

실시간 렌더링 파이프라인을 위한 OpenGL Compute Shader 기반의 고속 워터마킹: 4X4 블록 변환의 레지스터 효율성 및 강건성 분석

남 시 영¹ · 김 혜 영^{2*}¹홍익대학교 게임학과(공학계) 석사과정²홍익대학교 게임학부 교수

Real-Time Watermarking for Rendering Pipelines via OpenGL Compute Shaders: Analysis of Register Efficiency and Robustness in 4X4 Block Transforms

Si-Young Nam¹ · Hye-Young Kim^{2*}¹Master's Course, Department of Game, Hongik University, Sejong 30016, Korea²Professor, School of Game(Major in Game Software), Hongik University, Sejong 30016, Korea

[요 약]

기존의 8x8 블록 기반 워터마킹 기법은 GPU(Graphics Processing Unit) 스레드당 최적 레지스터 용량을 초과하여, 레지스터 스페일링 및 메모리 지연을 유발하는 문제가 있다. 본 논문에서는 OpenGL Compute Shader를 활용하여, 모든 변환 연산을 GPU 레지스터 내에서 완벽하게 처리할 수 있는 4x4 마이크로 블록 기반의 고속 워터마킹 프레임워크를 제안한다. 또한, 블록 크기 축소로 인한 신호 수용력 저하 문제는 반복 부호를 적용하여 보완하였다. PSNR(Peak Signal-to-Noise Ratio) 40dB 내외의 화질 기준에서 수행된 실험 결과, 제안 기법은 기존 기법 대비 FHD와 4K 해상도 모두에서 약 4배 빠른 성능을 달성하였다. 나아가 Crop 및 Salt&Pepper 공격에 대해 0.00의 BER(Bit Error Rate)을 기록하여, 시스템 실행 속도와 강건성 사이의 최적 균형을 입증하였다.

[Abstract]

Conventional watermarking techniques typically rely on 8x8 blocks, which often exceed the optimal register capacity per graphics processing unit (GPU) thread, causing memory latency due to register spilling. This study proposes a high-speed blind watermarking framework based on OpenGL Compute Shaders that adopts a 4x4 micro-block structure. This design enables transformation operations to be executed entirely within GPU registers, eliminating the need to access global memory. To mitigate the reduced signal capacity inherent in smaller blocks, we incorporate a repetition code strategy. Experimental results, obtained with embedding strength calibrated to a target peak signal-to-noise ratio (PSNR) of approximately 40 dB, demonstrate that the proposed scheme achieves approximately fourfold faster performance than conventional methods for both FHD and 4K resolutions. Furthermore, the method achieves a Bit Error Rate (BER) of 0.00 against cropping and salt-and-pepper attacks, demonstrating an optimal trade-off between execution speed and robustness.

색인어 : 4x4 블록 변환, GPU 최적화, 디지털 워터마킹, 반복 부호, 실시간 렌더링**Keyword** : 4x4 Block Transform, Digital Watermarking, GPU Optimization, Real-Time Rendering, Repetition Code<http://dx.doi.org/10.9728/dcs.2026.27.3.777>

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 24 December 2025; Revised 16 January 2026

Accepted 21 January 2026

*Corresponding Author; Hye-Young Kim

Tel: +82-44-860-2683

E-mail: hykim@hongik.ac.kr

I. 서론

최근 5G 통신 인프라의 확충과 GPU(Graphics Processing Unit) 컴퓨팅 성능의 비약적인 발전에 힘입어 4K/8K 초고화질 비디오뿐만 아니라 클라우드 게이밍, VR(Virtual Reality), AR(Augmented Reality), 메타버스 등 고사양 그래픽을 요구하는 실시간 인터랙티브 미디어 시장이 폭발적으로 성장하고 있다. 특히 생성형 인공지능 기술의 등장으로 텍스트, 오디오, 비디오 등 다양한 모달리티의 콘텐츠 생성이 자동화되면서, 허위 정보 유포나 신원 도용, 저작권 침해와 같은 사회적 위협이 급증하고 있다[1]. 이에 따라 콘텐츠의 매 프레임(Frame)에 저작권 정보를 실시간으로 은닉하여 소유권을 보호하는 디지털 워터마킹 기술이 필수적인 보안 요소로 대두되고 있다.

