

2025.04.07

Can Simple Averaging Defeat Modern Watermarks?

국민대학교 이재형

jaehyeong8121@gmail.com

Contents

01

Introduction

02

What is Steganalysis?

03

Results

04

Future Directions

Introduction

“Can simple Averaging Defeat Modern Watermarks?”

위 논문은
content-agnostic 워터마킹 기법이 가진 **steaganalysis** 공격 취약성을 밝히고



단순한 평균 연산만으로도 워터마크 제거 및 위조가 가능함을 보이며
이를 통해 향후 워터마킹 기법에 대한 새로운 보안 지침을 제안

01 Introduction

1.1 Two types of watermark

논문에서 워터마크를 **이미지의 content를 고려하느냐** 여부에 따라
2가지 유형으로 구분한다



Content-adaptive

- **이미지의 특성**(e.g. 색상, 텍스처)을 고려해서 워터마크를 삽입
- 왜곡과 공격에 더 강건함
- e.g. HiDDeN, RivaGAN

Content-agnostic

- 이미지 내용과 무관하게 고정되고, **미리 정의된 워터마크 패턴을 삽입**
- 계산하기 쉽고 구현하기 쉽다
- e.g. DwtDctSvd, Tree-Ring

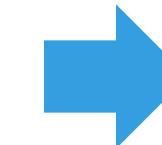
01 Introduction

1.2 Difference from previous studies

[기존 연구]

워터마크 강건성을 향상시키기 위해

- 워터마크의 기술적 설계를 고려
 - 워터마크 삽입 방식의 정교한 설계
함으로써 강건성 확보
- 훈련 중 데이터 증강
 - JPEG 압축, 노이즈, 교란과 같은 다양한 왜곡에 대한 견고성 입증



- Tree-Ring을 포함한 **content-agnostic** 워터마킹 기법이 **steganalysis**에 취약하

다는 사실을 처음으로 밝힘

- 특히 Tree-Ring 워터마크에서는 **내용과 무관한 ripple 패턴**이 **워터마크 검출의 핵심**임을 찾아냄



“Diffusion 기반 워터마킹 기법들이 과연 정말 정보 전달 목적의 워터마크를 넣고 있는가?”

OR

“**그저 저수준의 시각적 패턴만 반복하는 것인가?**”에 대해 기술적, 보안적 비판을 제기

01 Introduction

1.3 Main **contribution** of this paper

- 1 스테가분석(steganalysis)의 제거와 위치에 대한 content-agnostic 워터마크의 취약성을 밝힘
- 2 Blackbox 설정에서 Tree-Ring을 성공적으로 공격한 최초의 사례로,
Diffusion 노이즈 워터마킹 기법의 본질에 대한 더 깊은 통찰력을 제공함
- 3 스테가분석의 공격으로부터 방어하기 위해 미래 워터마킹 기법에 대한 새로운 보안 지침을 제안함

What is Steganalysis?

Blackbox Steganalysis 기반 공격

Blackbox 공격

- 공격자가 워터마킹 알고리즘의 내부 구조나 파라미터를 전혀 모르는 상태에서 진행하는 공격
- 즉, “몰라도 워터마크를 깰 수 있다!”는 의미

스테가분석 공격 특징

- 이미지 여러 개를 평균내서 워터마크 패턴을 추출함
- 사전 지식 없이도 워터마크 제거 or 위조 가능
- 고도화된 딥러닝 기반 공격 없이도 단순한 평균 연산으로 추출 가능

02 What is Steganalysis?

2.1 Notations used in this paper

기호	설명	[Flow 시각화]
x_{\emptyset}	원본 디지털 이미지 (original digital image)	[Original Image x_{\emptyset}] + [Watermark w] ↓ $E(x_{\emptyset}, w)$
w	워터마크 정보 (bit sequence or geometric pattern 등)	[Watermarked Image x_w] ↓ $D(x_w)$
E	워터마크 인코더 함수 : x_{\emptyset} 와 w 를 입력받아 워터마크 이미지 생성	[Recovered Watermark \hat{w}]
x_w	워터마크 삽입 이미지 : $x_w = E(x_{\emptyset}, w)$	
D	워터마크 디코더 함수 : $D(x_w) = \hat{w}$	

02 What is Steganalysis?

2.2 Threat model

적대자(Adversary)의 목표 : D를 속인다, T(.)라는 전략을 통해서

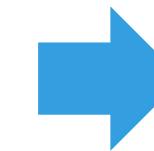
1. 워터마크 제거(removal)

$$\max_T \|D(T(x_w)) - w\|,$$

- 워터마크가 있는 이미지 x_w 을 공격해서 변형 $T(x_w)$ 를 만든 뒤
- 디코더 D가 워터마크 w 를 제대로 추출하지 못하게 함

[전제 조건]

T(x)는 원본 이미지 x와 **시각적으로 구분이 안되어야 함**



2. 워터마크 위조(forgery)

$$\min_T \|D(T(x_\emptyset)) - w\|,$$

- 워터마크가 없는 이미지 x_\emptyset 을 변경해서 $T(x_\emptyset)$ 를 만든 뒤
- 디코더가 w 라고 착각하도록 만듬

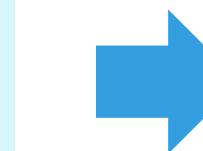
T를 강한 왜곡이 아닌 **스테가분석 방식으로 설계**

02 What is Steganalysis?

2.3 Steganalysis : watermark extraction, removal & forgery

기본 가정 : $x_w = x_\emptyset + \delta_w$, 워터마크가 단순히 이미지에 더해진 일종의 덧셈 패턴이다

$$\hat{\delta}_w = \frac{1}{n} \left(\sum_{i=1}^n x_{w,i} - \sum_{i=1}^n x_{\emptyset,i} \right)$$



워터마크 제거 $\hat{x}_\emptyset = T(x_w) = x_w - \hat{\delta}_w$
워터마크 위조 $\hat{x}_w = T(x_\emptyset) = x_\emptyset + \hat{\delta}_w$

(δ_w : content agnostic한 워터마크 패턴)

✓ 평균 기반 패턴 추출

- 기본 가정에 따라 원본 이미지가 존재한다면 단순 뺄셈만으로 워터마크 패턴 추출 가능
- 하지만, 실제로는 개별 이미지마다 노이즈나 변형이 있어 정확한 패턴을 구하기 어려움
- **여러 이미지를 평균을 내서** 안정적으로 패턴을 추출함
- 이 방식은 두 가지로 나뉨:
 - Greybox Setting : 워터마크 삽입 이미지와 원본 이미지의 쌍이 존재
 - Blackbox Setting : 원본 이미지를 모르는 상황

