ACM multimedia 2020 Seattle

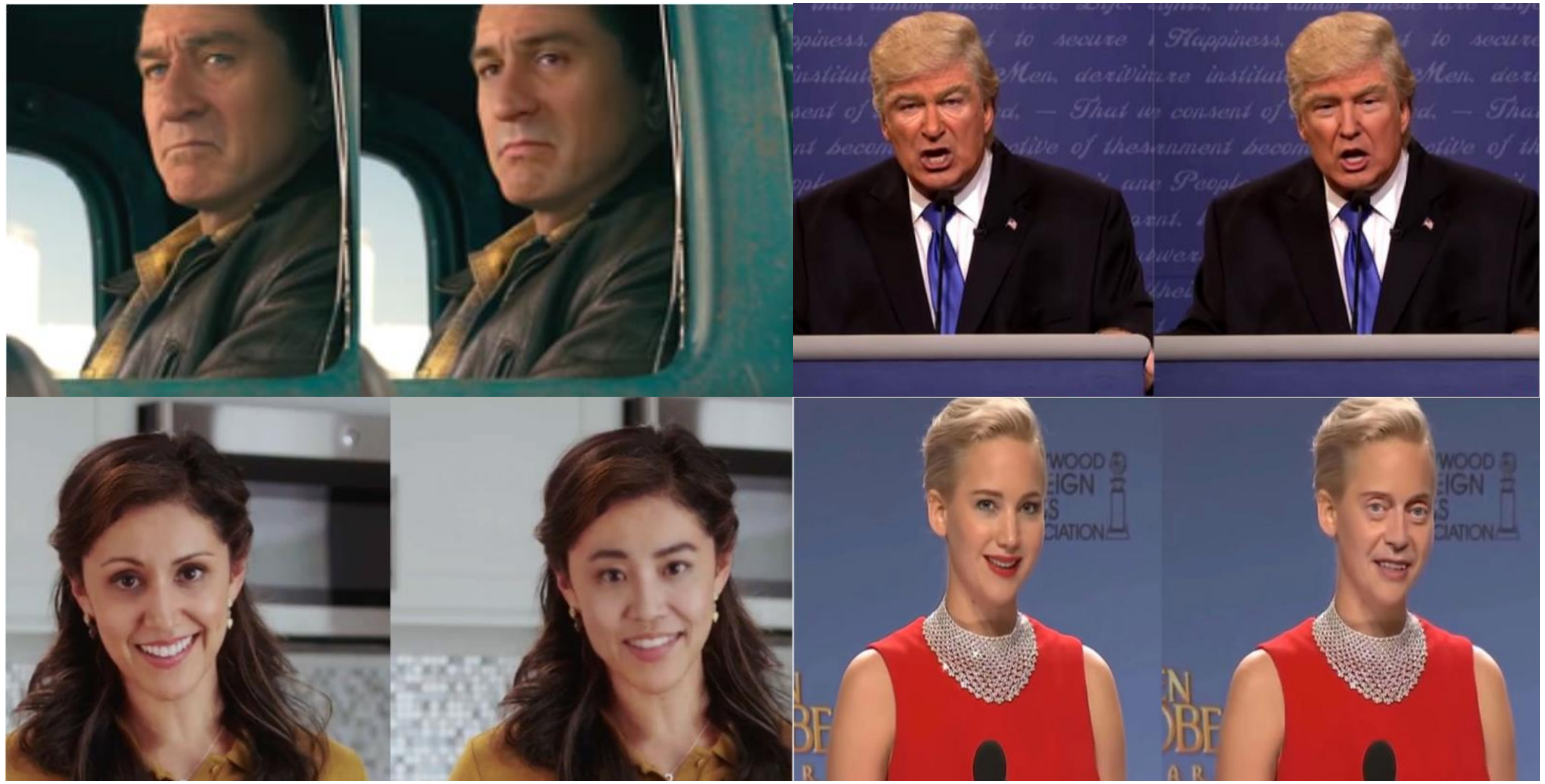
FakePolisher: Making DeepFakes More Detection-Evasive by Shallow Reconstruction

Yihao Huang[1], Felix Juefei-Xu[2], Run Wang[3], Qing Guo[3], Lei Ma[4], Xiaofei Xie[3], Jianwen Li[1], Weikai Miao[1], Yang Liu[3,5], Geguang Pu[1]

[1]East China Normal University, China [2]Alibaba Group, USA
[3]Nanyang Technological University,Singapore [4]Kyushu University, Japan
[5]Zhejiang University, China