기존의 디지털 워터마킹 연구는 주로 정지 영상이나 오프라인 비디오 처리에 초점을 맞추어 왔다. 비디오 게임 분야에서는 3D 메쉬 모델이나 텍스처 데이터를 보호하기 위한 시도가 있었으나[2], 최근에는 게임 텍스처 포맷인 DDS(DirectDraw Surface) 파일에 직접 정보를 은닉하려는 시도도 이루어졌다[3]. 그러나 이러한 기법들은 대부분 사후 처리 방식이거나 복잡한 연산 과정을 거치기 때문에, 실시간 렌더링 환경에서 높은 처리 지연을 유발하는 단점이 있다. 프레임 레이트(Frame Rate)와 해상도가 사용자 체감 품질에 직결되는 고사양 게임이나 스트리밍 환경에서는 1ms의 미세한 지연 시간조차 치명적인 영향을 미칠 수 있다[4].

현대의 이미지 병렬 처리 연구에 따르면, 데이터 전송 시간이 전체 연산 성능을 좌우하는 핵심 요인이 밝혀졌다[5]. 기존 기법들은 CPU(Central Processing Unit) 처리에 최적화되어 있거나, GPU를 사용하더라도 8x8 매크로 블록 또는 전역 변환 방식을 사용하여 심각한 메모리 병목 현상을 초래한다. 게임 엔진은 렌더링을 위해 GPU 자원을 극한으로 활용하는데, 워터마킹 과정에서 과도한 비디오 메모리 대역폭을 점유하거나 오프칩 메모리 접근을 빈번하게 유발할 경우 전체 렌더링 파이프라인의 프레임 레이트 저하를 피할 수 없다.

저자들의 선행 연구[6]에서는 이러한 문제를 해결하기 위해 Unity SRP(Scriptable Render Pipeline) 환경에서 LSB(Least Significant Bit), DCT(Discrete Cosine Transform), DWT(Discrete Wavelet Transform) 기반 워터마킹을 적용하여 실시간 화질 성능을 타진한 바 있다. 그러나 해당 연구는 상용 엔진의 추상화 계층으로 인해 워터마킹 알고리즘 고유의 순수 GPU(Graphic Process Unit) 부하를 정밀하게 격리하여 측정하는 데 한계가 있었으며, 기하학적 변환 공격 등에 대한 강건성(Robustness) 검증 또한 부재하였다. 따라서 게임 및 고해상도 비디오의 렌더링 파이프라인에 병목 없이 통합되기 위해서는 엔진 종속성을 배제한 환경에서 GPU 아키텍처 특성을 고려한 정밀한 성능 분석과 초경량 설계를 통한 강건성 확보가 필수적이다.

이에 본 논문에서는 OpenGL 기반의 네이티브 환경을 구축하여 분석 대상을 SVD(Singular Value Decomposition) 및 DFT(Discrete Fourier Transform) 기법까지 확장하고, 실시간 저작권 보호를 위해 GPU 렌더링 파이프라인에 프레임 단위로 통합 가능한 4x4 블록 기반 One-Shot 워터마킹 기법을 제안한다. 제안하는 기법은 4x4 마이크로 블록 구조와 레지스터 타일링 최적화를 통해 GPU 메모리 대역폭 의존도를 최소화하였으며, 이를 통해 렌더링과 워터마킹을 단일 커널에서 처리하는 One-Shot 수행이 가능하다.

본 논문의 구성으로, 2장에서는 주파수 도메인 워터마킹의 원리와 GPU 병렬 처리 아키텍처의 특성을 살펴본다. 3장에서는 제안하는 4x4 One-Shot 파이프라인의 구조를 상세히 기술한다. 4장에서는 실험을 통해 제안 기법의 연산 효율성, 비가시성과 강건성을 분석하며, 마지막으로 5장에서는 결론 및 향후 연구 방향을 제시한다.

II. 주파수 도메인 워터마킹 및 GPU 병렬 처리 아키텍처

2-1 디지털 워터마킹과 요구 사항

디지털 워터마킹은 멀티미디어 콘텐츠에 소유권 정보를 암호화하여 은닉하는 기술로, 일반적으로 비가시성(Imperceptibility), 강건성, 그리고 삽입 용량(Capacity)의 세 가지 핵심 요소 간의 트레이드오프(Trade-off) 관계를 가진다. 초기의 공간 도메인 기법은 픽셀의 진폭 변조 등을 통해 구현이 간단하고 대용량 삽입이 가능하나[7], 신호 처리 공격에 취약하다는 한계가 있다. 이에 따라 현대의 연구들은 영상 신호를 주파수 성분으로 변환하여 인간 시각 시스템(HVS; Human Visual System)이 덜 민감한 대역에 정보를 삽입하는 주파수 도메인 방식을 주로 채택한다[8]. 특히 Cox 등[9]이 제안한 확산 대역(Spread Spectrum) 기법은 워터마크 신호를 가우시안 잡음 형태로 주파수 전역에 분산시켜, 압축이나 필터링과 같은 다양한 공격에 대해 탁월한 강건성을 입증하였다. 또한, HVS의 마스크 효과를 고려하여 텍스처가 복잡하거나 밝기 변화가 심한 영역에 더 강한 워터마크를 삽입함으로써 비가시성을 극대화하는 연구도 지속되고 있다[10].

