Exploring Frequency Adversarial Attacks for Face Forgery Detection

Shuai Jia[1] Chao Ma[1]* Taiping Yao[2] Bangjie Yin[2] Shouhong Ding[2] Xiaokang Yang[1]

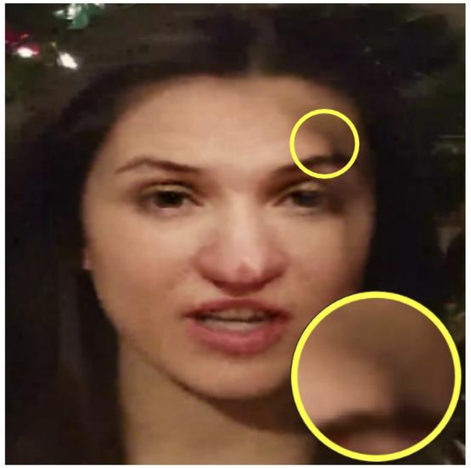
[1]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University

[2]Youtu Lab, Tencent

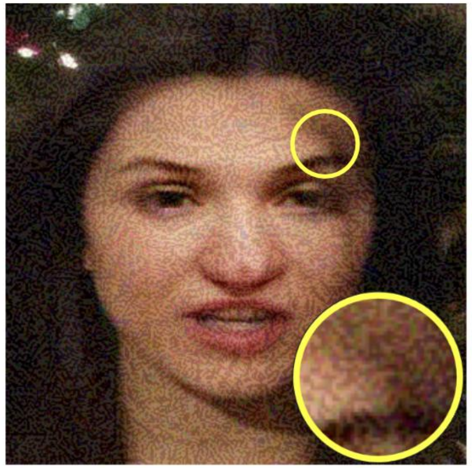
CVPR 2022

Motivation

- Various facial manipulation techniques have drawn serious public concerns in morality, security, and privacy.
- Existing face forgery classifiers are vulnerable to adversarial examples with injected imperceptible perturbations on the pixels.
- Many face forgery detectors always utilize the frequency diversity between real and fake faces as a crucial clue.

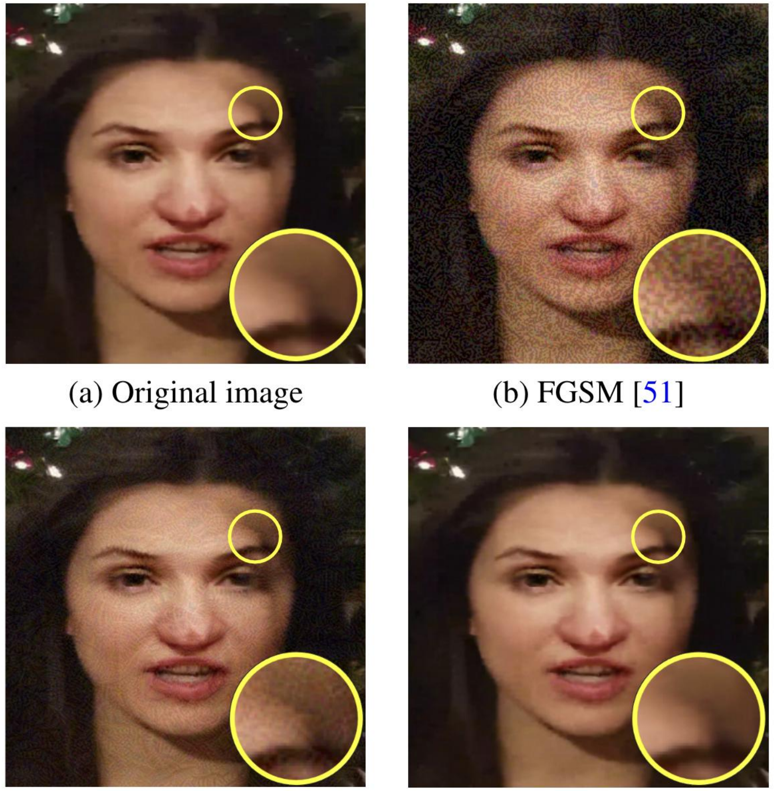


(a) Original image



(b) FGSM [51]

Motivation



(a) Original image

(b) FGSM [51]

(c) PGD [31]

(d) Ours

In this paper, instead of injecting adversarial perturbations into the spatial domain, we propose a frequency adversarial attack method against face forgery detectors.