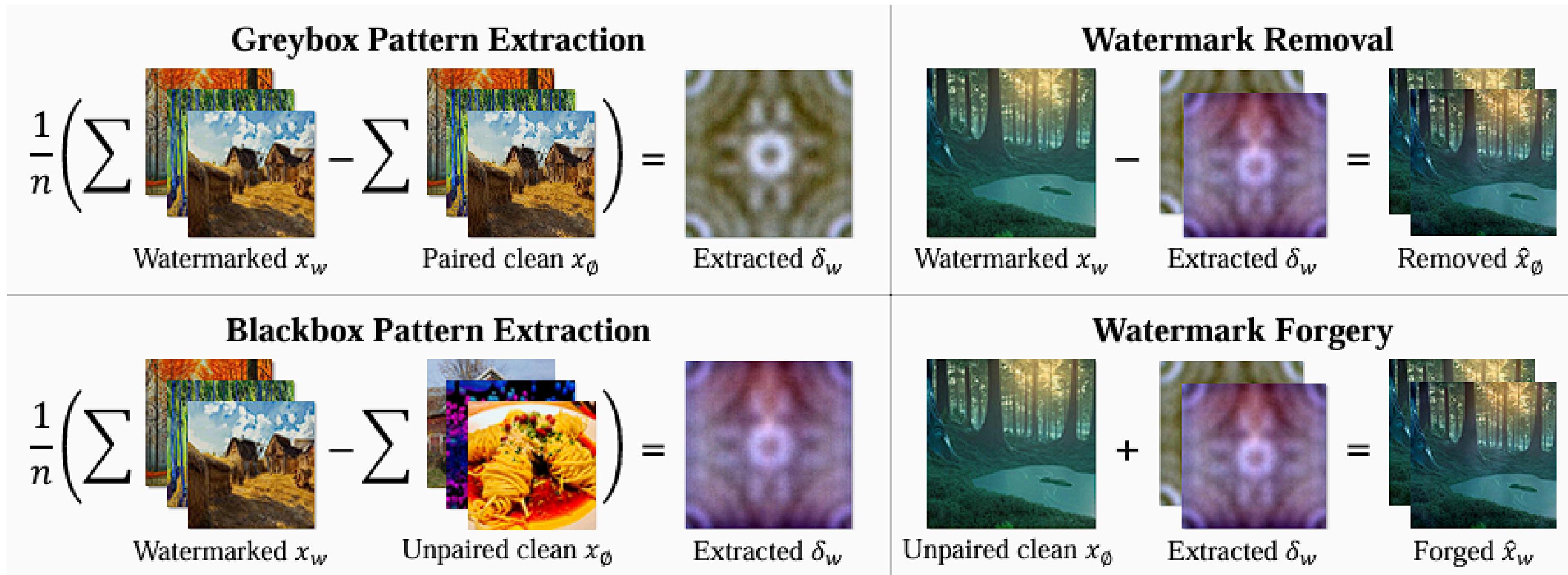


워터마크가 없는 원본 이미지를
공격자가 모르더라도,

워터마크 없는 이미지들 여러 장을
평균해서
 x_\emptyset 를 근사할 수 있다

02 What is Steganalysis?

2.3 Steganalysis : watermark extraction, removal & forgery



- Blackbox Pattern Extraction : x_w 에 해당하는 원본 x_\emptyset 를 몰라도, 인터넷에서 가져온 아무 이미지들을 평균을 계산해서 δ_w 를 추출해낼 수 있음
- Blackbox 환경에서도 공격이 가능함을 보임

Results

3.1 Experimental Setup

1 Image experiments

논문에서 제안한 스테가분석 기반 공격을 12가지 워터마킹 기법에 대해 평가

[Greybox setting]

- WmAdaptor, Stable Signature : COCO 2017 valid set
- Tree-Ring, RingID : Stable Diffusion Prompts 사용
- 그외 : Diffusion DB에서 non-watermarked 이미지 x_\emptyset 로 사용

[평가 지표]

- 워터마크 제거 성능
 - 다양한 n값(평균에 사용되는 이미지수)에 따라 달라지는 패턴 추출 성능을 통해 확인
- 성능 평가 방식
 - Tree-Ring, RAWatermark : Detection AUC
 - 나머지 : Bit acc.
- 이미지 품질 비교 (x_w vs 원본)
 - main text : PSNR
 - 나머지 : SSIM, LPIPS, SIFID

[Blackbox setting]

- ImageNet test set을 깨끗한 이미지로 사용하여 평균 계산

Results

3.1 Experimental Setup

2 Audio experiments

논문에서 제안한 스테가분석 기반 공격을 12가지 워터마킹 기법에 대해 평가

[워터마킹 기법]

- AudioSeal, WavMark

[공격 방식]

- Greybox : 원본 오디오를 사용하여 공격
- Blackbox : 원본 없이 임의의 깨끗한 오디오 평균 내서 공격

[Dataset]