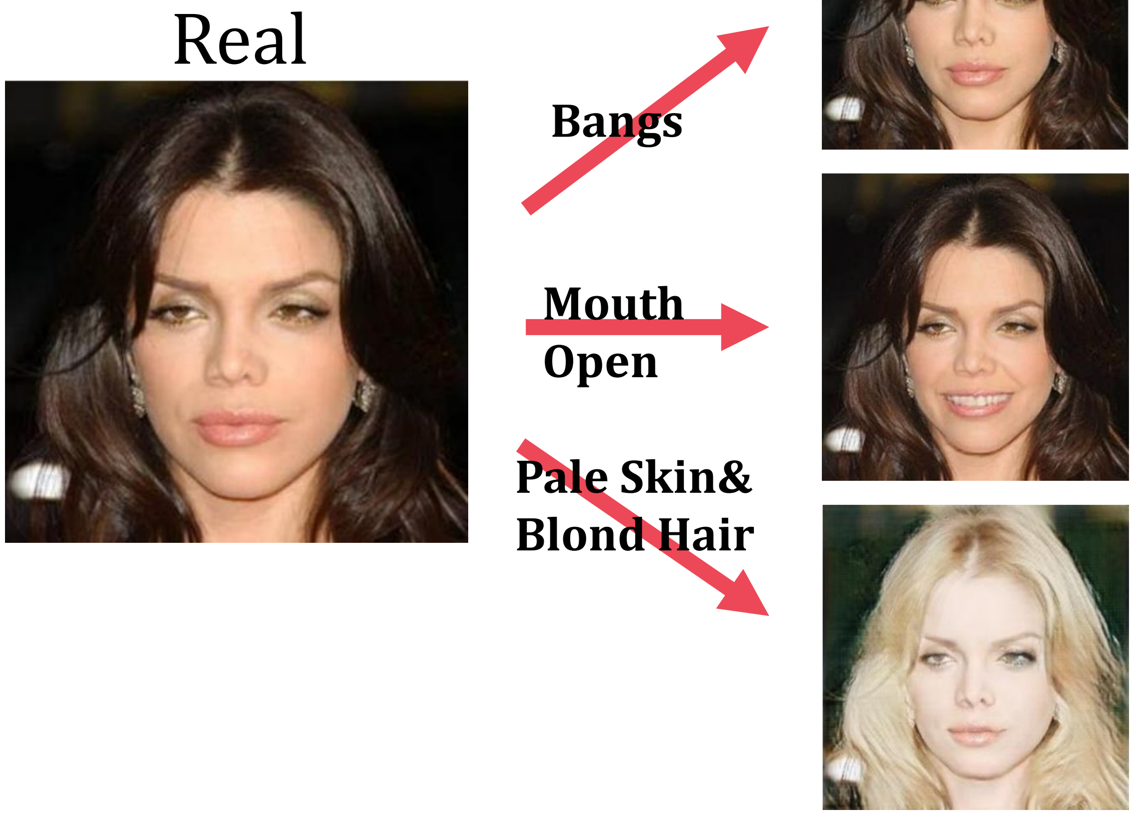
What is DeepFake?

Can you select out the real image?