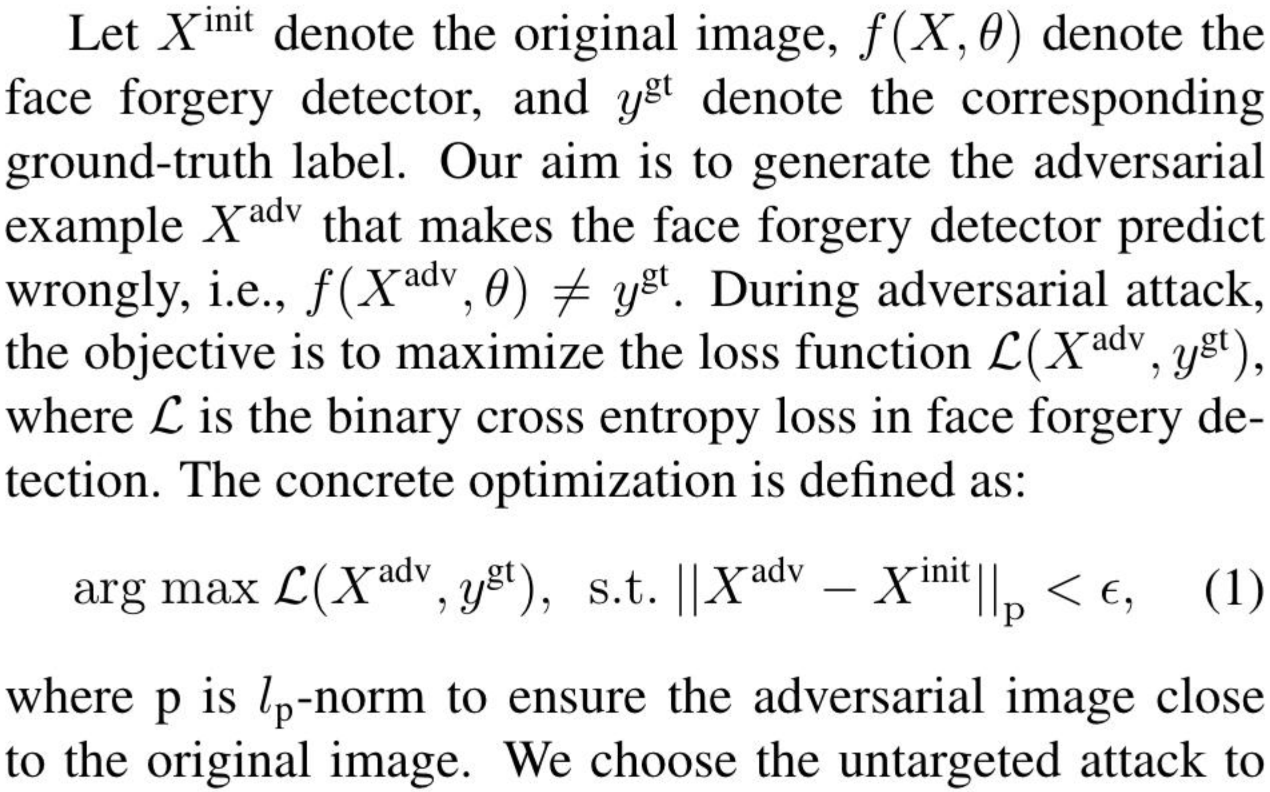
Contributions

(a) For the task of face forgery detection, we propose anovel adversarial attack method to generate perturbations in the frequency domain. Compared with the previous attacks, our method generates more imperceptible perturbations for human observers.

(b) To further boost the transferability of the attack, we propose a hybrid adversarial attack based on the strategy of meta-learning to simultaneously perform attacks on the spatial and frequency domain.

(c) We perform the proposed method both on the spatial-based face forgery detectors and the state-of-the-art frequency-based detectors. Extensive experiments on benchmarks demonstrate the effectiveness of our attack under both white-box and black-box settings.

Methodology



Let X^{init} denote the original image, $f(X, \theta)$ denote the face forgery detector, and y^{gt} denote the corresponding ground-truth label. Our aim is to generate the adversarial example X^{adv} that makes the face forgery detector predict wrongly, i.e., $f(X^{\text{adv}}, \theta) \neq y^{\text{gt}}$. During adversarial attack, the objective is to maximize the loss function $\mathcal{L}(X^{\text{adv}}, y^{\text{gt}})$, where $\mathcal{L}$ is the binary cross entropy loss in face forgery detection. The concrete optimization is defined as:

$$\arg\max \mathcal{L}(X^{\text{adv}}, y^{\text{gt}}), \quad \text{s.t. } ||X^{\text{adv}} - X^{\text{init}}||_{\text{p}} < \epsilon, \qquad (1)$$

where p is l_{p}-norm to ensure the adversarial image close to the original image. We choose the untargeted attack to

$$\begin{tikzcd}
X^{init} \arrow[r] \arrow[d] & \boxed{f(X,\theta)} \arrow[r, "\times" marking] & y^{gt} \\
X^{adv} \arrow[ur] & &
\end{tikzcd}$$

