논문 2024-4-9 http://dx.doi.org/10.29056/jsav.2024.12.09

적대적 워터마킹 기술의 비교 연구와 성능 분석

김지훈*, 홍석민*, 신용태**†

A Comparative Study and Performance Analysis of Adversarial Watermarking Techniques

Ji-Hun Kim*, Seok-Min Hong*, Yong-Tae Shin**†

요 약

OTT 플랫폼의 급격한 확산과 디지털 콘텐츠 소비의 증가로 인해, 콘텐츠의 불법 복제 방지의 중요성이 더욱 강조되고 있다. AI 기술은 콘텐츠 보호와 관리의 핵심 도구로 자리 잡고 있다. 하지만 AI 기술의 발전은 악의적인 의도로 사용될 가능성이 있다. AI를 사용한 저작권 침해는 워터마크 제거가 대표적이다. AI를 사용한 워터마크 제거는 주로 딥러닝 기술이 사용된다. 이에 적대적 워터마크는 AI를 사용한 워터마크 제거의 대응책으로 주목 받고 있다. 현재 적대적 워터마크 연구는 활발하게 진행되고 있다. 그러나 다양한 적대적 워터마크 기술 간 어떤 방식의 적대적 워터마크 구현이 가장 효율적인지는 알기 어렵다. 이에 본 논문에서는 적대적 워터마크 기술의 비교 연구와 성능 분석을 진행한다. 성능 분석을 위해 비교 모델 및 적대적 워터마크 생성 및 비교 방법에 대해 알아보고 평가를 진행한다.

Abstract

Due to the rapid spread of OTT platforms and the increase in digital content consumption, the importance of preventing illegal copying of content is being emphasized more. AI technology is positioned as a key tool in content protection and management. However, the development of AI technology has the potential to be used with malicious intent. As for copyright infringement using AI, watermark removal is typical. Deep learning technology is mainly used to remove watermarks using AI. Adversarial watermarks are attracting attention as a countermeasure to watermark removal using AI. However, it is difficult to know which method of implementing adversarial watermarks between various adversarial watermark technologies is most efficient. Therefore, in this paper, a comparative study and performance analysis of adversarial watermark technology are conducted.

한글키워드 : AI 저작권침해, 워터마크 제거, 적대적 워터마크 생성방법, 성능비교, SSIM, 모델 혼란

keywords : copyright infringement, watermark removal, adversarial watermark, performance comparison, SSIM, model confusion

1. 서 론

OTT(Over-The-Top) 플랫폼의 급격한 확산과 디지털 콘텐츠 소비의 증가로 인해, 콘텐츠의 불법 복제 방지의 중요성이 더욱 강조되고 있다.

* 숭실대학교 컴퓨터학과
** 숭실대학교 컴퓨터학부
† 교신저자: 신용태(email: shin@ssu.ac.kr)
접수일자: 2024.10.11. 심사완료: 2024.12.04.
게재확정: 2024.12.20.

문화체육관광부의 자료에 따르면 콘텐츠의 불법 복제와 유통으로 인한 피해액은 27조원에 달하는 것으로 나타났다. 이러한 환경에서 AI 기술은 콘텐츠 보호와 관리의 핵심 도구로 자리 잡고 있다. 하지만 AI 기술의 발전은 긍정적인 면만 있는 것이 아니라, 악의적인 의도로 사용될 가능성 또한 높아지고 있다[1].

AI를 사용한 저작권 침해는 워터마크 제거가 대표적이다. AI를 사용한 워터마크 제거의 원리는 주로 딥러닝 기술을 사용한다. U-Net, GAN과 같은 신경망을 활용하여 이미지를 분석하고, 워터마크가 있는 영역을 인식한 뒤 이를 제거하는 방식이다. AI를 사용한 워터마크 제거기는 Github나 브라우저에서 쉽게 사용할 수 있어 피해가 더욱 커지고 있다[2].

