Can We Leave Deepfake Data Behind in Training Deepfake Detector ?

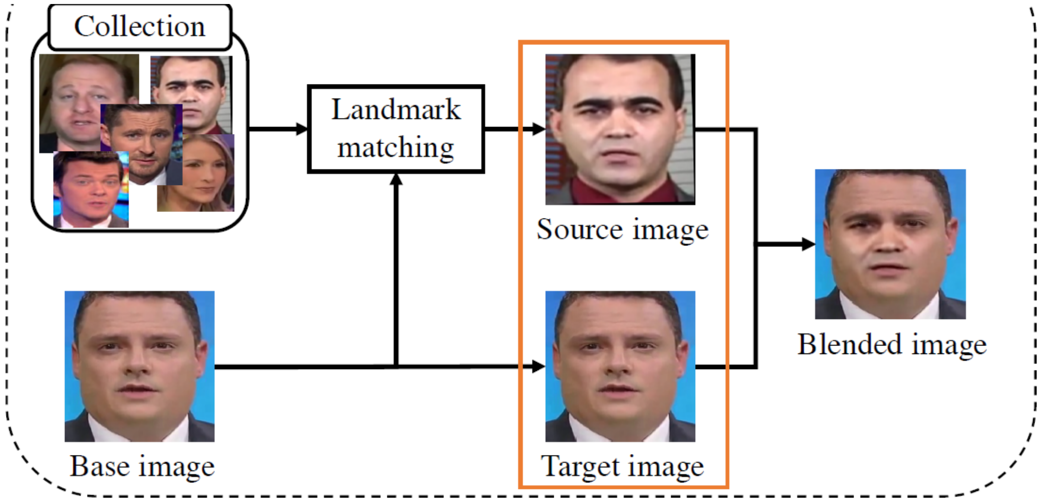
Jikang Cheng, Zhiyuan Yan, Ying Zhang, Yuanhao Luo, Zhongyuan Wang, Chen Li

Background

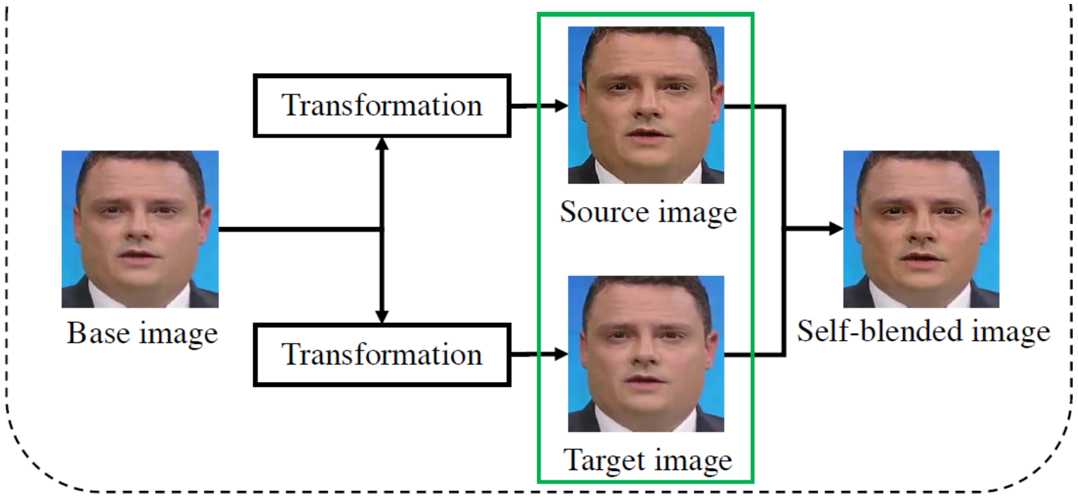
Learning common forgery clues without overfitting to the specific one

Recently advanced methods take only non-DL synthetic faces (Blendfake) during training, e.g. SBI and BI. Actual Deepfake training data is excluded.