Spatial Adversarial Attack

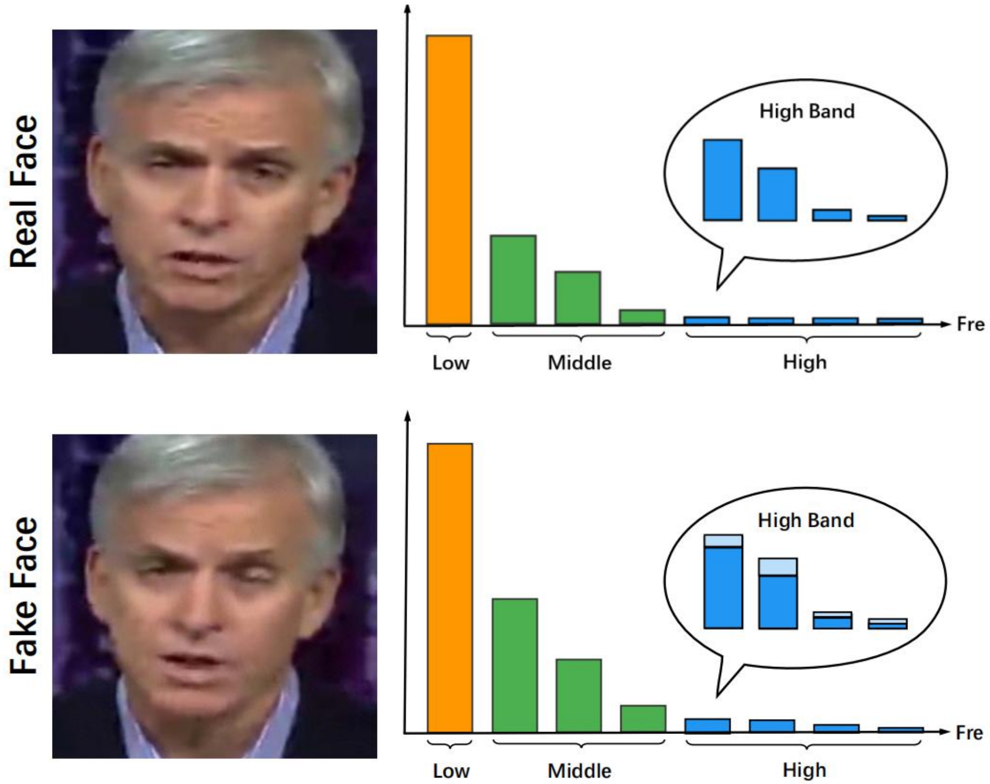
- Fast Gradient Sign Method (FGSM). FGSM is a single-step attack method that calculates the perturbations based on the gradient of the adversarial loss. The optimization is defined as:

$$X^{\mathrm{adv}} = X^{\mathrm{init}} + \epsilon \cdot \mathrm{sign}(\nabla_X \mathcal{L}(X^{\mathrm{adv}}, y^{\mathrm{gt}})). \qquad (2)$$

- Projected Gradient Descent (PGD). PGD is a multi-step variant of FGSM. Meanwhile, it adopts a random initialization of perturbations at the first step. The update procedure is defined as:

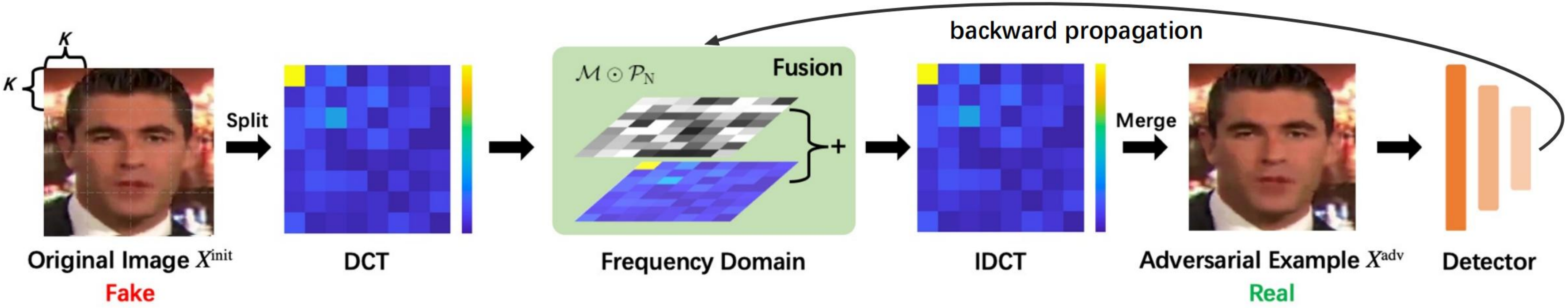
$$\begin{aligned} X_0^{\mathrm{adv}} &= X^{\mathrm{init}}, \\ X_{\mathrm{n+1}}^{\mathrm{adv}} &= \mathrm{Clip}\left\{X_{\mathrm{n}}^{\mathrm{adv}} + \alpha \cdot \mathrm{sign}(\nabla_X \mathcal{L}(X_{\mathrm{n}}^{\mathrm{adv}}, y^{\mathrm{gt}}))\right\}. \end{aligned} \qquad (3)$$