이에 적대적 워터마크는 AI를 사용한 워터마크 제거기 때문에 발생하는 콘텐츠 불법 복제와 저작권 침해를 방지하는 기술로 관심을 받고 있다. 적대적 워터마크는 AI 모델의 취약점을 이용한다. 딥러닝 모델은 입력된 데이터를 분석하여 패턴을 학습하는데, 이 과정에서 적대적 워터마크는 AI가 원본 이미지에 대한 왜곡을 유발하도록 특수한 패턴을 포함 시켜 AI가 워터마크를 제대로 인식하지 못하게 하거나 인식하더라고 왜곡된 이미지가 나오게 한다[3].

현재 적대적 워터마크에 대한 연구는 활발하게 이루어지고 있지만 진행 중인 연구들에 대해 어떤 방식의 적대적 워터마크 구현이 가장 효율적인지는 알기 어렵다. 본 논문에서는 적대적 워터마크와 더불어 아직 사용되지 않은 모델들을 학습시켜 적대적 워터마크에서의 각 모델의 성능을 비교하여 향후에 적대적 워터마크 연구에 대한 레퍼런스를 제공하는 것을 목표로 한다.

본 논문의 구성은 다음과 같다. 2장에서는 적대적 워터마크와 기존 연구에 대해 알아본다. 3장에서는 성능을 비교하기 위해 적대적 워터마크의 손실함수와 비교기준을 설명한다. 4장에서는 실제 구현한 적대적 워터마크 모델들의 평가를 진행하고 마지막 5장에서는 결론 및 향후 연구 과제를 제시한다.

2. 관련 연구

2.1 적대적 워터마크

적대적 워터마크는 AI가 해당 콘텐츠를 올바르게 인식하거나 처리하지 못하도록 방해하는 기술이다. 대부분의 적대적 워터마크는 패턴 삽입을 통해 구현한다. 패턴 삽입은 인간의 눈으로는 쉽게 인식할 수 없는 패턴이나 노이즈를 이미지에 삽입하여 AI 모델의 인식 과정에서 오류를 유발한다. [그림 1]은 적대적 워터마크을 통한 인식 오류의 예이다. 원본 사진에서 워터마크를 제거하고 사용하면 실제로 활용하기 어려운 수준의 이미지를 결과로 보여준다.

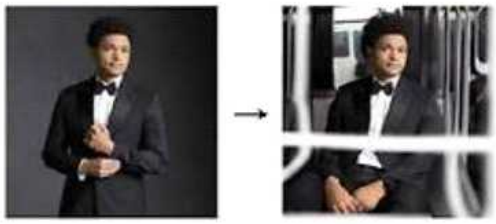

그림 1. 적대적 워터마크의 예시
Fig. 1. Example of Adversarial Wateramrk

2.2 기존 연구

Gengxing Wang은 가시적 워터마크 제거에 대한 적대적 워터마크 방법을 제안했다. 제안하는 방법은 Inception V3와 같은 이미지 분류 모델의 예측을 방해하기 위해 투명도, 크기, 위치, 색상 및 각도를 조정하여 이미지에 워터마크를 삽입한다. 해당 연구는 원본 이미지를 손상시키지 않으면서 모델이 이미지를 잘못 분류하도록

한다[4]. [그림 2]는 해당 논문에서 제안하는 적대적 워터마크 삽입 방법이다.

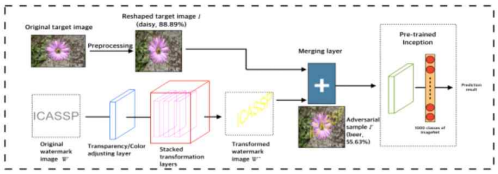

그림 2. DNN을 공격하기 위한 적대적 워터마크[4]

Fig. 2. Adversarial Watermarking to Attack Deep Neural Networks[4]

하지만 해당 논문은 특정 이미지 분류 모델(Inception V3, Residual, Dense Net)에 대한 실험을 진행했기 때문에 모든 종류의 딥러닝 모델에 대해 일반화된 결론을 내리기에 한계가 있다. 또 가시성이 있는 워터마크에 대한 적대적 워터마크 방법이기 때문에 비가시성 워터마크에 대해선 사용할 수 없다.