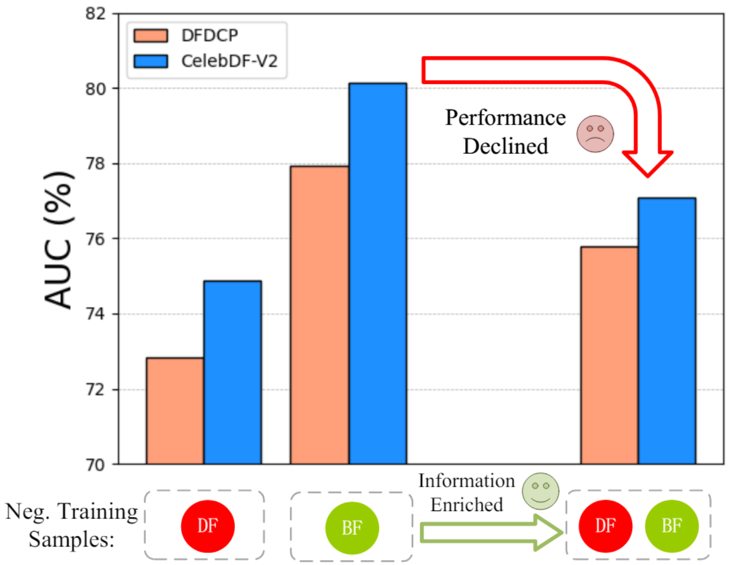
BI (Blend Image)



SBI (Self-Blend Image)

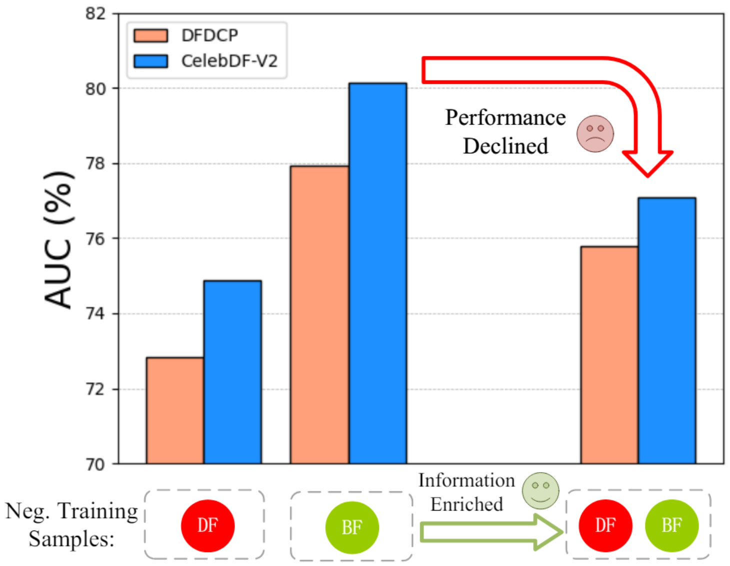


Observation



- General without specific forgery clues
- Harder samples than deepfake, making the detector more sensitive.

Observation



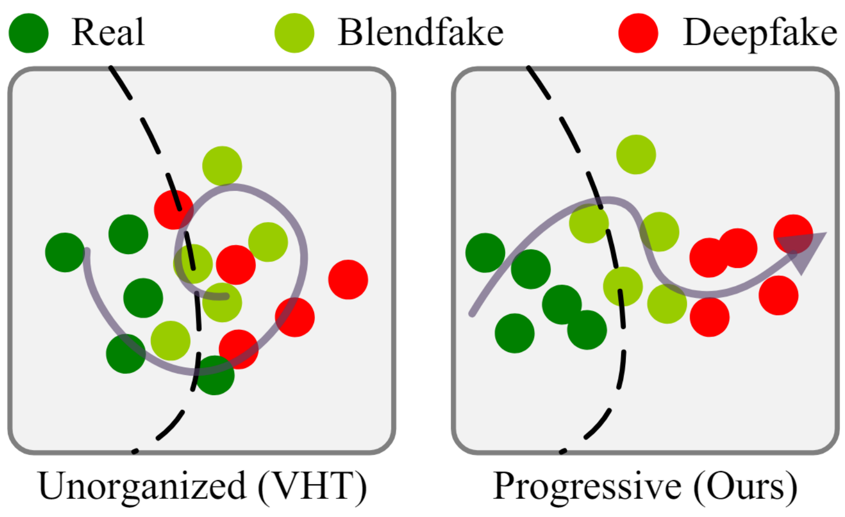
- General without specific forgery clues
- Harder samples than deepfake, making the detector more sensitive.