Frequency Adversarial Attack



(a) Image (b) Frequency distribution

Frequency Adversarial Attack

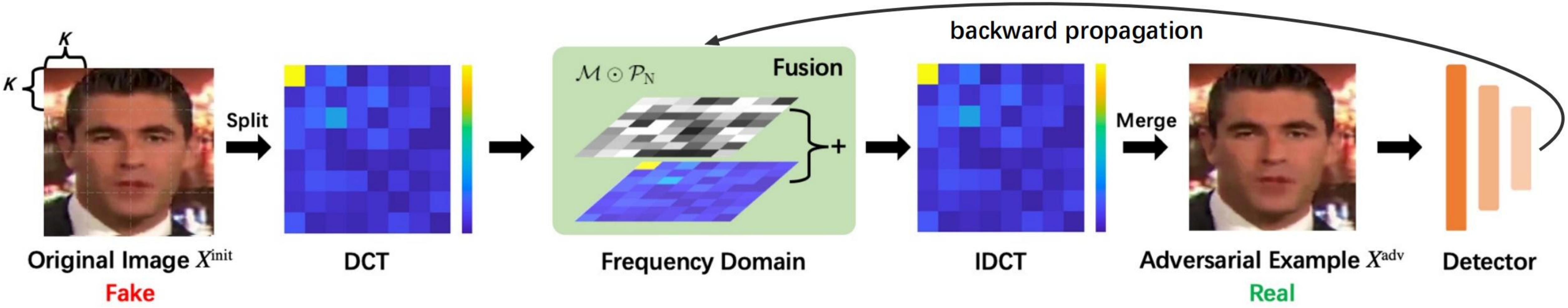

backward propagation
K
K
Split
$\mathcal{M} \odot \mathcal{P}_N$
Fusion
+
Merge
Original Image X^{init}
Fake
DCT
Frequency Domain
IDCT
Adversarial Example X^{adv}
Real
Detector

We summarize the optimization procedure as follows:

$$\begin{aligned} &\arg\ \max\ \mathcal{L}(\mathcal{D}'(\mathcal{F}(\mathcal{D}(X^{\text{adv}}))), \theta, y^{\text{gt}}), \\ &\text{s.t.}\ ||\mathcal{D}(\mathrm{X}^{\text{adv}}) - \mathcal{D}(\mathrm{X}^{\text{init}})||_{\mathrm{p}} < \epsilon, \end{aligned} \tag{4}$$

where $\mathcal{D}(\cdot)$ denotes discrete cosine transform (DCT), $\mathcal{D}'(\cdot)$ denotes inverse discrete cosine transform (IDCT), $\mathcal{F}$ **represents the fusion module to modify the energy in the frequency domain.**

Frequency Adversarial Attack


backward propagation
K
K
Split
$\mathcal{M} \odot \mathcal{P}_{\mathrm{N}}$
Fusion
+
Merge
Original Image X^{init}
Fake
DCT
Frequency Domain
IDCT
Adversarial Example X^{adv}
Real
Detector

The complete fusion module is defined as:

$$\mathcal{F}(X_{\mathrm{n}}^{\mathrm{adv}}) = \mathcal{D}(X_{\mathrm{n}}^{\mathrm{adv}}) + \mathcal{M} \odot \mathcal{P}_{\mathrm{n+1}}, \qquad (6)$$

where $\odot$ is Hadamard product. During the optimization, $\mathcal{P}_{\mathrm{n+1}}$ is updated as follows:

$$\mathcal{P}_{\mathrm{n+1}} = \mathcal{P}_{\mathrm{n}} + \lambda \cdot \mathrm{sign}(\nabla_{\mathcal{P}} \mathcal{L}(\mathcal{D}'(\mathcal{F}(\mathcal{D}(X_{\mathrm{n}}^{\mathrm{adv}}))), \theta, y^{\mathrm{gt}})), \quad (7)$$

where λ is the step size in each iteration.