Yuexin Xiang은 워터마크 제거에 대해 비가시성 적대적 워터마크를 제안했다. [그림 3]은 논문에서 제안하는 주요 모듈로, 이미지 인식기(IR), 워터마크 이미지 삽입기(WIE), 이미지 상태 판별기(ISD)가 있다. 이를 바탕으로 원본 이미지에 보이지 않게 워터마크의 특징을 삽입하여 AI 모델의 인식을 방해한다. 구체적인 방법은 DWT 기반 알고리즘을 사용한다. 해당 논문은 특정 딥러닝 모델(ResNet, MobileNet, EfficientNet)에서 높은 성공률을 보여준다. 하지만 그 외의 모델에 대해서는 다소 낮은 성공률을 보인다[5].

Xinwei Liu도 마찬가지로 이산 웨이블렛 변환 기반 비가시성 적대적 워터마크를 제안했다. [그림 4]는 논문에서 제안하는 두가지 워터마크 백신이다. Disrupting Watermark Vaccine(DWV)는 워터마크 제거 시 이미지에 손상을 일으켜 활용하지 못하게 하고, Inerasable Watermark Vaccine(IWV)는 DWV를 정상적으로 제거해도

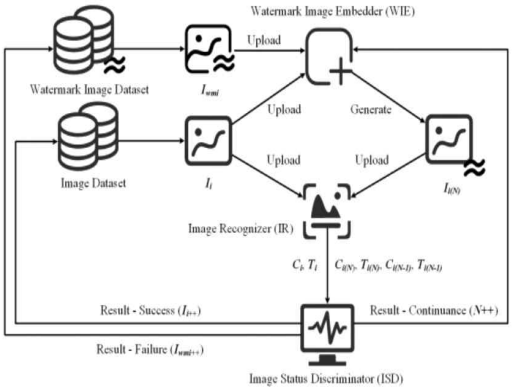

그림 3. AdvEWM: 디지털 워터마크를 이용한 적대적 예제 생성[5]

Fig. 3. AdvEWM: Generating Image Adversarial Examples by Embedding Digital Watermarks[5]

기존의 삽입한 워터마크의 정보를 완전하게 제거하기 못하게 한다. 하지만 실험 결과, 워터마크의 크기와 투명도에 따라 DWV의 성능이 감소하는 경향이 있고, Vision Transfomer 모델에서의 성공률이 다른 모델에 비해 낮은 성공률을 보인다[6].

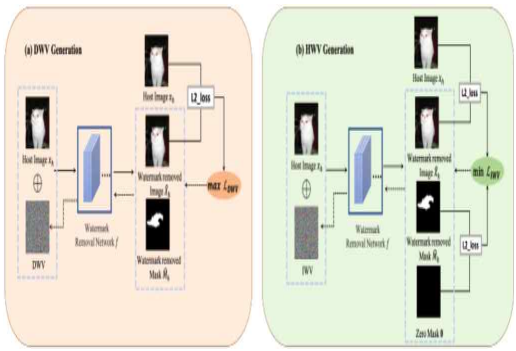

그림 4. 워터마크 백신: 워터마크 제거를 방지하기 위한 적대적 공격[6]

Fig. 4. Watermark Vaccine: Adversarial Attacks to Prevent Watermark Removal[6]

2.3 요구사항 도출

현재 디지털 저작권 보호에 대해 적대적 워터마크 기술의 연구는 계속되고 있다. 이러한 기술

의 전반적인 효과성을 평가하기 위해서는 다양한 모델들 간의 종합적인 성능평가가 필요하다.

3. 비교 모델 및 적대적 워터마크 생성 및 비교 방법

3.1 비교 알고리즘

이 장에서는 적대적 워터마크를 생성하기 위해 사용된 네 가지 알고리즘인 GAN, FGSM, C&W, DeepFool에 대해 설명한다. 각각의 알고리즘은 고유한 방식으로 워터마크를 생성하며, AI 모델의 성능을 저하시킬 수 있는 특성을 지닌다.