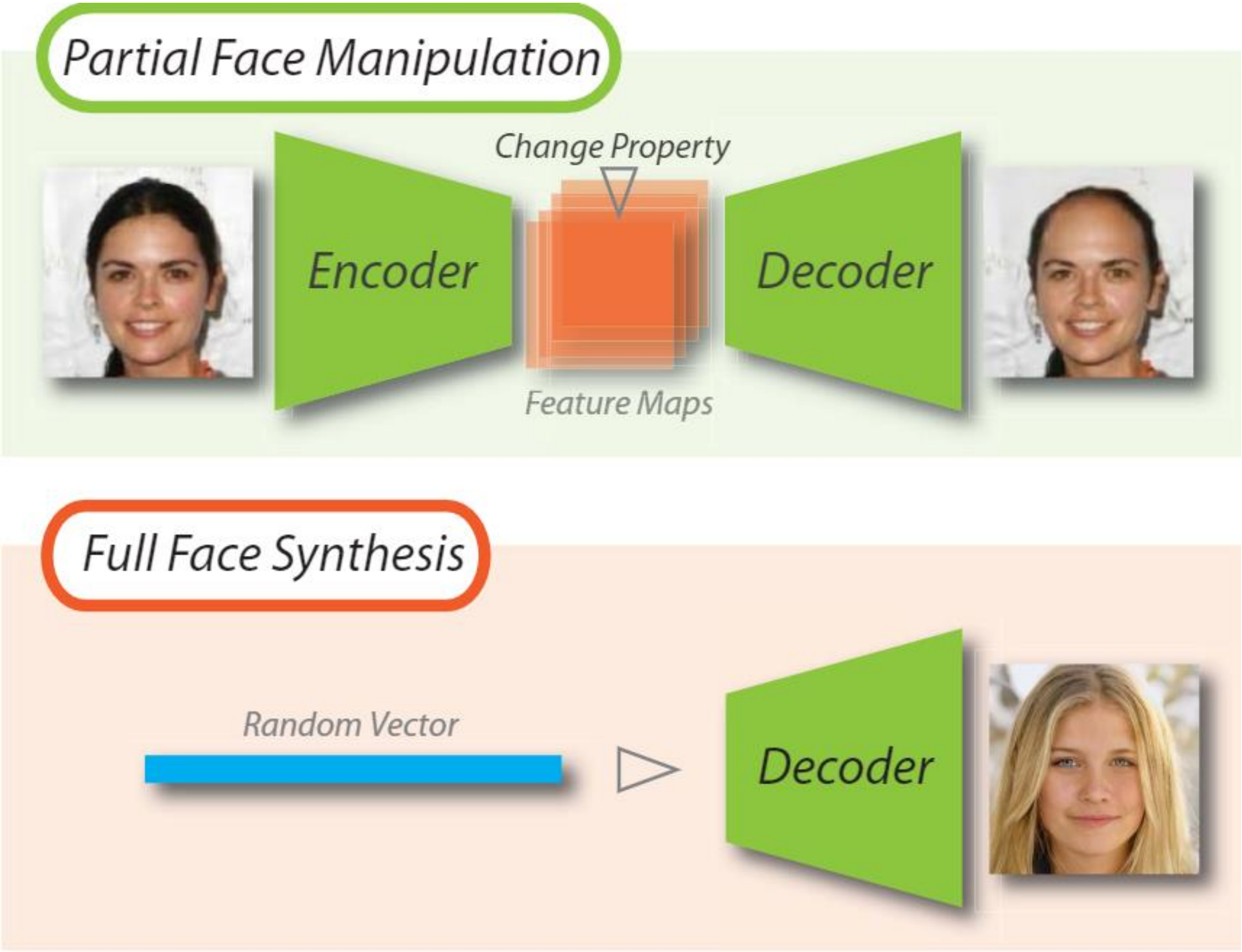
GAN-based image generation

Partial image manipulation



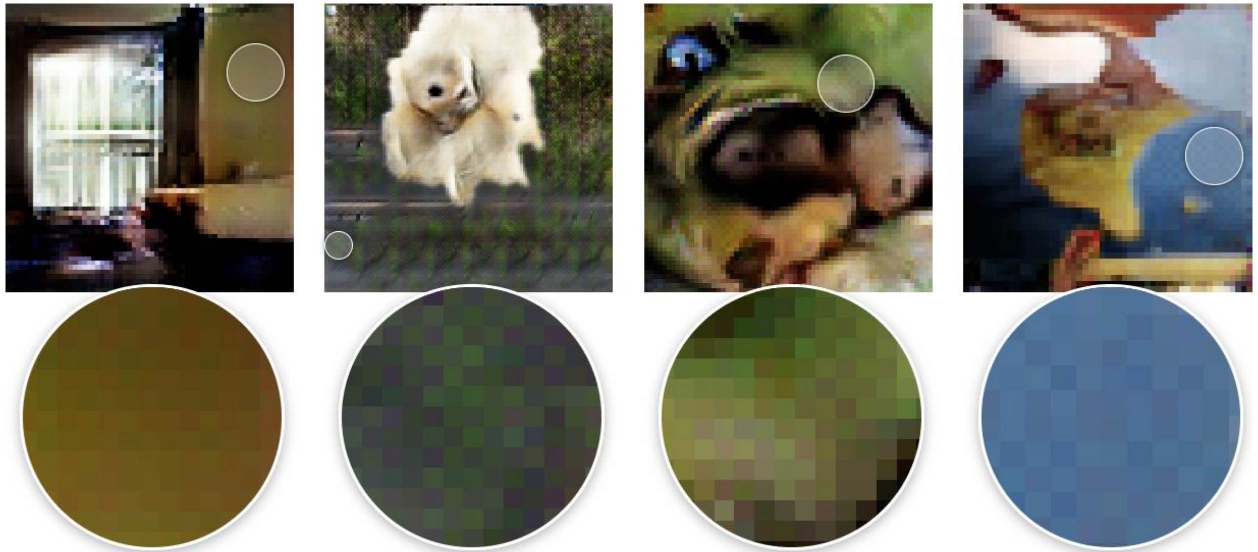
Full image synthesis

Architecture of GAN-based image generation

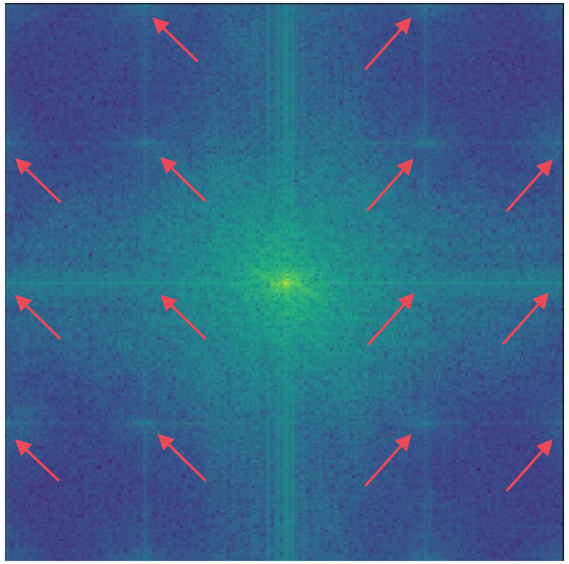


Artifact patterns

Upsampling introduce the artifact patterns !



Spatial domain

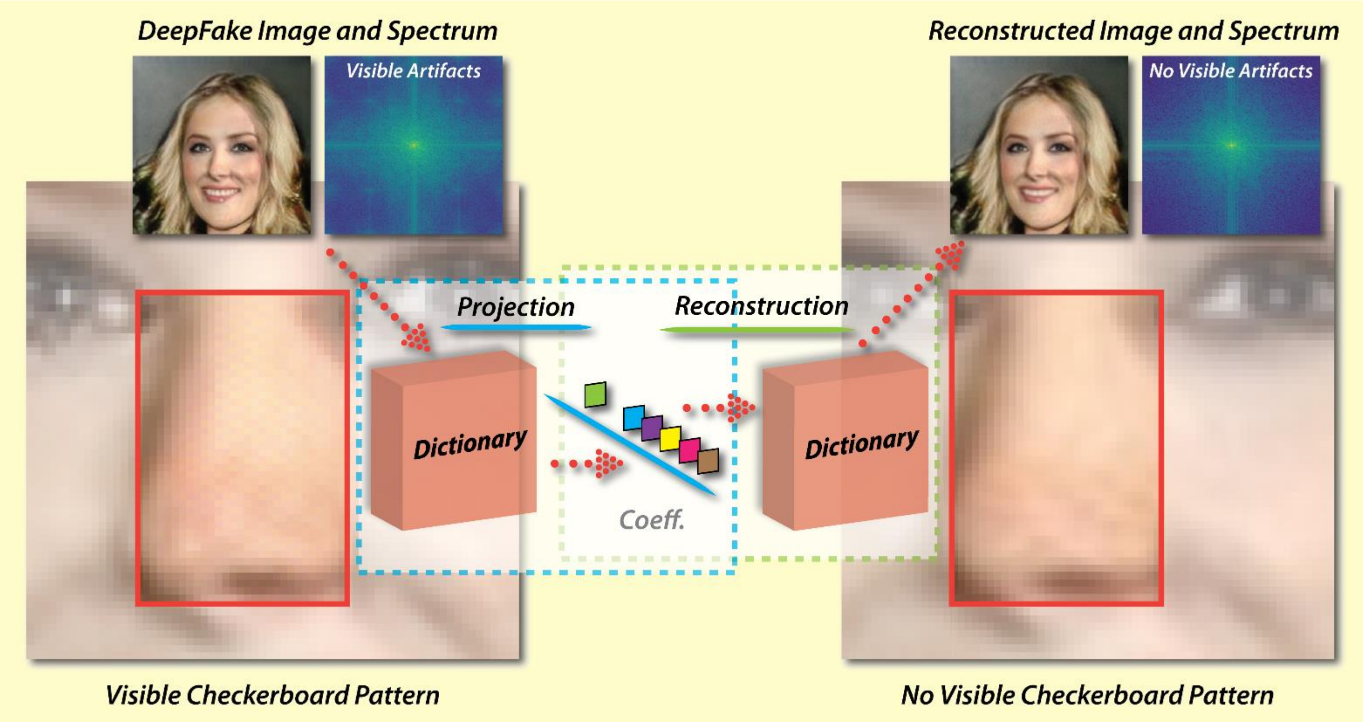


Frequency domain

FakePolisher

Contribution:

- **Dictionary learning-based** post-processing **shallow reconstruction** method
- **Does not** rely on **any information** of the GAN used for generation
- Fooling **three representative SOTA** fake image detection methods over fake images generated by **16** GAN-based methods.
- Indicating that existing methods can **highly leverage the manipulation of footprint** information from different perspectives.