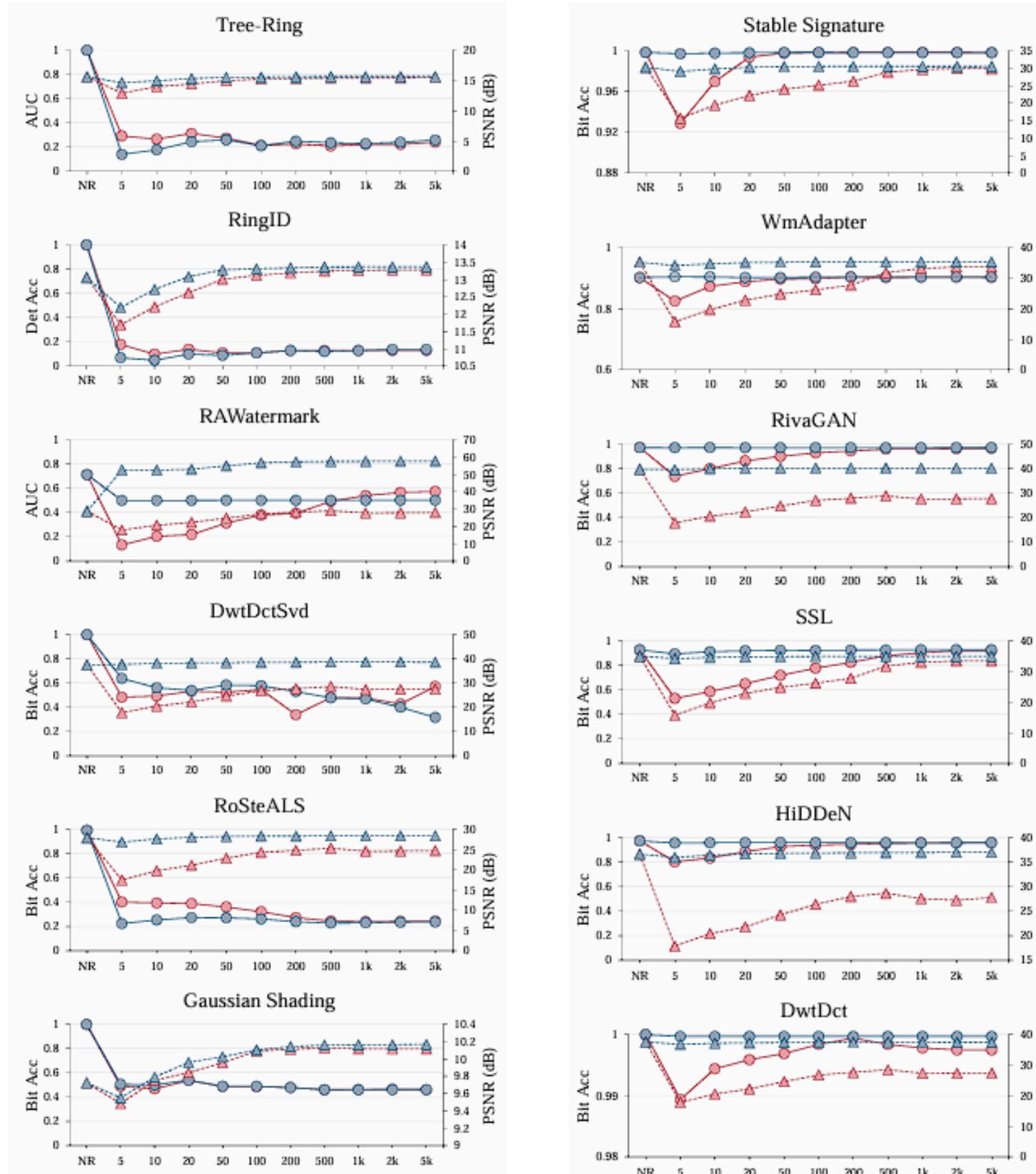
- Common Vooice 중 zh-CN(중국어) 하위 set 사용

[평가 지표]

- AudioSeal → 워터마크 탐지 정확도(Det acc.)
- WavMark → 워터마크 디코딩 정확도 (Bit acc.)
- SI-SNR → 워터마크 제거 후 오디오 품질 변화 측정

03 Results

3.2 Quantitative analysis on watermark removal



[목표]

- 목표 : 스테가분석 기반 워터마크 제거 방식이 실제 워터마크 탐지 성능에 미치는 영향을 평가
- **-*-*-*** : Blackbox / **-**-**-** : Greybox 성능 지표

[실험 결과]

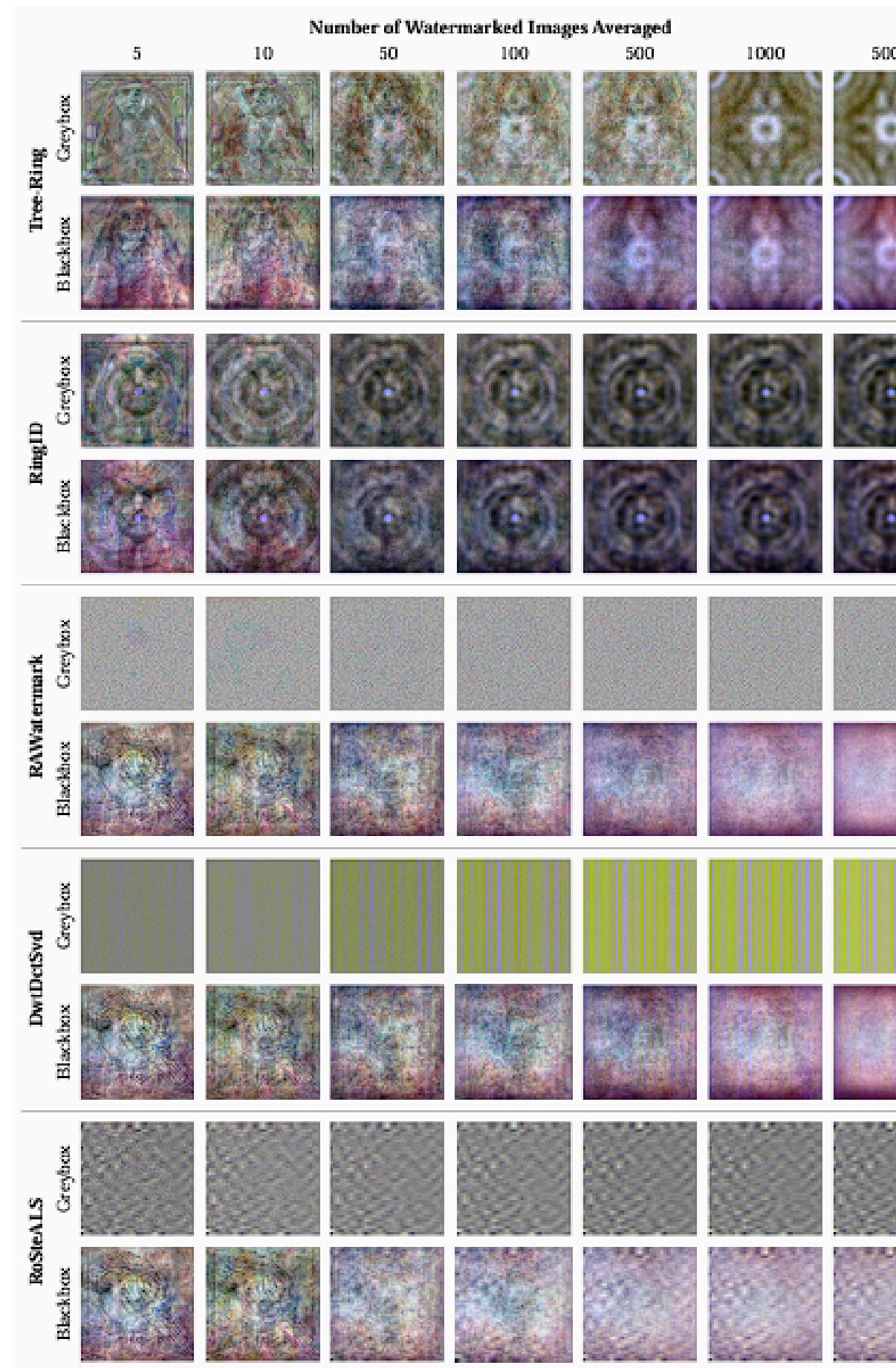
- Content-agnostic 기법 (이미지 왼쪽 열)
 - **스테가분석 기반 공격에 매우 취약**
 - Tree-Ring (AUC 0.24), RoSteALS (Bit acc. 0.24), RAWatermark (AUC 0.57) 등 탐지 성능 급감
 - n이 작을 때, 탐지기의 노이즈에 인한 혼란으로 AUC 급격히 하락
 - n이 점차 커지면서 워터마크 패턴 제거 효과 상승 → AUC 낮게 유지
- Content-adaptive 기법
 - 스테가분석 공격에도 높은 탐지 정확도 유지 (AUC or Bit acc. > 0.95)
 - 강한 복원력 → 콘텐츠 기반 워터마킹 설계의 중요성 부각