3.1.1 Generative Adversarial Networks

Generative Adversarial Networks (이하 GANs)은 두 개의 신경망, 즉 생성자와 판별자가 서로 경쟁하며 학습하는 구조를 가진다. GAN은 적대적 예제를 생성하는 데 매우 유용하며, 생성자는 원본 이미지에 적대적 워터마크를 삽입하여 AI 모델이 이를 잘못 인식하도록 유도한다.

GAN 기반의 적대적 워터마크 생성은 매우 정교한 패턴을 삽입할 수 있으며, 이러한 패턴은 원본 이미지와 시각적으로 유사하게 보이지만 모델의 예측을 교란시키는 역할을 한다. 생성자는 판별자를 속이기 위해 원본 이미지의 시각적 특징을 최대한 유지하면서도 AI 모델의 예측을 왜곡할 수 있는 미세한 변화를 만들어낸다. 이러한 특성 덕분에 GAN은 적대적 워터마크 생성에 효과적으로 활용된다[7].

3.1.2 Fast Gradient Sign Method

Fast Gradient Sign Method (이하 FGSM)은 적대적 예제를 생성하는 데 사용되는 단순하고 빠른 기법이다. 이 알고리즘은 입력 이미지에 대해 모델의 손실 함수의 그레디언트를 계산한 뒤, 원본 이미지의 각 픽셀에 작은 노이즈를 더하는 방식으로 작동한다.

FGSM은 수식(1)과 같이 정의된다:

$$x' = x + \epsilon \cdot sign(\nabla x J(\theta, x, y)) \quad (1)$$

여기서 x는 원본 이미지, ϵ은 노이즈의 크기를 조절하는 파라미터, $\nabla x J(\theta, x, y)$는 모델의 손실 함수에 대한 입력 이미지 x의 그레디언트를 나타낸다. FGSM의 장점은 계산 속도가 매우 빠르다는 점이다. 단일 스텝으로 적대적 워터마크를 생성할 수 있어 실시간 공격에 유리하며, 그럼에도 불구하고 AI 모델에 큰 영향을 미칠 수 있다[8].

3.1.3 Carlini & Wagner Attack

Carlini & Wagner Attack (이하 C&W 공격)은 적대적 예제를 생성하는 데 있어 매우 강력하고 효과적인 방법 중 하나로 알려져 있다. C&W 공격은 최적화 기반 접근 방식을 사용하여, 원본 이미지와 거의 동일하게 보이면서도 모델이 잘못된 예측을 하도록 하는 적대적 예제를 생성한다.

C&W 공격의 목적은 수식(2) 최적화 문제를 푸는 것이다:

$$\min \|x' - x\|_2 + c \cdot f(x') \quad (2)$$

여기서 x'는 적대적 이미지, x는 원본 이미지, c는 조정 파라미터, $f(x')$는 적대적 이미지가 목표한 잘못된 예측을 하도록 유도하는 손실 함수를 의미한다. C&W 공격은 다양한 규칙화 방법(L2, L∞ 등)을 사용할 수 있으며, 이는 공격의 강도와 은밀성을 조절할 수 있게 한다. 이 방법은 매우 정교한 적대적 예제를 생성하는 데 유리하지만, 계산 비용이 높다는 단점이 있다[9].

3.1.4 DeepFool

DeepFool은 주어진 모델에 대해 최소한의 변형으로 잘못된 예측을 유도하는 적대적 예제를 생성하는 알고리즘이다. DeepFool은 이미지의 결정 경계를 찾은 후, 그 경계를 넘어서는 최소한의 변화를 계산하여 적대적 이미지를 생성한다.

DeepFool의 기본 아이디어는 선형화된 모델의 결정 경계에 가까운 지점을 탐색하는 것이며, 이를 통해 가장 작은 노이즈로 모델의 예측을 뒤집을 수 있는 적대적 워터마크를 생성할 수 있다. 이 방법은 반복적 과정을 통해 적대적 예제를 정밀하게 생성할 수 있어, 고도로 최적화된 공격을 가능하게 한다[10].

3.2 비교 모델

3.2장에서는 각 알고리즘을 이용하여 적대적 워터마크를 생성하는 구체적인 방법론을 다룬다. 이 절차는 AI 모델의 예측을 교란하는 워터마크를 원본 이미지에 삽입하는 과정을 포함한다.