DeepFake Image and Spectrum
Visible Artifacts
Reconstructed Image and Spectrum
No Visible Artifacts
Projection
Reconstruction
Dictionary
Coeff.
Dictionary
Visible Checkerboard Pattern
No Visible Checkerboard Pattern

Procedure of FakePolisher

There are three steps:

1. Train a dictionary model with a real image dataset
2. Seek the representation of a DeepFake image by linear projection or sparse coding depending on the over-completeness of the learned dictionary
3. Reconstruct the `fake-free' version of the DeepFake image by using the said dictionary

Global vs. Local Dictionary Learning

Global dictionary learning

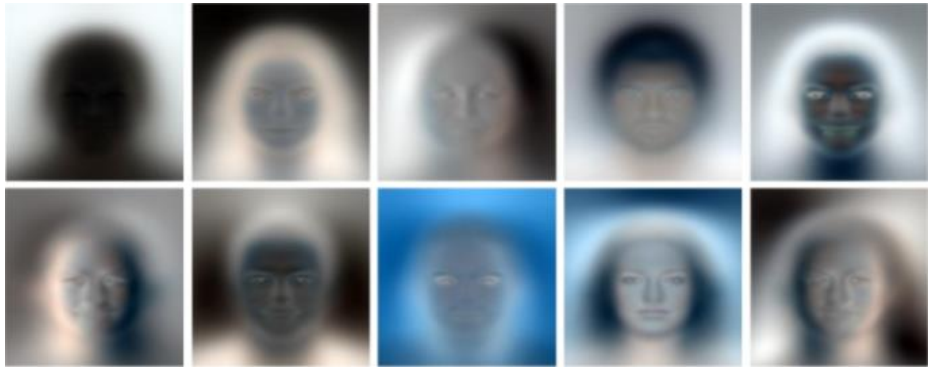
- Spans the entire image
- Suitable for face align images

Local dictionary learning

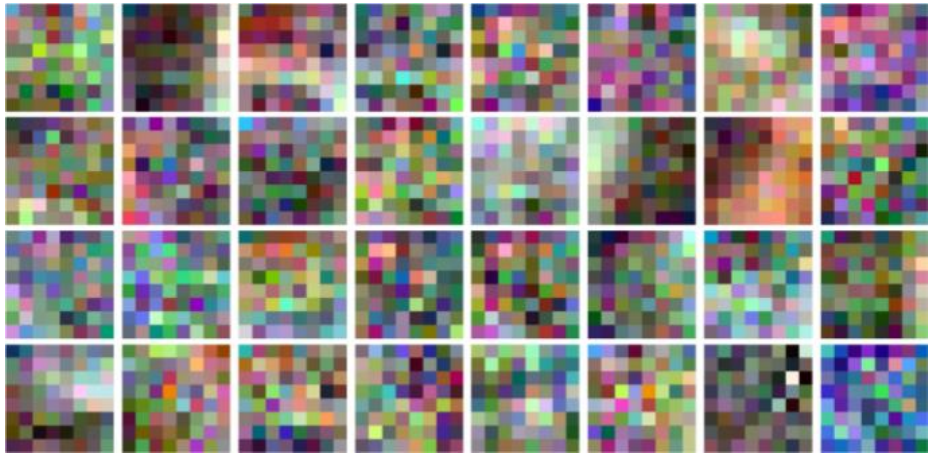
- Local patch-based
- Suitable for ImageNet images