[추가 Insights]

- 스테가분석은 content-agnostic 워터마킹 기법에 대하여 **단순 평균만으로 워터마크 패턴을 효과적 추출 가능**
- n(평균 대상 이미지 수) 증가 → 패턴 정확도 ↑, 시각적 왜곡 ↓

03 Results

3.3 Qualitative analysis



1 Extracted Patterns

[실험 개요]

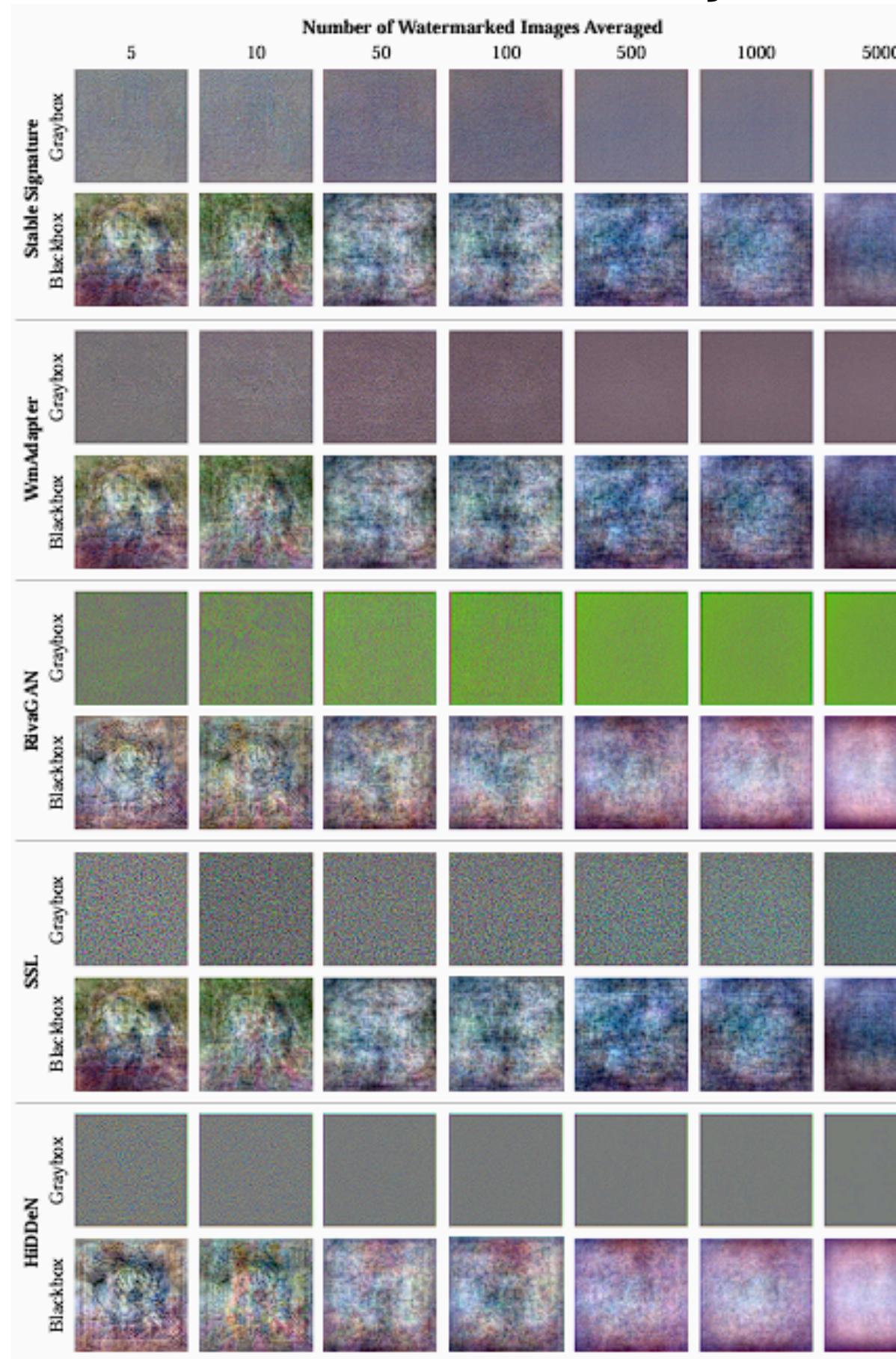
- 다양한 워터마킹 기법에서 추출한 패턴을 먼저 조사하고, 이러한 워터마크 제거가 이미지 품질에 어떠한 영향을 미치는지 논의함

[실험 결과] : <Content-agnostic 기법>

- 이미지와 상관없는 고정된 저수준 패턴을 삽입
- 명확한 시각적 패턴이 존재 → 지문처럼 식별 가능
 - RingID : 중앙에 밝은 점이 있는 동심원
 - DwtDctSvd : 바코드와 유사한 수직선
 - RoSteALS : 비균일한 조명을 가진 격자