Hybrid Adversarial Attack

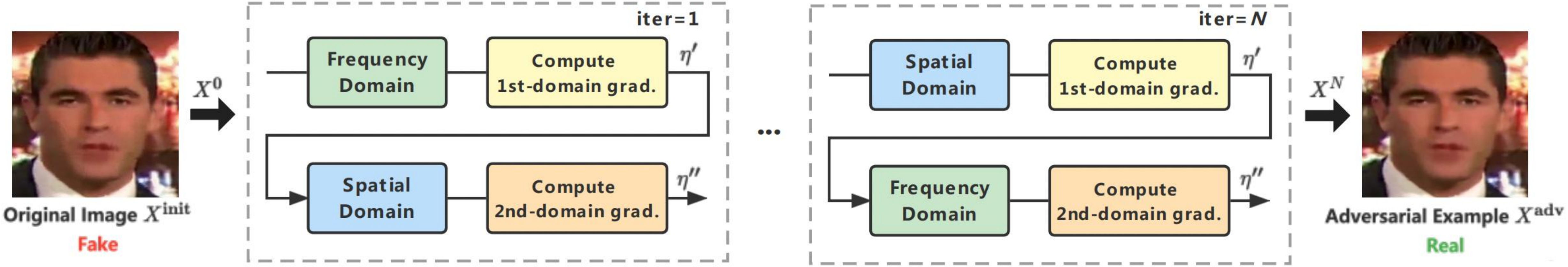


Figure 4. The procedure of hybrid adversarial attack. To combine the adversarial attack in different domains, we calculate gradients from both domains in order and update the perturbations. Then, we switch the order of domains in the next step. After iterations, the adversarial example gathers the gradients from both domains, leading to a stronger adversarial attack on both white-box and black-box settings.

Experiments

Datasets: DFDC and FaceForensics++

Metric: Attack Success Rate

Forgery Models: Spatial-based: EfficientNet b4, ResNet 50, and XceptionNet;

Frequency-based: F3-Net and LRL.

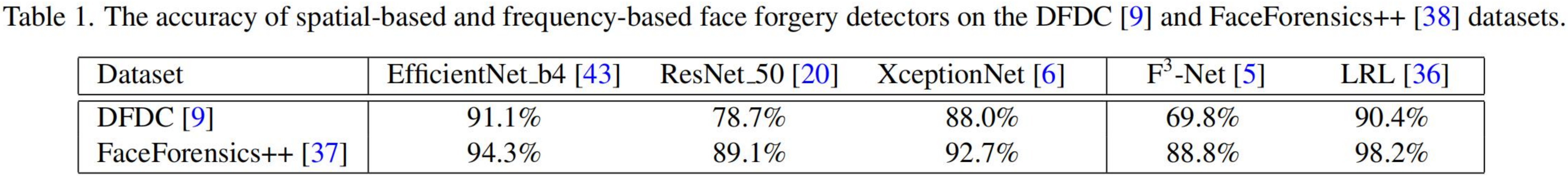


Table 1. The accuracy of spatial-based and frequency-based face forgery detectors on the DFDC [9] and FaceForensics++ [38] datasets.

Dataset	EfficientNet_b4 [43]	ResNet_50 [20]	XceptionNet [6]	F^3-Net [5]	LRL [36]
DFDC [9]	91.1%	78.7%	88.0%	69.8%	90.4%
FaceForensics++ [37]	94.3%	89.1%	92.7%	88.8%	98.2%