Dictionary and reconstructed images

PCA

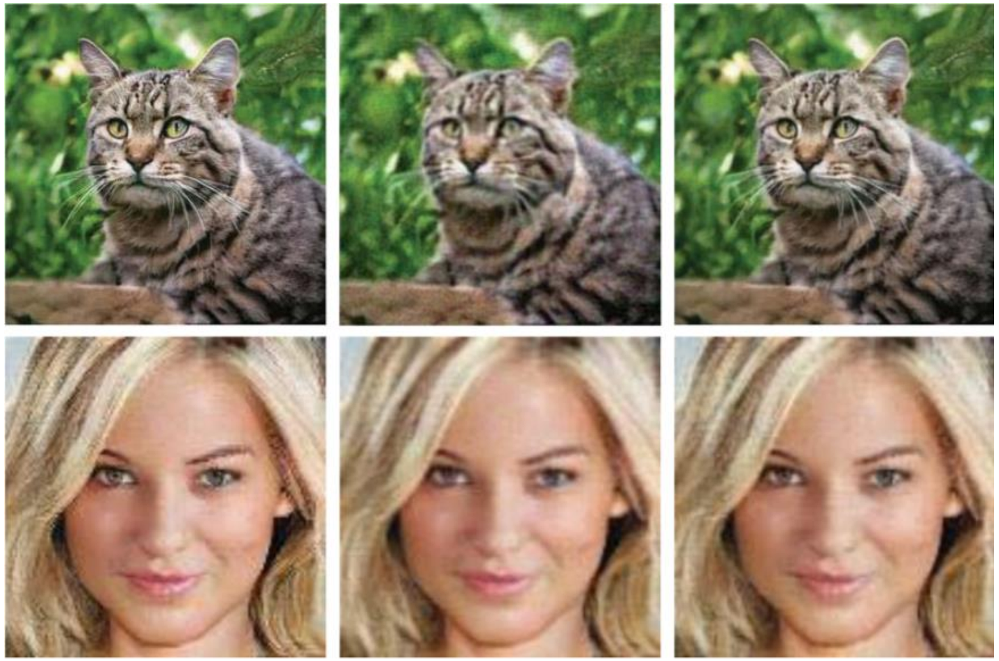


K-SVD



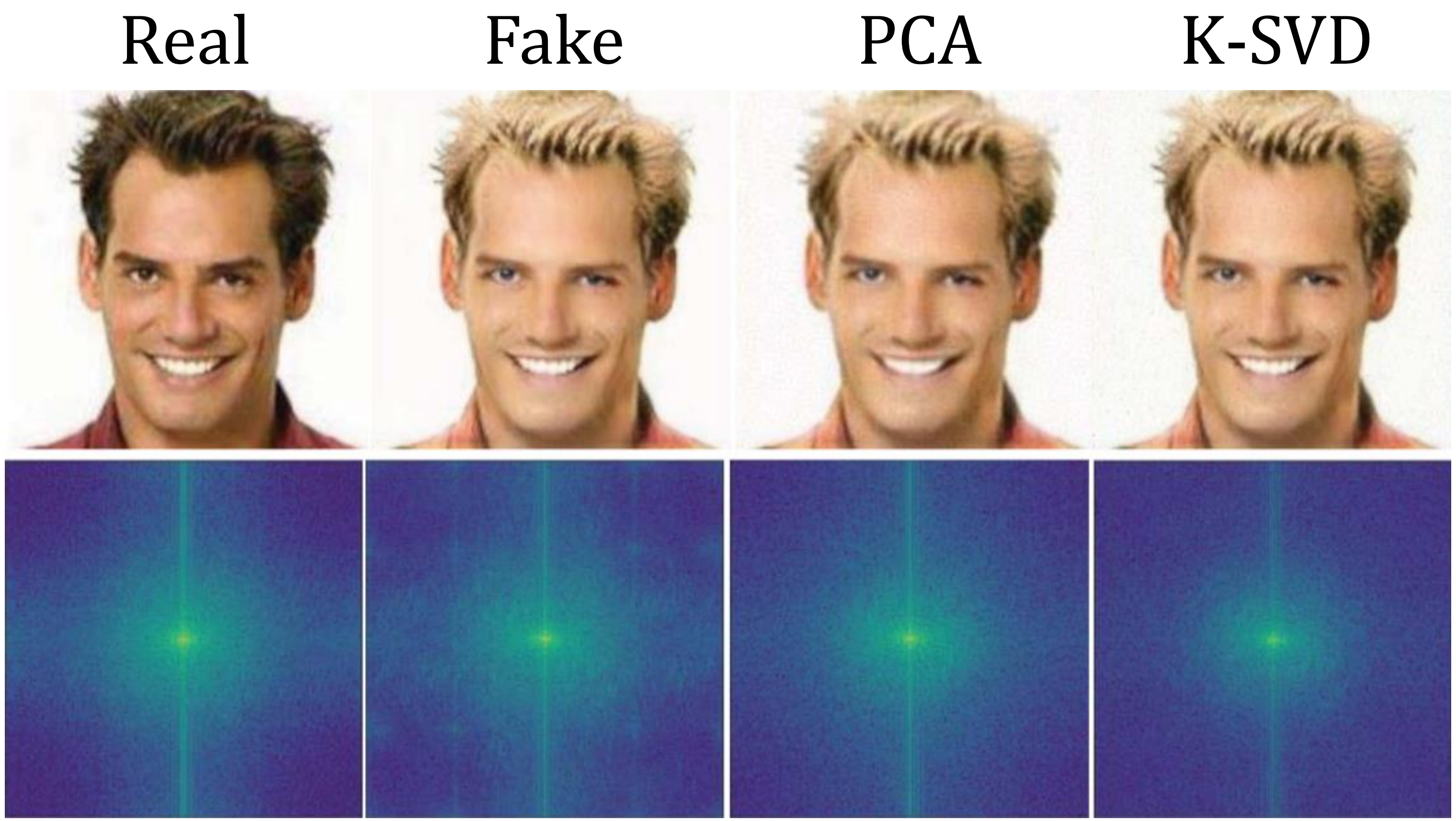
Reconstructed images

Fake PCA K-SVD



Comparison of spectrum

No artifact patterns in reconstructed images !



Experiment Setup

Dataset:

- CelebA, FFHQ, LSUN

Detectors:

- (fingerprint-based) GANFingerprint
- (spectrum-based) DCTA
- (image-based) CNNDetector

GAN-based image generation methods:

- ProGAN, SNGAN, CramerGAN, MMDGAN, StyleGAN, BigGAN, CycleGAN, StarGAN, GauGAN, CRN, IMLE, SITD, SAN, DeepFakes, StyleGAN2, whichfaceisreal