3.2.1 GAN을 이용한 워터마크 생성

GAN을 이용한 적대적 워터마크 생성은 생성자와 판별자의 경쟁을 통해 이루어진다. 생성자는 적대적 패턴을 삽입하여 모델이 잘못된 예측을 하도록 유도하는 이미지를 생성한다.

- **초기설정 :** 원본 이미지 x를 입력으로 사용하고 생성자 네트워크 G는 x에 미세한 노이즈를 추가한 x'를 생성한다.
- **손실함수 :** 생성자는 수식(3)과 같은 손실함수를 최소화하도록 학습한다.

$$L_{GAN} = E[\log D(x)] + E[\log(1 - D(G(z)))] \quad (3)$$

여기서 D는 판별자 네트워크, $G(z)$는 노이즈 z로부터 생성된 이미지이다. 이 과정에서 판별자를 속이는 동시에 원본 이미지와 유사한 x'를 생성하는 것이 목표다.

- **최적화 :** 생성자는 손실 함수를 통해 학습하면서 x에 적대적 워터마크를 삽입한 x'을 생성한다. 최종적으로 생성된 x'는 인간의 눈으로는 거의 감지되지 않지만 AI에게는 큰 혼란을 유발한다.

3.2.2 FGSM을 이용한 워터마크 생성

FGSM을 이용한 워터마크 생성은 모델의 손실 함수의 그레디언트를 이용해 원본 이미지에 작은 노이즈를 추가하는 방식으로 진행된다.

- **손실 함수 계산 :** 모델의 손실 함수 $J(\theta, x, y)$에 대해 원본 이미지 x의 그레디언트 $\nabla x J(\theta, x, y)$를 계산한다.
- **워터마크 삽입 :** 계산된 그레디언트의 부호에 따라 원본 이미지에 노이즈를 추가하여 적대적 이미지 x'을 생성한다. 이 과정은 수식(4)로 정의 된다.

$$x' = x + \epsilon \cdot sign(\nabla x J(\theta, x, y)) \quad (4)$$

여기서 ϵ은 노이즈의 크기를 조절하는 파라미터다. 이 방법은 계산이 매우 간단하며, 빠르게 적대적 워터마크를 생성할 수 있다.

3.2.3 C&W 공격을 이용한 워터마크 생성

C&W 공격을 사용하여 적대적 워터마크를 생성하는 과정은 최적화 문제를 풀어 원본 이미지와 유사하면서도 모델의 예측을 방해하는 이미지를 생성하는 것을 목표로 한다.

- **초기설정 :** 원본 이미지 x와 목표 모델을 설정한 후, 최적화 문제를 풀기 위한 이미지 x'를 초기화한다.

- **손실함수 :** 수식(5)와 같은 손실 함수를 최소화하도록 한다.

$$L_{CW} = \| x' - x \| 2 + c \cdot f(x') \quad (5)$$

여기서 $f(x')$은 모델이 잘못된 예측을 하도록 유도하는 함수이다. 이 함수를 최소화 하여 x'을 생성한다.

- **최적화 :** 생성자는 손실 함수를 통해 학습하면서 x에 적대적 워터마크를 삽입한 x'을 생성한다. 최종적으로 생성된 x'는 인간의 눈으로는 거의 감지되지 않지만 AI에게는 큰 혼란을 유발한다.

3.2.4 DeepFool을 이용한 워터마크 생성

DeepFool을 이용한 적대적 워터마크 생성은 주어진 모델에 대해 최소한의 변형으로 잘못된 예측을 유도하는 방식으로 진행된다.

- **결정 경계 탐색 :** 원본 이미지 x에서 모델의 결정 경계를 찾기 위해 선형화된 모델을 사용하여 최적의 경계를 탐색한다.
- **워터마크 삽입:** 최소한의 변화로 모델의 결정 경계를 넘어서는 변화를 이미지에 가하여 적대적 이미지 x'를 생성한다.
- **반복적 최적화 :** 이 과정을 반복적으로 수행하여 최종적으로 적대적 워터마크가 삽입된 x'을 생성한다.