Attack on Spatial-based Models

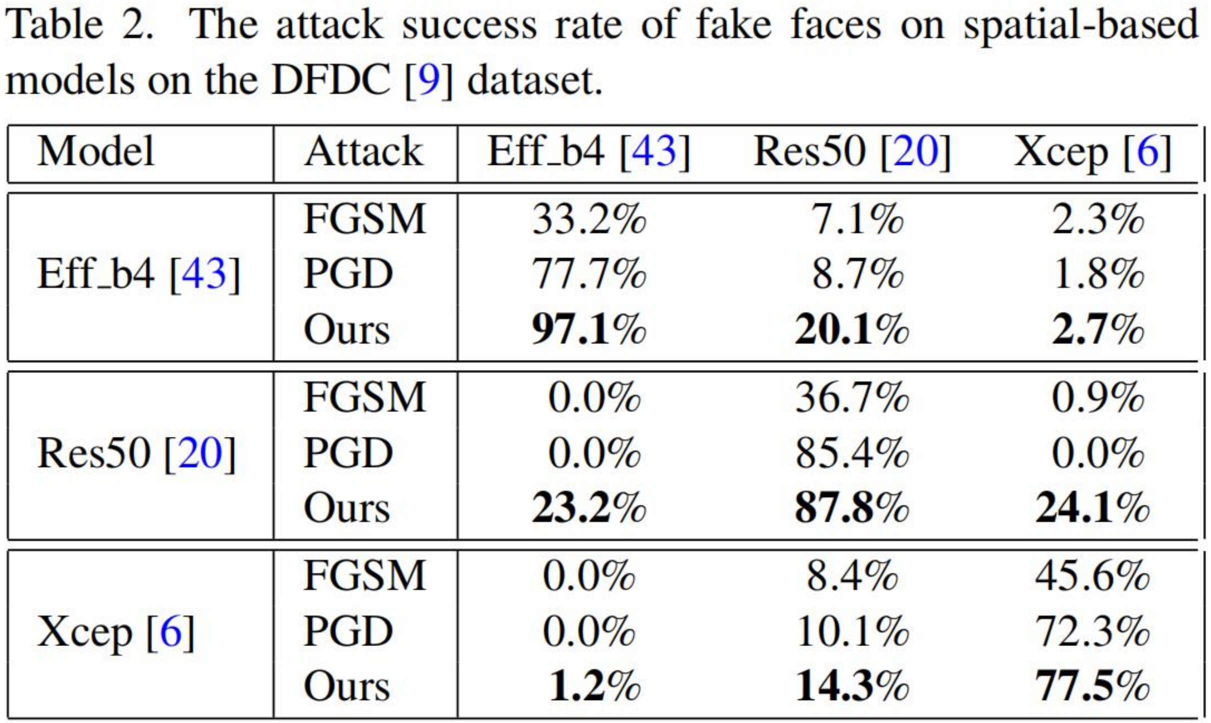


Table 2. The attack success rate of fake faces on spatial-based models on the DFDC [9] dataset.

Model	Attack	Eff_b4 [43]	Res50 [20]	Xcep [6]
Eff_b4 [43]	FGSM	33.2%	7.1%	2.3%
	PGD	77.7%	8.7%	1.8%
	Ours	**97.1%**	**20.1%**	**2.7%**
Res50 [20]	FGSM	0.0%	36.7%	0.9%
	PGD	0.0%	85.4%	0.0%
	Ours	**23.2%**	**87.8%**	**24.1%**
Xcep [6]	FGSM	0.0%	8.4%	45.6%
	PGD	0.0%	10.1%	72.3%
	Ours	**1.2%**	**14.3%**	**77.5%**

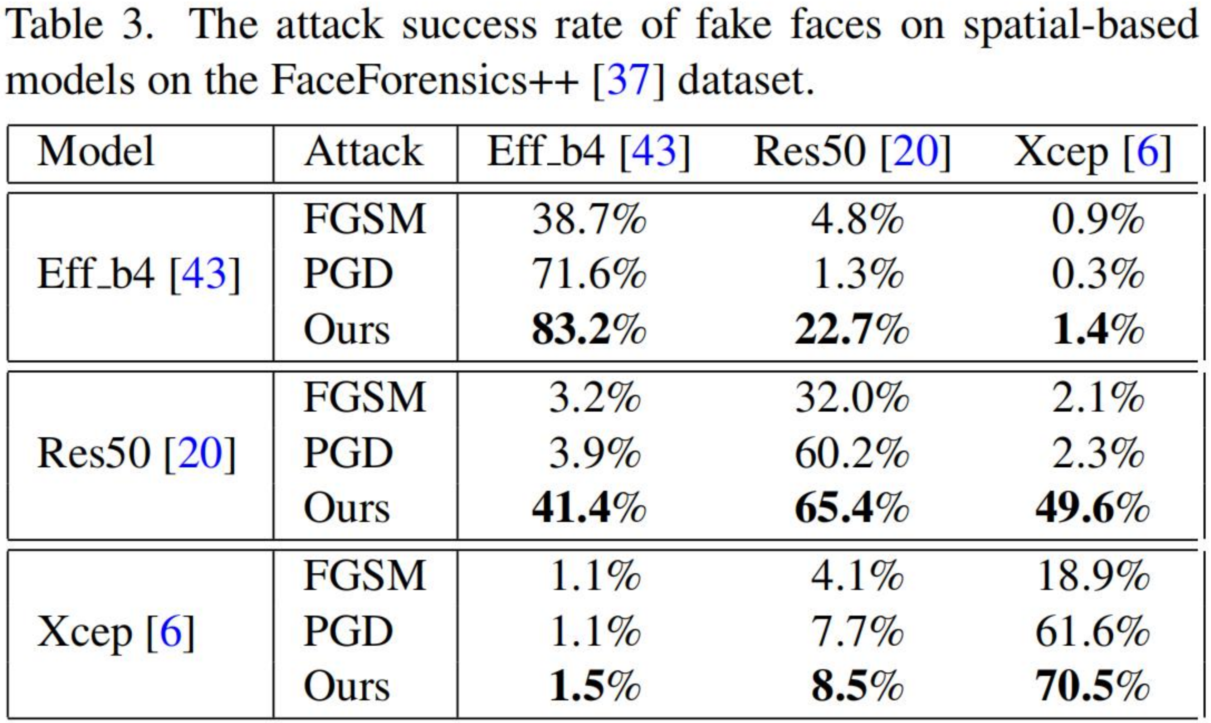
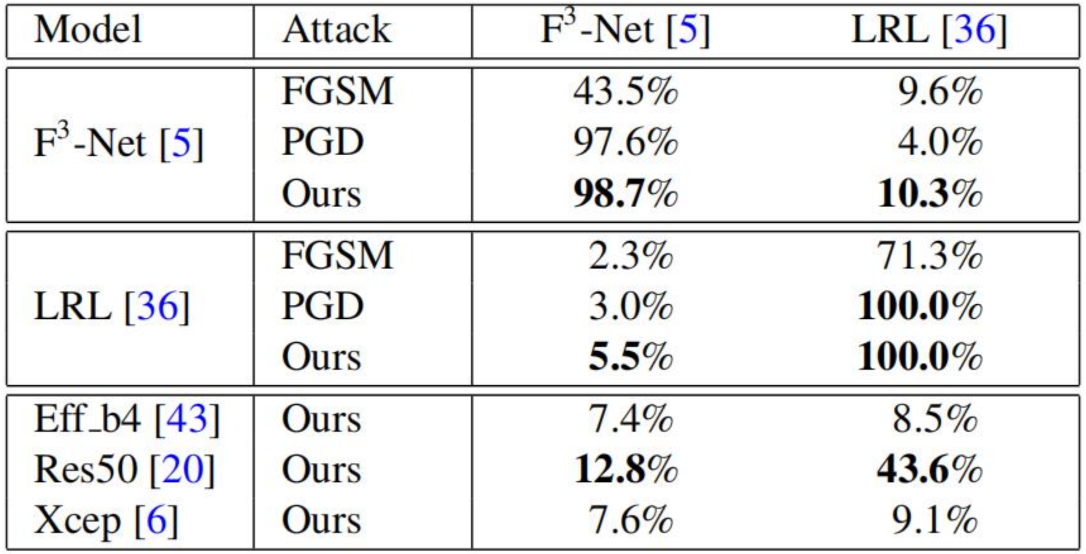