Are deepfake faces actually worthless for detector training?

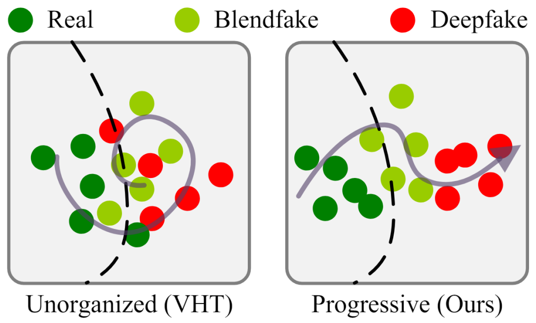
They should include extra useful information.

Basic Idea

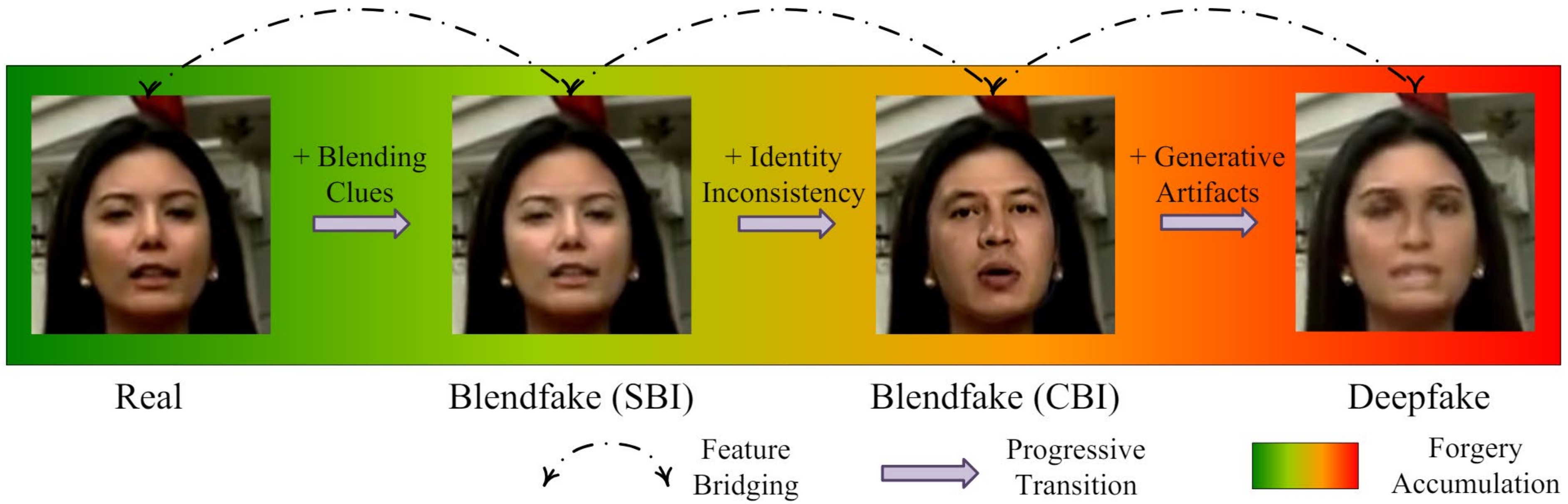
- Unorganized latent-space distribution
- Fail to disentangle the learned representation.