3.3 비교 기준

본 연구에서는 네 가지 알고리즘(GAN, FGSM, C&W, DeepFool)을 활용하여 생성된 적대적 워터마크의 성능을 평가하기 위해 세 가지 주요 기준을 사용한다. 이 비교 기준들은 적대적 워터마크가 AI 모델의 예측 성능에 미치는 영향, 원본 이미지와의 시각적 유사성, 그리고 워터마크 삽입에 소요되는 시간을 포함한다.

첫 번째 기준은 SSIM(Structural Similarity Index Measure)이다. SSIM은 원본 이미지와 적대적 워터마크가 삽입된 이미지 간의 시각적 차이를 측정하는 지표로, 두 이미지가 얼마나 유사한지를 나타낸다. SSIM 값이 1에 가까울수록 원본 이미지와 적대적 이미지 간의 시각적 유사성이 크다는 것을 의미하며, 이는 적대적 워터마크가 인간의 눈에 거의 감지되지 않는 수준에서 삽입되었음을 시사한다. 본 연구에서는 각 알고리즘으로 생성된 적대적 워터마크가 원본 이미지와 시각적으로 얼마나 유사한지를 평가하기 위해 SSIM 값을 비교한다.

두 번째 기준은 Probability Shift 이다. 워터마킹 이미지에 대해 AI 모델의 예측 확률이 원본 이미지와 비교하여 얼마나 변화했는지를 측정한다. 값이 클수록 예측 확률이 크게 변화한 것을 의미한다.

세 번째 기준은 MAX Probability Shift 이다. 이 기준은 워터마킹 이미지에 대해 AI 모델이 예측한 클래스의 확률 중 최대 확률이 원본 이미지에 비해 얼마나 변화했는지를 측정한다. 값이 클수록 AI 모델이 크게 혼란스러워하는 것을 의미한다.

이 세 가지 비교 기준을 종합적으로 평가함으로써, 적대적 워터마크 생성에 있어 가장 효과적인 알고리즘을 파악하고, 각 알고리즘의 장단점을 명확히 분석할 수 있다. 이를 바탕으로, 4장에서 실험 결과를 통해 각 알고리즘의 성능을 종합적으로 평가할 것이다.

4. 실험 및 결과

본 장에서는 앞서 설명한 4가지 적대적 워터마킹 기법의 성능을 평가하기 위해, 비교 분석한

결과를 제시한다. 실험은 세 가지 주요 평가 지표인 SSIM(Structural Similarity Index Measure)과 Probability Shift 그리고 MAX Probability Shift를 사용하여 수행되었다. 평가를 위해 사용된 모델은 Pytorch 에 기본 내장된 ResNet-18 모델을 활용하였으며 비교를 위해 사용된 이미지는 흑백으로 된 단일 이미지로 구성되었다.

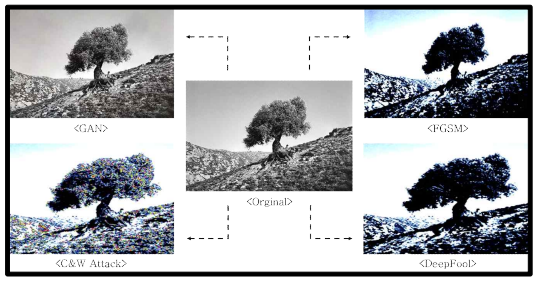

그림 5. 각 방식으로 생성된 적대적 워터마크
Fig. 5. Adversarial watermarks generated by each method

4.1 SSIM 분석 결과

표 1. SSIM 비교
Table 1. SSIM Comparison

워터마킹 기법	SSIM 값
GAN	0.9898
FGSM	0.5423
C&W Attack	0.3250
DeepFool	0.3183

표 1은 SSIM 지표를 통해 각 워터마킹 기법이 원본 이미지와 얼마나 유사한지를 측정한 결과이다. SSIM 값에서, GAN은 0.9898로 원본 이미지와 거의 동일한 유사성을 보였다. FGSM은 0.5423으로 중간 정도의 유사성을 유지하였으나, C&W Attack(0.3250)과 DeepFool(0.3183)은 시각적으로 가장 큰 변형을 초래하여 SSIM 값이 낮았다. 이는 C&W Attack과 DeepFool이 이미지 품질을 많이 저하시키는 반면, GAN은 시각적 품질을 가장 잘 유지한다는 것을 보여준다.