03 Results

3.3 Qualitative analysis



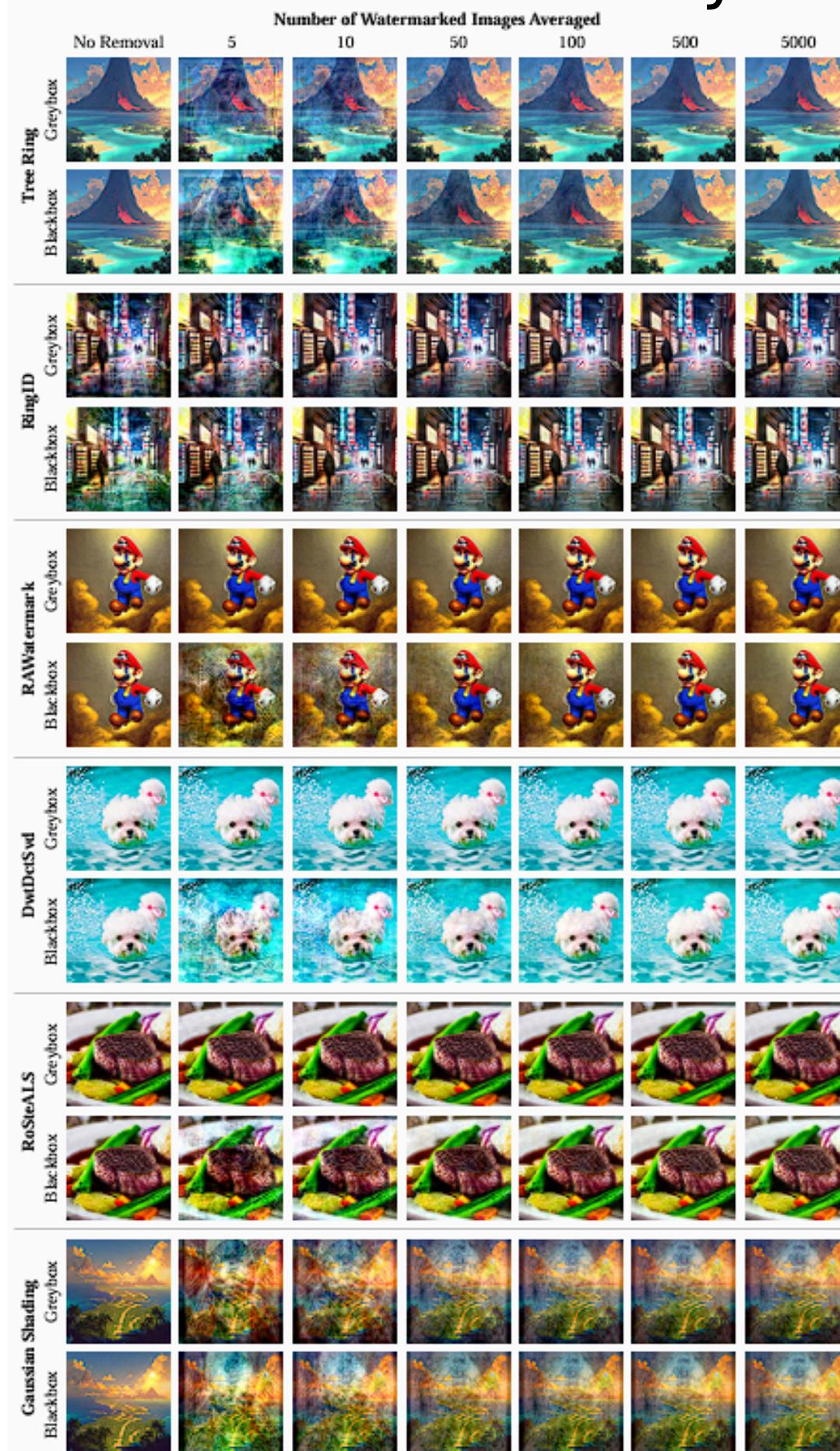
1 Extracted Patterns

[실험 결과] : <Content-adaptive 기법>

- 추출된 패턴이 뚜렷하지 않다
- e.g. HiDDeN : greybox 설정에서 추출된 패턴이 0에 수렴
- 스테가분석 기반 공격에 대한 저항성이 높음

03 Results

3.3 Qualitative analysis



2 Visual quality degradation

[실험 결과]

- Greybox setting:
 - 대부분의 기법에서 50장 이상 이미지 평균 시, 워터마크 제거 후 시각적 아티팩트가 거의 없음
- Blackbox setting:
 - 100장 이상 평균 시, 대부분의 아티팩트가 제거됨

[예외]

- Gaussian Shading 기법:
 - 추출된 워터마크 패턴의 세기(magnitude)가 크기 때문에 제거 시 명확한 시각적 왜곡이 발생함
 - 이는 워터마크 제거가 이미지 품질에 큰 영향을 미치는 유일한 경우

[결론]

- 충분한 수의 이미지 평균(n)을 통해 시각적 품질 저하 없이 워터마크 제거 가능

03 Results

3.4 Case Study: Tree-Ring watermarks

- 스테가분석 기반 공격이 content-agnostic 워터마킹 알고리즘에 왜 큰 위협이 되는지 탐구

Tree-Ring

1. 고정된 주파수 기반의 ripple 패턴을 설계

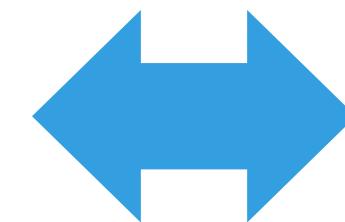
- 이는 이미지 내용과 무관한 패턴
- 보통 2D sinc 함수의 중첩처럼 중심에서 퍼져나가는 동심원 형태의 패턴