Basic Idea

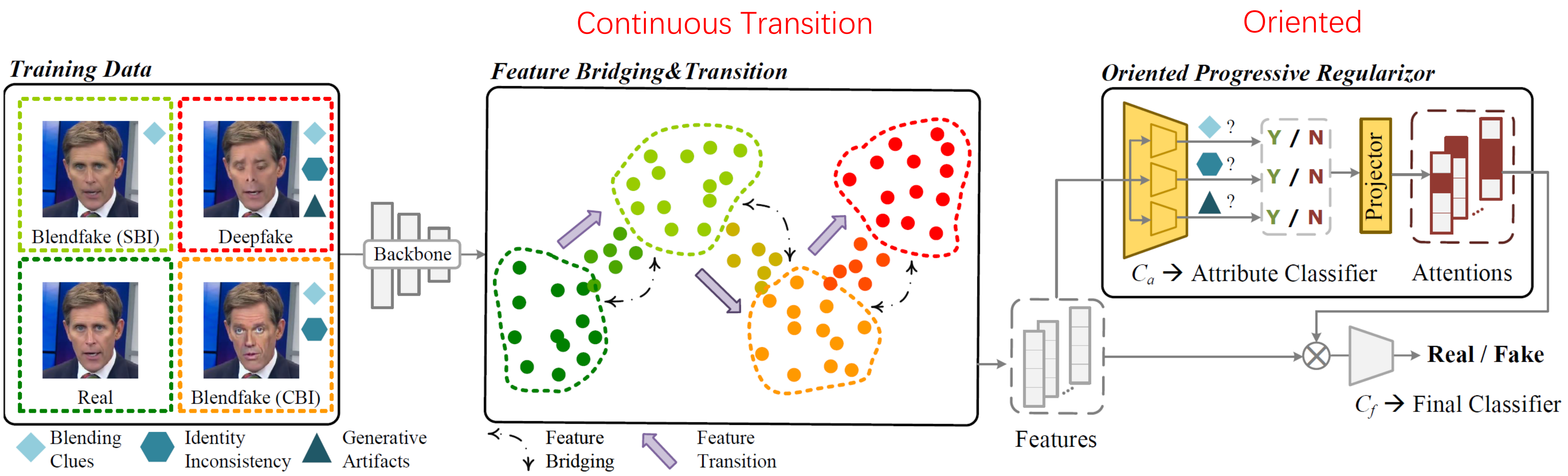


- Real->Blendfake->Deepfake is a continuous progressive process.




Real
Blendfake (SBI)
Blendfake (CBI)
Deepfake
Real
Blendfake (SBI)
Blendfake (CBI)
Deepfake

Method (ProDet)



Forgery Attributes Accumulation represents Oriented Separated Anchoring

Experiments: Comparison

Method	Venues	FF++	CDFv1	CDFv2	DFDCP	DFDC	C-Avg.
Xception [10]	CVPR'17	0.9637	0.7794	0.7365	0.7374	0.7077	0.7403
Meso4 [1]	WIFS'18	0.6077	0.7358	0.6091	0.5994	0.5560	0.6251
FWA [26]	CVPRW'18	0.8765	0.7897	0.6680	0.6375	0.6132	0.6771
EfficientB4 [38]	ICML'19	0.9567	0.7909	0.7487	0.7283	0.6955	0.7408
Capsule [31]	ICASSP'19	0.8421	0.7909	0.7472	0.6568	0.6465	0.7104
CNN-Aug [42]	CVPR'20	0.8493	0.7420	0.7027	0.6170	0.6361	0.6745
X-ray [25]	CVPR'20	0.9592	0.7093	0.6786	0.6942	0.6326	0.6787
FFD [12]	CVPR'20	0.9624	0.7840	0.7435	0.7426	0.7029	0.7433
F3Net [33]	ECCV'20	0.9635	0.7769	0.7352	0.7354	0.7021	0.7374
SPSL [29]	CVPR'21	0.9610	0.8150	0.7650	0.7408	0.7040	0.7562
SRM [30]	CVPR'21	0.9576	0.7926	0.7552	0.7408	0.6995	0.7470
I2G-PCL [48]	ICCV'21	0.9312	0.7112	0.6992	0.7358	0.6555	0.7004
CORE [32]	CVPRW'22	0.9638	0.7798	0.7428	0.7341	0.7049	0.7404
Recce [6]	CVPR'22	0.9621	0.7677	0.7319	0.7419	0.7133	0.7387
SLADD [7]	CVPR'22	0.9691	0.8015	0.7403	0.7531	0.7170	0.7530
SBI [36]	CVPR'22	0.8176	0.8311	0.8015	0.7794	0.7139	0.7814
IID [22]	CVPR'23	**0.9743**	0.7578	0.7687	0.7622	0.6951	0.7462
UCF [44]	ICCV'23	0.9705	0.7793	0.7527	0.7594	0.7191	0.7526
Ours	-	0.9591	**0.9094** ($\uparrow$ 9.42%)	**0.8448** ($\uparrow$ 5.40%)	**0.8116** ($\uparrow$ 4.13%)	**0.7240** ($\uparrow$ 0.68%)	**0.8225** ($\uparrow$ 5.26%)

