aggressive target that clearly separates manipulated regions from the original depth manifold. However, because the imposed plateau lies outside the natural prediction range of the frozen DAv2 model, the resulting supervision can be too strong. As observed in Fig. 5(b), this may destabilize the adapted model and produce responses that are less localized to the manipulated region.

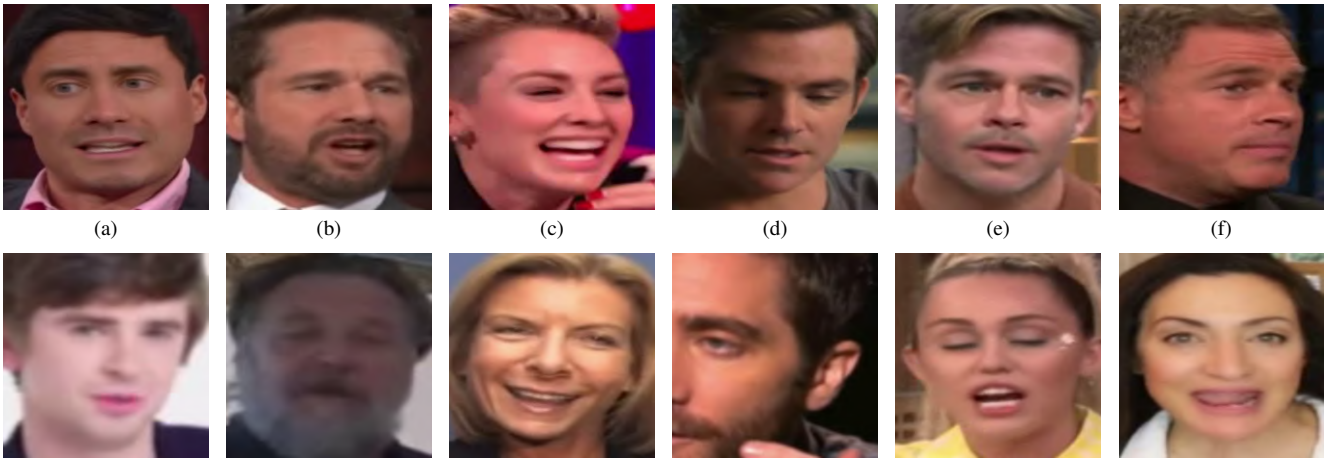
Out-of-range plateau with a small margin. The third strategy, shown in Fig. 5(c), follows the same idea as the previous variant but reduces the magnitude of the out-of-range displacement. In our implementation, the plateau is defined as

$$\tau^{0.01}_{\text{out}} = \min(D^f) - 0.01\cdot\max(D^f), \tag{14}$$

leading to

$$\begin{aligned}\widetilde{D}^f_{\text{out},0.01} &= D^f + P \odot \big(\tau^{0.01}_{\text{out}} - D^f\big) \\ &= (1-P)\odot D^f + P \odot \tau^{0.01}_{\text{out}}.\end{aligned} \tag{15}$$

Compared to the large-margin version, this target remains closer to the original depth range and therefore reduces



(a) (b) (c) (d) (e) (f)

(g) (h) (i) (j) (k) (l)

Figure 2. Representative failure cases of FADepth. The first row shows false negatives, where manipulated samples are classified as authentic. The second row shows false positives, where authentic samples are classified as manipulated.

the severity of the imposed perturbation. Nevertheless, it still relies on a range-violating constant plateau and is consequently less image-adaptive than the proposed mean-anchored target.

Fixed constant plateau. The fourth strategy, shown in Fig. 5(d), replaces the manipulated region with a fixed constant value. More generally, this family of targets can be written as

$$\widetilde{D}^{f}_{\text{const}} = (1 - P) \odot D^{f} + P \odot c, \tag{16}$$

where c is a predefined constant. In the implementation used for Fig. 5(d), we set $c = 0$, which gives

$$\widetilde{D}^{f}_{\text{const}} = D^{f} \odot (1 - P). \tag{17}$$

This construction is simple and produces a clear plateau inside the manipulated area. However, unlike the proposed mean-anchored strategy, the imposed value is not adapted to the depth distribution of the current image. Depending on the scale of D^{f}, the constant target can therefore be either too weak or too disruptive, leading to less consistent forgery-aware behavior.