2. 위 패턴을 이미지 생성 시 초기 노이즈(latent noise)에 직접 추가

- 이후 일반적인 diffusion 과정을 통해 이미지를 생성

3. Tree-Ring 검출기

- 생성된 이미지에서 DDIM 역변환을 통해 다시 초기 노이즈로 되돌린 후
- 그 안에 리플 패턴이 있는지를 확인하여 워터마크 존재를 판단함



Tree-Ring 전용 스테가분석 공격

1. DDIM Inversion 적용

- 여러 개의 워터마크 이미지에 대해 DDIM 역변환을 수행하여 초기 latent space로 복원

2. 평균 기반 ripple 패턴 추출

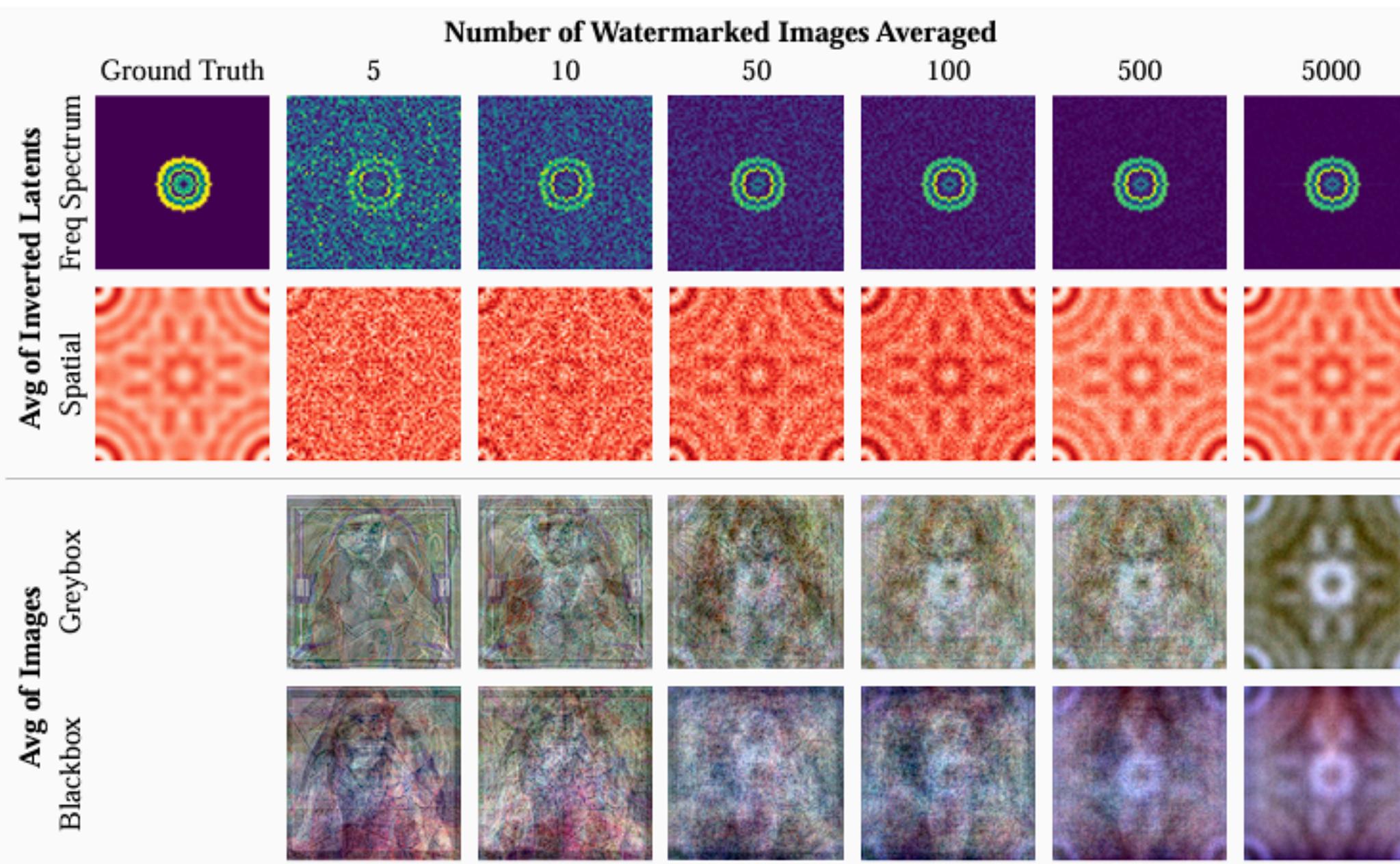
- 워터마크가 포함된 이미지들에서 DDIM으로 복원된 latent를 평균 냄
- latent space에서 ripple 패턴을 평균 냄으로써 더 선명하게 추출할 수 있음



Tree-Ring은 늘 동일한 ripple 패턴을 삽입하니까, 여러 개의 워터마킹 이미지의 차이 평균을 통해 그 패턴을 추출할 수 있다는 점을 이용함!!

03 Results

3.4 Case Study: Tree-Ring watermarks



[구조 요약]

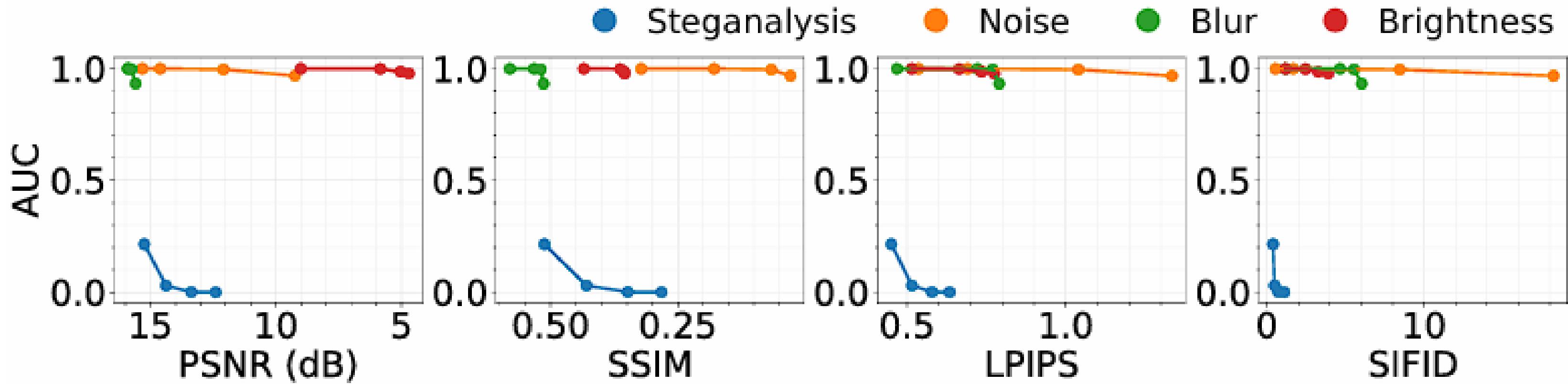
- 가로 방향: 워터마크가 삽입된 이미지의 개수를 평균한 결과
- 세로 방향:
 - Freq Spectrum : DDIM 역변환된 latent의 푸리에 변환
 - Spatial : DDIM 역변환된 latent들의 평균
 - Greybox : 워터마킹 이미지와 깨끗한 이미지의 쌍을 평균해 패턴 추출
 - Blackbox : 워터마킹 이미지만으로 평균해 패턴 추출

[실험 결과]

- 패턴이 반복적이고 content-agnostic 하다는 것의 증거
 - Ground Truth 패턴과 5000장을 평균한 결과는 거의 동일
 - Tree-Ring이 삽입하는 고정된 주파수 기반 워터마크임을 시각적으로 나타냄
- DDIM Inversion은 매우 효과적인 추출 방식
 - Tree-Ring은 latent noise z 단계에 워터마크를 삽입하므로 최종 생성된 이미지만으로는 z를 알 수 없음
 - 그러므로 latent domain에서 분석하는 것이 매우 효과적임
- 결론적으로 Tree-Ring이 사용하는 고정된 ripple 패턴이 스템가분석으로 추출 가능하며, 깨끗한 이미지 없이 blackbox 세팅에서 평균만으로 추출 가능함을 시각적으로 증명함