Table 3. The attack success rate of fake faces on spatial-based models on the FaceForensics++ [37] dataset.

Model	Attack	Eff_b4 [43]	Res50 [20]	Xcep [6]
Eff_b4 [43]	FGSM	38.7%	4.8%	0.9%
	PGD	71.6%	1.3%	0.3%
	Ours	**83.2%**	**22.7%**	**1.4%**
Res50 [20]	FGSM	3.2%	32.0%	2.1%
	PGD	3.9%	60.2%	2.3%
	Ours	**41.4%**	**65.4%**	**49.6%**
Xcep [6]	FGSM	1.1%	4.1%	18.9%
	PGD	1.1%	7.7%	61.6%
	Ours	**1.5%**	**8.5%**	**70.5%**

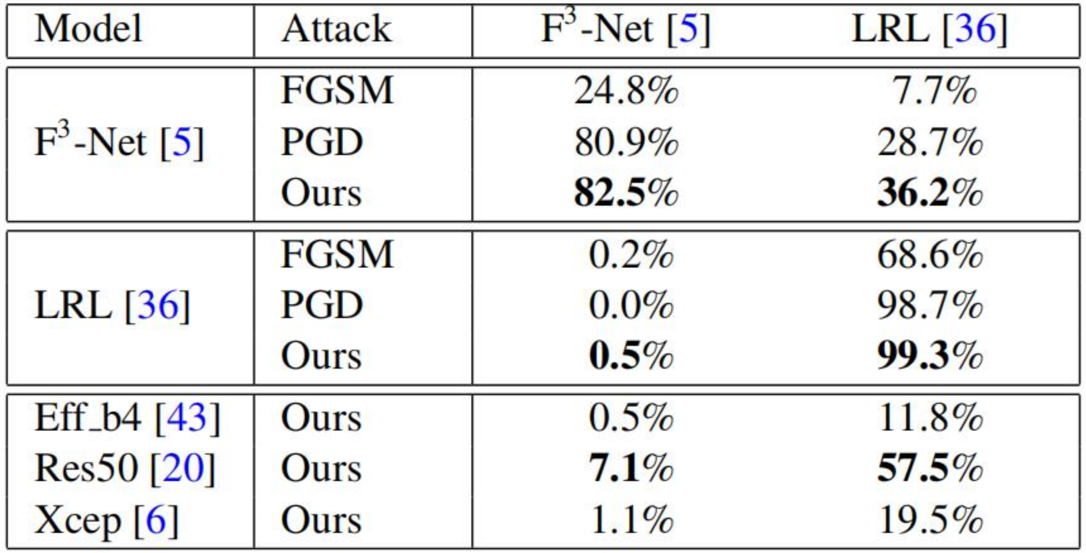
Attack on Frequency-based Models

Table 6. The attack success rate of fake faces on frequency-based models on the DFDC [9] dataset.