All variants are trained with the same DAv2 initialization, pseudo-manipulated samples, masks, regression loss, and optimization protocol. The differences shown in Fig. 5 of the main paper therefore isolate the effect of the target construction itself. The proposed mean-anchored plateau provides the most stable trade-off: it preserves the pre-trained depth behavior on authentic content, keeps the altered target within the image-specific prediction range, and still introduces a localized depth deviation inside the manipulated region. In contrast, the out-of-range strategies impose stronger but less stable supervision, while the fixed constant plateau lacks image adaptivity. This explains why the proposed target construction produces the cleanest and most localized forgery-aware response among the compared variants.

6. Failure Case Analysis

We additionally analyze representative failure cases of FADepth in Fig. 2. The first row (Fig. 2(a)–(f)) shows false negatives, i.e., manipulated samples classified as authentic, while the second row (Fig. 2(g)–(l)) shows false positives, i.e., authentic samples classified as manipulated. The examples illustrate that most errors occur in visually ambiguous cases, where the evidence available to the detector is either too weak to reveal the manipulation or too similar to the artifacts observed in real low-quality videos.

For the false negatives in the first row, the manipulated faces are generally highly realistic and preserve plausible global facial structure. In several cases, the visible manipulation evidence is subtle and localized, while the remaining facial appearance is coherent, making the samples challenging even for human inspection (e.g., Fig. 2(a), (d), and (e)). The side-profile example (Fig. 2(f)) further illustrates this difficulty: although the eye region appears imperfect, the non-frontal pose reduces visible facial symmetry and weakens the canonical geometric cues that are useful for depth-centric reasoning. Similarly, blur, compression, and limited resolution (e.g., Fig. 2(b), (c), and (e)) can suppress fine-grained manipulation traces, making it difficult to distinguish whether local irregularities originate from the forgery process or from ordinary video degradation.

The false positives in the second row reveal the complementary failure mode. These are authentic frames, but they contain visual patterns that resemble manipulation artifacts. Some errors are caused by imperfect face cropping, where only part of the face is visible and the expected facial ge-

ometry is disrupted before classification (Fig. 2(j)). Other examples contain local occlusions or isolated artifacts in semantically important regions, such as the eye or cheek area (Fig. 2(k)). We also observe authentic frames with blending-like boundaries (Fig. 2(i)), unusual mouth appearance (Fig. 2(l)), strong blur (Fig. 2(g) and (h)), poor illumination (Fig. 2(h)), or severe compression (Fig. 2(g) and (h)). Such effects can resemble the local inconsistencies introduced by pseudo-manipulations during training and may therefore trigger a fake prediction.

Overall, the failure cases indicate that FADepth is sensitive to localized structural and appearance inconsistencies, which is beneficial for detecting manipulated faces but can also lead to errors in ambiguous conditions. False negatives typically arise when high-quality deepfakes preserve global facial geometry and hide artifacts under compression or pose variation (e.g., Fig. 2(a)–(f)). False positives, on the other hand, occur when authentic frames contain acquisition, cropping, or compression artifacts that mimic manipulation-like evidence (e.g., Fig. 2(g)–(l)).

References

[1] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo. Face x-ray for more general face forgery detection. In *CVPR*, 2020.

[2] K. Shiohara and T. Yamasaki. Detecting deepfakes with self-blended images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18720–18729, 2022.

[3] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2024.

[4] T. Zhao, X. Xu, M. Xu, H. Ding, Y. Xiong, and W. Xia. Learning self-consistency for deepfake detection. In *IEEE/CVF ICCV*, 2021.