Experiments: Ablation Study

Table 2: Ablations for each network component (AUC↑ and EER↓). All variants are trained on FF++ (in-dataset) and evaluated on other datasets (cross-dataset). BF-only represents using only blendfake data as the negative samples. M-C, M-L, and TB denotes Multi-Class, Multi-Label, and Triplet Binary strategies, respectively.

Variant	FF++		CDFv1		CDFv2		DFDCP		C-Avg.	
	AUC	EER	AUC	EER	AUC	EER	AUC	EER	AUC	EER
BF-only	0.8096	0.2811	0.8413	0.2171	0.8006	0.2804	0.7791	0.3019	0.8070	0.2665
VHT	0.9353	0.1435	0.8145	0.2603	0.7710	0.2768	0.7577	0.3026	0.7811	0.2799
w/o L_o	0.9311	0.1493	0.8401	0.2281	0.7959	0.2705	0.7901	0.2737	0.8087	0.2574
w/o FB	0.9601	0.0816	0.8696	0.2001	0.8278	0.2537	0.8037	0.2811	0.8337	0.2449
w/o L_t	0.9535	0.1326	0.8890	0.1799	0.8356	0.2301	**0.8174**	0.2636	0.8473	0.2245
M-C	**0.9677**	**0.0835**	0.8630	0.2108	0.8092	0.2739	0.7965	0.2658	0.8229	0.2501
M-L	0.9576	0.0994	0.8757	0.1893	0.8229	0.2533	0.7939	0.2748	0.8308	0.2391
TB (**Ours**)	0.9591	0.1014	**0.9094**	**0.1688**	**0.8448**	**0.2136**	0.8116	**0.2628**	**0.8553**	**0.2151**

Table 3: Ablations on leveraging oriented anchors progressively (AUC). All variants are trained on FF++ (in-dataset) and evaluated on other datasets (cross-dataset).

SBI	SBI-FB	CBI	CBI-FB	DF	DF-FB	FF++	CDFv1	CDFv2	DFDCP	C-Avg.
✓						0.8176	0.8311	0.8015	0.7794	0.8040
✓	✓					0.8343	0.8507	0.8136	0.7659	0.8101
✓	✓	✓				0.8191	0.8439	0.7917	0.7910	0.8089
✓	✓	✓	✓			0.8210	0.8551	0.8151	0.8081	0.8254
✓	✓	✓	✓	✓		0.9539	0.8891	0.8336	0.7947	0.8391
✓	✓	✓	✓	✓	✓	**0.9591**	**0.9094**	**0.8448**	**0.8116**	**0.8553**

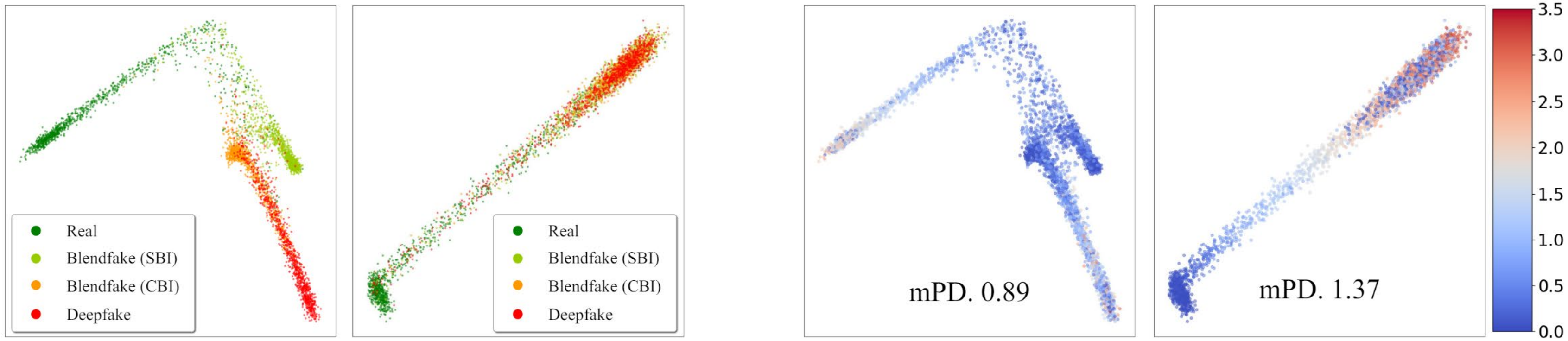
Table 7: Generalization evaluations on comprehensive datasets.