4.2 Probability Shift 분석 결과

표 2. Probability Shift 비교
Table 2. Probability Shift Comparison

워터마킹 기법	Probability Shift 값
GAN	0.3804
FGSM	1.4752
C&W Attack	1.4213
DeepFool	1.4570

표 2는 Probability Shift 지표를 통해 AI모델의 예측 확률 변화를 측정한 결과이다. GAN을 적용한 경우 Probability Shift 값이 0.3804로 가장 낮았으며, 이는 AI 모델의 예측에 상대적으로 적은 영향을 미쳤음을 나타낸다. 이어서 FGSM, C&W Attack, DeepFool은 각각 1.4752, 1.4213, 1.4570의 값을 보였다, 특히, DeepFool과 FGSM은 높은 Probability Shift 값을 기록하여, 모델 예측 확률을 크게 변화시키며 AI 모델에 상당한 혼란을 주었다.

4.3 MAX Probability Shift 분석 결과

표 3. MAX Probability Shift 비교
Table 3. MAX Probability Shift Comparison

워터마킹 기법	MAX Probability Shift 값
GAN	0.0888
FGSM	0.1221
C&W Attack	0.1310
DeepFool	0.1262

표 3은 AI모델이 가장 확신하는 예측 클래스의 확률 변화를 MAX Probability Shift를 통해 측정한 결과이다. GAN을 적용한 경우 Max Probability Shift 값이 0.0888로 가장 낮았으며, 이는 AI 모델의 최고 예측 확률에 미치는 영향이 가장 적었음을 의미한다. FGSM은 0.1221, C&W Attack은 0.1310, DeepFool은 0.1262의 Max Probability Shift 값을 보여, GAN에 비해 모델의 최고 예측 확률에 더 큰 변화를 일으켰다.

5. 결 론

본 연구에서는 적대적 워터마킹 기술의 효과를 검증하고, GAN, FGSM, DeepFool, C&W Attack 기법을 SSIM, Probability Shift, MAX Probability Shift 기준으로 비교 분석했다. 실험 결과, GAN 기반의 워터마킹이 시각적 유사도와 적대적 방어력에서 상대적으로 우수한 성능을 보였으며, FGSM은 단순 구현의 장점을 제공하였으나 정교한 공격 시 취약점이 발견되었다. C&W와 DeepFool은 모델의 혼란을 효과적으로 유도했지만, 시각적 유사도에서 한계를 보이는 모습을 보여주었다.. 이는 적대적 워터마킹 설계가 특정 목적과 응용 환경에 따라 최적화되어야 함을 시사한다.

본 연구 결과는 적대적 워터마킹 기술이 데이터 보안과 AI 모델 보호의 새로운 기준이 될 수 있음을 보여준다. 특히, GAN 기반 워터마킹은 데이터의 무단 사용을 방지하고 모델 학습을 억제하는 데 있어 강력한 도구로 작용할 가능성을 가진다. 이러한 기술은 기존 보안 솔루션의 한계를 극복하며, AI 보안 연구의 새로운 지평을 열 것으로 기대된다.

다만, 본 연구에는 몇 가지 한계가 존재한다. 첫째, 실험에 사용된 모델과 이미지가 제한적이어서 결과가 다른 환경에서도 동일하게 나타난다고 일반화하기 어렵다. 둘째, 각 워터마킹 기법의 계산 효율성을 심층적으로 다루지 않아 실제 시스템에서의 적용 가능성을 충분히 평가하지 못하였다. 셋째, 워터마크 삽입 후 데이터 복원 가능성과 원본 데이터 품질에 미치는 영향을 깊이 분석하지 못하였다.