03 Results

3.4.1 Comparison with distortion-based removal techniques



[그래프 정보]

- x축 : 이미지 품질 측정 지표
- y축 : AUC, 워터마크 탐지 정확도(작을수록 워터마크 제거가 잘 되었음을 의미)

[실험 결과]

- 파란색 점(스테가분석 기반 제거) 분포:
 - 왼쪽 아래에 몰려있음 = AUC가 낮고, 이미지 품질 손상 적음 → 워터마크도 잘 제거되고, 이미지도 안 망가뜨림!
- 반면, 기존 왜곡 기반 제거(노이즈, 블러, 밝기 조정)는 이미지 품질 손상에 비해 워터마크 제거 효과가 낮음

03 Results

3.4.2 Watermark forgery

[Detection Accuracy at 1% FPR]

# Imgs Avged	NRmv	5	10	20	100	200	500	1000	2000	5000
Removal	1.00	0.08	0.09	0.13	0.13	0.13	0.14	0.14	0.14	0.14
Forgery	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

[목적]

- Tree-Ring 워터마크에 대해 단순 제거뿐만 아니라, 비워터마크 이미지에 워터마크를 위조(Forgery) 가능함을 입증

[방법]

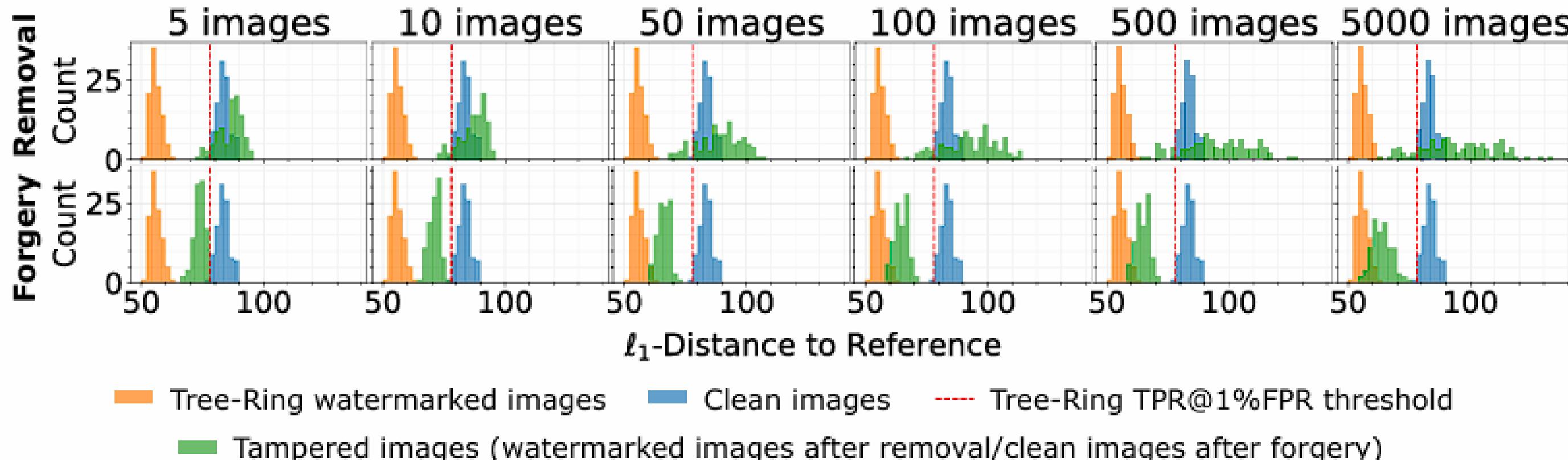
- 워터마크가 없는 이미지에 추출한 워터마크 패턴을 더함으로써 위조 이미지 생성: $\hat{x}_w = T(x_\emptyset) = x_\emptyset + \hat{\delta}_w$
- 1% FPR에서의 탐지 정확도를 측정하여 나타냄

[실험 결과]

- Removal : 수치가 0.00은 아니므로 완벽한 제거는 아니지만, 탐지 정확도가 0.08~0.14 수준으로 대부분의 경우 탐지기가 제거된 이미지를 워터마크가 없다고 착각함
- Forgery : 0.00 → **위조된 이미지가 Tree-Ring 검출기를 완벽히 속임**

03 Results

3.4.2 Watermark forgery



[그래프 정보]

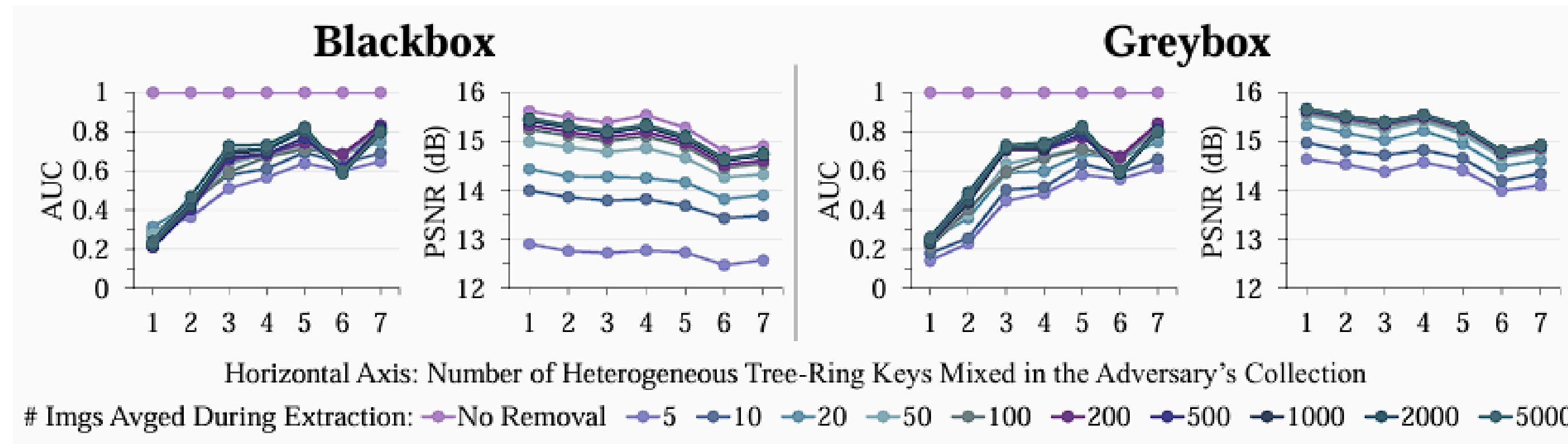
- x축 : L1-distance to Reference
 - 기준 워터마크 패턴과의 거리, Tree-Ring 워터마크 탐지기는 이 거리를 기준으로 워터마크 존재를 판단
- y축 : 이미지 수(Count) → 거리 범위 내에 해당되는 이미지 수