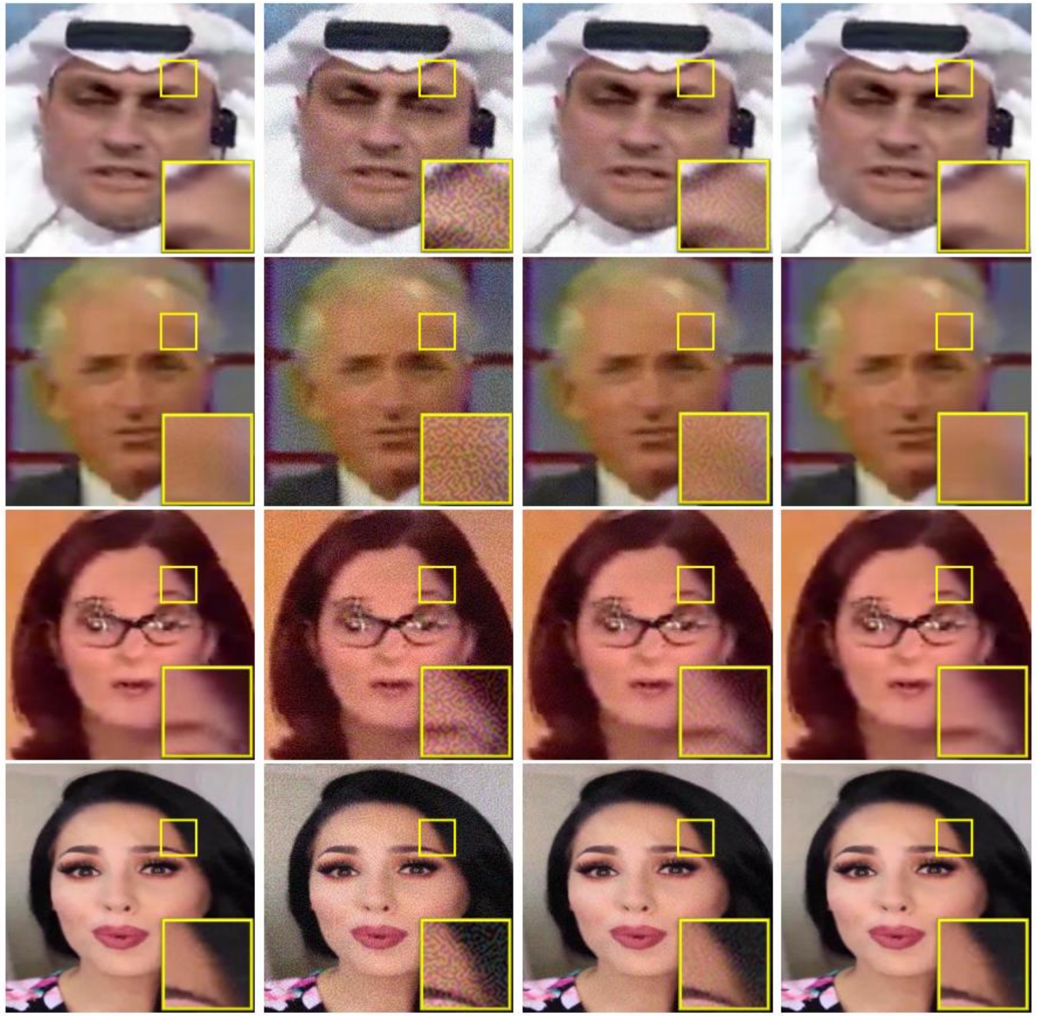
Model	Attack	F^3-Net [5]	LRL [36]
F^3-Net [5]	FGSM	43.5%	9.6%
	PGD	97.6%	4.0%
	Ours	**98.7%**	**10.3%**
LRL [36]	FGSM	2.3%	71.3%
	PGD	3.0%	**100.0%**
	Ours	**5.5%**	**100.0%**
Eff_b4 [43]	Ours	7.4%	8.5%
Res50 [20]	Ours	**12.8%**	**43.6%**
Xcep [6]	Ours	7.6%	9.1%

Table 7. The attack success rate of fake faces on frequency-based models on the FaceForensics++ [37] dataset.



Model	Attack	F^3-Net [5]	LRL [36]
F^3-Net [5]	FGSM	24.8%	7.7%
	PGD	80.9%	28.7%
	Ours	**82.5%**	**36.2%**
LRL [36]	FGSM	0.2%	68.6%
	PGD	0.0%	98.7%
	Ours	**0.5%**	**99.3%**
Eff_b4 [43]	Ours	0.5%	11.8%
Res50 [20]	Ours	**7.1%**	**57.5%**
Xcep [6]	Ours	1.1%	19.5%

Image Quality Assessment



(a) Original (b) FGSM [15] (c) PGD [31] (d) Ours

Table 10. Quantitative evaluation of adversarial examples generated by FGSM [15], PGD [31] and our method on the FaceForensics++ [37] dataset.

Attack method	MSE (↓)	PSNR (↑)	SSIM (↑)
FGSM	0.0279	23.3	0.0881
FGD	0.0238	30.4	0.1343
Ours	**0.0027**	**42.7**	**0.1763**

Q&A

Thank you!