Methods	DFD	DF1.0	FAVC	WDF	DiffSwap	UniFace	E4S	BlendFace	MobileSwap
DF-only	0.8144/0.8621	0.7462/0.7474	0.8404/0.9150	0.7275/0.6883	0.7959/-	0.7775/0.8212	0.6514/0.6955	0.7813/0.8296	0.8475/0.9053
BF-only	0.8378/0.8901	0.7345/0.7811	0.8627/0.9237	0.7563/**0.7965**	0.8265/-	0.6745/0.6998	0.6797/0.7113	0.8041/0.8529	0.8883/0.9399
VHT	0.8215/0.8505	0.7702/0.8312	0.8402/0.9125	0.7263/0.7811	0.7961/-	**0.8445**/0.8979	0.6704/0.7101	0.8311/0.8930	0.8729/0.9295
Ours	**0.8581/0.9073**	**0.7902/0.8536**	**0.9077/0.9766**	**0.7718**/0.8287	**0.8459**/-	0.8441/**0.9077**	**0.7103/0.7711**	**0.8619/0.9287**	**0.9285/0.9748**

Experiments: latent-space organization

$$\mathrm{PD} = \sum_{i=1}^{n} \frac{\sqrt{(\mathbf{F}_i)^2 + (\mathbf{F})^2}}{n\mathbf{F}_{std}},$$

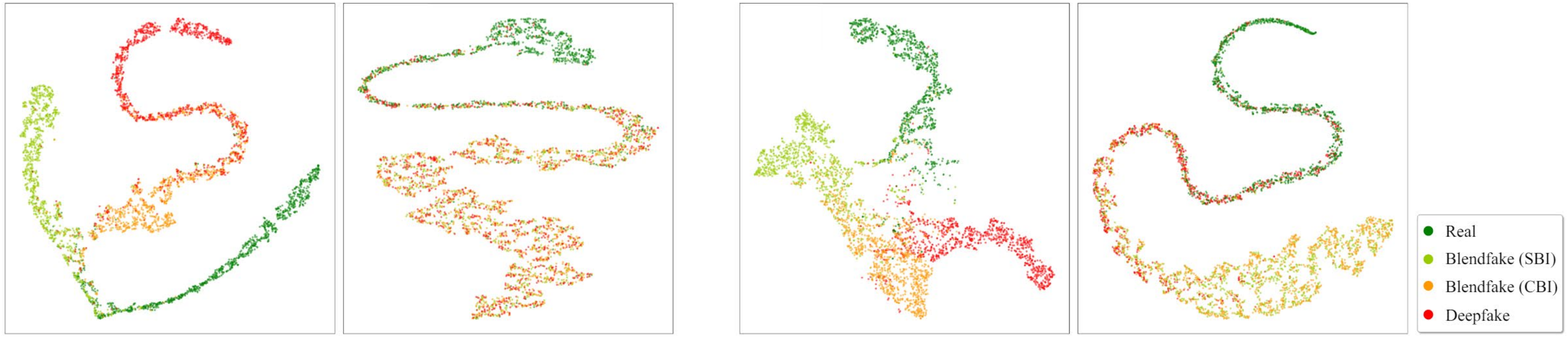
Toy



Ours VHT Ours VHT

(a) (b)

T-SNE



Ours (toy) VHT (toy) Ours (original) VHT (original)

Conclusion

- Reversing a **stereotype** in research community, that is, deepfake is left behind during detector training.
- Proposing to leverage the **progressive transition** from Real->Blenefake->Deepfake.
- Designing ProDet to effectively simulate progressive transition with **superior generalization ability**.