Experiment

Detector:

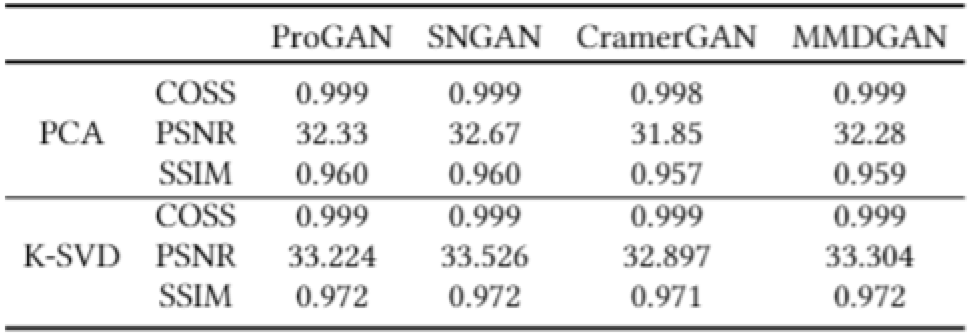
- GANFingerprint
  Accuracy reduction (PCA): on average 63.09%, worst case:92.92%
  Accuracy reduction (K-SVD): on average 45.39%, worst case:62.17%
- DCTA
  Accuracy reduction (PCA): on average 72.57%, worst case:72.57%
  Accuracy reduction (K-SVD): on average 68.55%, worst case:68.55%
- CNNDetector
  Accuracy reduction (PCA): on average 43.70%, worst case:93.30%
  Accuracy reduction (K-SVD): on average 19.40%, worst case:54.00%

Experiment

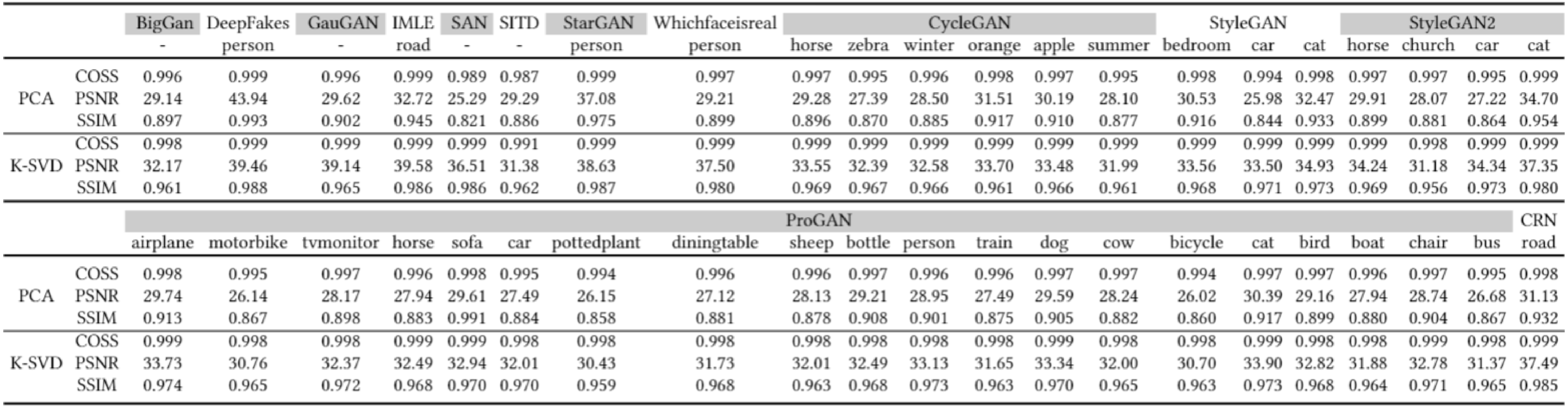
Similarity metrics:

- Cosine similarity (COSS)
- Peak signal-to-noise ratio (PSNR)
- Structural similarity (SSIM)

Experiment



		ProGAN	SNGAN	CramerGAN	MMDGAN
PCA	COSS	0.999	0.999	0.998	0.999
	PSNR	32.33	32.67	31.85	32.28
	SSIM	0.960	0.960	0.957	0.959
K-SVD	COSS	0.999	0.999	0.999	0.999
	PSNR	33.224	33.526	32.897	33.304
	SSIM	0.972	0.972	0.971	0.972



		BigGan	DeepFakes	GauGAN	IMLE	SAN	SITD	StarGAN	Whichfaceisreal	CycleGAN						StyleGAN			StyleGAN2			
		-	person	-	road	-	-	person	person	horse	zebra	winter	orange	apple	summer	bedroom	car	cat	horse	church	car	cat
PCA	COSS	0.996	0.999	0.996	0.999	0.989	0.987	0.999	0.997	0.997	0.995	0.996	0.998	0.997	0.995	0.998	0.994	0.998	0.997	0.997	0.995	0.999
	PSNR	29.14	43.94	29.62	32.72	25.29	29.29	37.08	29.21	29.28	27.39	28.50	31.51	30.19	28.10	30.53	25.98	32.47	29.91	28.07	27.22	34.70
	SSIM	0.897	0.993	0.902	0.945	0.821	0.886	0.975	0.899	0.896	0.870	0.885	0.917	0.910	0.877	0.916	0.844	0.933	0.899	0.881	0.864	0.954
K-SVD	COSS	0.998	0.999	0.999	0.999	0.999	0.991	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.999	0.998	0.999	0.999
	PSNR	32.17	39.46	39.14	39.58	36.51	31.38	38.63	37.50	33.55	32.39	32.58	33.70	33.48	31.99	33.56	33.50	34.93	34.24	31.18	34.34	37.35
	SSIM	0.961	0.988	0.965	0.986	0.986	0.962	0.987	0.980	0.969	0.967	0.966	0.961	0.966	0.961	0.968	0.971	0.973	0.969	0.956	0.973	0.980

		ProGAN																				CRN
		airplane	motorbike	tvmonitor	horse	sofa	car	pottedplant	diningtable	sheep	bottle	person	train	dog	cow	bicycle	cat	bird	boat	chair	bus	road
PCA	COSS	0.998	0.995	0.997	0.996	0.998	0.995	0.994	0.996	0.996	0.997	0.996	0.996	0.997	0.997	0.994	0.997	0.997	0.996	0.997	0.995	0.998
	PSNR	29.74	26.14	28.17	27.94	29.61	27.49	26.15	27.12	28.13	29.21	28.95	27.49	29.59	28.24	26.02	30.39	29.16	27.94	28.74	26.68	31.13
	SSIM	0.913	0.867	0.898	0.883	0.991	0.884	0.858	0.881	0.878	0.908	0.901	0.875	0.905	0.882	0.860	0.917	0.899	0.880	0.904	0.867	0.932
K-SVD	COSS	0.999	0.998	0.998	0.999	0.999	0.998	0.998	0.998	0.998	0.998	0.998	0.998	0.999	0.998	0.998	0.999	0.998	0.998	0.999	0.998	0.999
	PSNR	33.73	30.76	32.37	32.49	32.94	32.01	30.43	31.73	32.01	32.49	33.13	31.65	33.34	32.00	30.70	33.90	32.82	31.88	32.78	31.37	37.49
	SSIM	0.974	0.965	0.972	0.968	0.970	0.970	0.959	0.968	0.963	0.968	0.973	0.963	0.970	0.965	0.963	0.973	0.968	0.964	0.971	0.965	0.985

Thanks for listening !