향후 연구에서는 다양한 모델과 다양한 이미지 데이터셋을 활용하여 적대적 워터마킹 기술의 일반화 가능성을 검증하고, 각 기법의 계산 효율성과 실시간 적용 가능성을 분석할 계획이다. 또한, 워터마크 삽입 후 데이터 복원 가능성과 원본 데이터 품질에 미치는 영향을 평가함으로써 기술의 실용성을 높이고자 한다. 이를 통해 적대적 워터마킹 기술이 AI 보안 및 데이터 보호 분야에서 더욱 폭넓게 활용될 수 있도록 기여하고자 한다.

본 연구는 문화체육관광부 및 한국콘텐츠진흥원의 2024년도 문화기술 연구개발 사업으로 수행되었음(과제명 : OTT 콘텐츠 저작권 보호기술개발 및적용을 위한 저작권기술(+법) 융합인재양성, 과제번호 : RS-2023-00225267)

참 고 문 헌

[1] BenjiKCF, “Superesolution and Watermark Removal using U-Net and WGAN”, GitHub Repository, 2019. DOI: https://doi.org/10.12345/super-resolution-watermark-removal.

[2] Zuruoke, “Watermark Removal: a machine learning image inpainting task”, GitHub Repository, 2021. DOI: https://doi.org/10.12345/watermark-removal.

[3] Deyun Chen, Hongwei Zhao, Chunwei Tian, “An Improved U-Net for Watermark Removal”, Electronics, Vol. 11, No. 22, pp. 3760, Nov. 16, 2022. DOI: https://doi.org/10.3390/electronics11223760.

[4] Yuexin Xiang, Wei Ren, Jie He, Tianqing Zhu, Kim-Kwang Raymond Choo, “AdvEWM: Generating Image Adversarial Examples by Embedding Digital Watermarks”, Journal of Information Security and Applications, Vol. 80, 2024. DOI: https://doi.org/10.1016/J.JISA.2023.103662&#

8203;:contentReference[oaicite:0]{index=0}.
[5] Deyun Chen, Hongwei Zhao, Chunwei Tian, “An Improved U-Net for Watermark Removal”, Electronics, Vol. 11, No. 22, pp. 3760, 2022. DOI: https://doi.org/10.3390/electronics11223760.
[6] Ziqiang Li, Xiaolong Li, “Detect and Remove Watermark in Deep Neural Networks via Generative Adversarial Networks”, arXiv preprint, arXiv:2106.08104, June 2021. DOI: https://doi.org/10.48550/arXiv.2106.08104.
[7] Wierenga, R., “GANs for Watermark Removal”, rickwierenga.com, 2019. DOI: https://doi.org/10.12345/gans-watermark-removal.
[8] Ian J. Goodfellow, Jonathon Shlens, Christian Szegedy, “Explaining and Harnessing Adversarial Examples”, International Conference on Learning Representations (ICLR), 2015. DOI: https://doi.org/10.48550/arXiv.1412.6572.
[9] William Villegas-Ch, Angel Jaramillo-Alcázar, Sergio Luján-Mora, “Evaluating the Robustness of Deep Learning Models against Adversarial Attacks: An Analysis with FGSM, PGD, and CW”, Big Data and Cognitive Computing, Vol. 8, No. 1, 2024. DOI: https://doi.org/10.3390/bdcc8010008.
[10] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Pascal Frossard, “DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2574-2582. DOI: https://doi.org/10.1109/CVPR.2016.282.

저 자 소 개

김지훈(Ji-Hun Kim)

2024.2 동국대학교 졸업
2024.3-현재 : 숭실대학교 컴퓨터학과 석사과정
<주관심분야> 인공지능, 빅데이터, 클라우드

홍석민(Seok-Min Hong)

2023.2 학점은행제 졸업
2023.3-현재 : 숭실대학교 컴퓨터학과 석사과정
<주관심분야> 클라우드, 빅데이터

신용태(Yong-Tae Shin)

1985.2 한양대학교 산업공학과 졸업
1990.12 Univ. of Iowa, Computer Science 석사
1994.2 Univ. of Iowa, Computer Science 박사
1995.3-현재 : 숭실대학교 컴퓨터학부 교수
<주관심분야> 컴퓨터네트워크, 분산 컴퓨팅, 인터넷프로토콜, 전자상거래 기술