[실험 결과]

- n이 커질수록:
 - 위조 : Tampered 이미지가 주황색 워터마크 이미지와 겹쳐짐 → Tree-Ring 탐지기가 진짜 워터마크 이미지라고 착각함
 - 제거 : Tampered 이미지의 분포가 다양한 콘텐츠 이미지에 적용되면서 제거 결과의 다양성(분산)이 생김

03 Results

3.4.3 Effectiveness of removal under multiple watermarks



[목적]

- 공격자의 이미지 컬렉션에 서로 다른 여러 종류의 워터마크 패턴(key)이 포함되어 있을 때, 스테가분석 기반 워터마크 제거가 잘 작동하는지?
 - Tree-Ring은 워터마크를 삽입할 때 key를 기반으로 특정한 패턴을 생성함

[그래프 정보]

- x축 : key 수
- y축 : AUC, PSNR

[실험 결과]

- key가 많을수록
 - AUC ↑ : 0.2 → 0.7로 증가, 워터마크 탐지가 더 잘됨
 - PSNR ↓ : 이미지 품질 하락
- 다양한 key로 삽입된 서로 다른 워터마크가 섞이면 공통된 패턴 추출이 어려워져서 공격효과가 감소된다
- 이러한 **multi-key watermarking**이 방어 전략이 될 수 있지만, AUC = 1.0 (완벽 탐지)는 아니므로 **스테가분석의 기본적 방어는 되지 못함**

03 Results

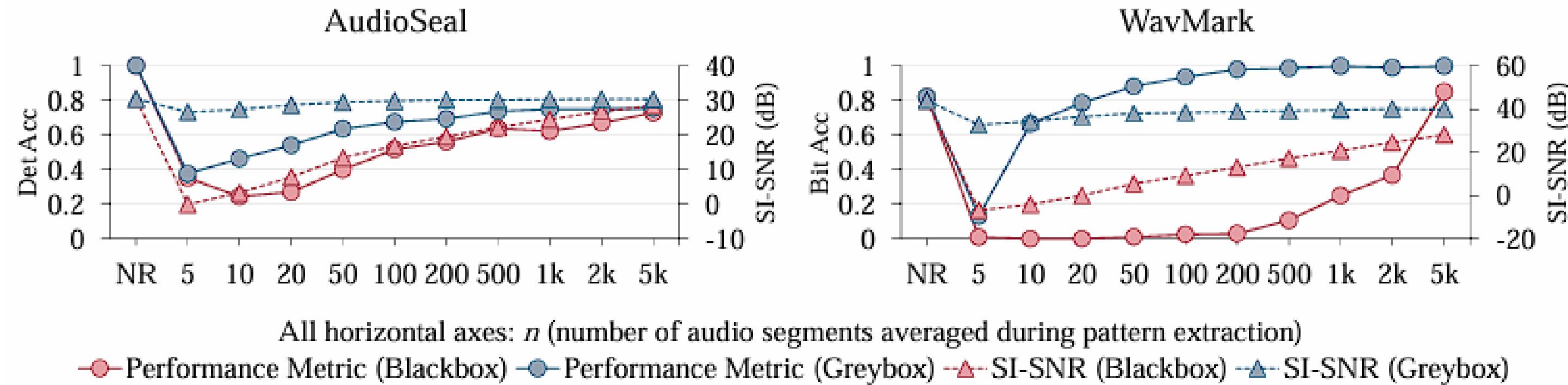
3.4.4 Case Study(Tree-Ring) Summary

✓ 핵심 요약

- Tree-Ring의 보안 취약점은 이미지의 의미(Semantic)와 무관한 저수준(content-agnostic)의 ripple 패턴을 워터마크로 사용하는 데 있음
- 이로 인해, 알고리즘에 대한 접근 없이도, 단순한 스테가분석 기반 평균 기법으로 워터마크 제거 가능
- 이전 연구에서 Tree-Ring은 왜곡이나 재생성 공격에는 강한 것으로 알려져 있었지만,
- 이 논문의 스테가분석 기반 공격으로 Tree-Ring 워터마크를 효과적으로 제거한 최초의 사례임

03 Results

3.5 Audio Watermark Steganalysis



[목적]

- 이미지 외 오디오에도 스테가분석 기법이 효과적인가? 를 검증
- 두 가지 오디오 워터마킹 기법 대상:
 - AudioSeal, WavMark
- 이미지는 픽셀 평균을 계산했다면, 오디오는 시간 도메인에서 평균을 계산하여 패턴 추출

[실험 결과]

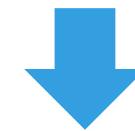
- AudioSeal : 탐지 정확도 : $1.0 \rightarrow 0.75$ 로 급감, 아주 간단한 평균 기반 스테가분석에도 취약
- WavMark : 특이하게도, n 이 커질수록 bit acc.가 상승 ($0.8 \rightarrow 1.0$)
 - 추출된 패턴이 워터마크 정보와 상관 있는 systematic bias일 가능성
 - 향후 정교한 공격에 취약할 수 있음을 시사

03 Results

3.6 Guidelines towards steganalysis-secure watermarking

content-agnostic한 워터마킹 기법들이 **스테가분석 기반 공격에 취약함**을 드러냄

특히 Tree-Ring과 같은 심층 신경망을 기반으로 하는 복잡하고 비선형적인 방법조차도 취약함



워터마크는 **content-adaptive** 이어야 함

- **이미지나 오디오 특성을 고려하여 삽입**
- HiDDeN : 이미지 특성을 concatenation 방식으로 삽입
- RivaGAN : attention으로 이미지특성 활용

다양한 스테가분석 공격에 대한 평가가 필수적

- RGB, 16kHz 오디오 대상 실험을 넘어서
- **다양한 색공간, 주파수 영역 등에서의 공격에도 대비**
- 흔히 왜곡(압축, 노이즈)에 대한 강건성 평가를 많이 다루지만
- 스테가분석 기반 공격도 **워터마크 보안 평가의 필수 항목**으로 다루어야 함

Future Directions

- 1** 더 정교한 content-aware watermarking 알고리즘 설계의 필요성
 - 이미지 특징 기반 위치 선정, attention 기반 삽입 관련 연구 조사

- 2** 단순 평균 연산을 넘어선 더 정교한 방법의 steganalysis 기반 공격에 대한 연구 조사