본 저자들은 선행 연구[6]를 통해 상용 게임 엔진(Unity) 환경에서 LSB, DCT, DWT 기법을 구현하고 실시간 렌더링 시의 화질 효율성을 분석한 바 있다. 해당 연구는 게임 엔진 내 워터마킹의 적용 가능성을 확인했다는 점에서 의의가 있었으나, 엔진 내부의 복잡한 추상화 계층으로 인해 순수 알고리즘의 연산 부하를 정밀하게 측정하는 데 제약이 따랐다. 또한, 당시 연구 범위는 화질 분석에 집중되어 있어, 기하학적 변환이나 신호 처리 공격에 대한 강건성 검증은 미비하였다. 따라서 실시간 렌더링 환경에서 신뢰성 있는 저작권 보호를

수행하기 위해서는 기존의 화질 중심 분석을 넘어, SVD 및 DFT 등 다양한 변환 기법의 수학적 강건성과 GPU 아키텍처에 최적화된 연산 방식에 대한 심층적인 고찰이 선행되어야 한다. 본 장에서는 제안 시스템의 설계 기반이 되는 각 변환 알고리즘의 특성과 GPU 병렬 처리 시의 주요 제약 사항을 구체적으로 분석한다.

2-2 주파수 변환 알고리즘의 특성

본 연구에서는 영상 처리에 널리 사용되는 네 가지 대표적인 변환 기법을 활용한다. 각 알고리즘은 고유한 수학적 특성에 따라 상이한 강건성과 연산 부하를 가진다.

1) DCT (Discrete Cosine Transform)

DCT는 JPEG(Joint Photographic Experts Group) 압축 표준의 기반이 되는 기술로, 신호의 에너지를 저주파 성분에서 집중시키는 에너지 압축 효율이 매우 뛰어나다. Katharotiya 등[11]의 비교 분석에 따르면, DCT는 연산 속도 면에서 유리하지만 고압축 환경에서의 블록화 현상이 발생할 수 있다. 이를 보완하기 위해 중간 주파수 대역을 활용하거나 전처리 필터링을 결합하여 검출율을 높이는 연구가 진행되고 있다 [12],[13].

2) DWT (Discrete Wavelet Transform)

DWT는 이미지를 주파수 대역별로 분해하여 다해상도 분석을 수행한다. Haar 웨이블릿과 같은 기저 함수를 이용하여 영상을 저주파(LL; Low-Low)와 고주파(LH; Low-High, HL; High-Low, HH; High-High) 대역으로 재귀적으로 분할하며[14], 이 과정에서 공간적 정보와 주파수 정보를 동시에 보존하는 특성을 가진다[11]. 이러한 특성 덕분에 DWT는 이미지 크기 변환이나 해상도 저하 공격에 대해 탁월한 강건성을 보인다. 최근 연구에서는 DWT와 DCT를 결합하여 각 변환의 장점인 스케일링 강건성과 압축 효율성을 동시에 확보하려는 시도가 보고되었다[15],[16].

3) DFT (Discrete Fourier Transform)

DFT는 공간 도메인의 신호를 정현파의 합으로 표현되는 주파수 도메인으로 변환한다. DFT의 크기 성분은 이동에 불변하며, 회전 및 크기 변환에 대해서도 예측 가능한 변형 특성을 가진다. 이는 기하학적 공격에 대한 내성을 확보하는 데 유리하며, 최근 AI(Artificial Intelligence) 생성 콘텐츠 탐지를 위한 워터마킹 기술에서도 주파수 아티팩트 분석에 활용되고 있다[1]. 또한, DCT, DFT, SVD를 모두 통합하여 기하학적 공격과 신호 처리 공격에 동시에 대응하려는 다중 변환 기법도 제안된 바 있다[17].

4) SVD (Singular Value Decomposition)

SVD는 영상을 행렬로 간주하여 대수적으로 분해하는 기법이다. Sun 등[18]의 연구에 따르면, SVD 변환 후의 특이값들은 영상의 기하학적 변형이나 노이즈 추가에도 그 비율이나 순서가 크게 변하지 않는 고유한 관계를 유지한다. 최신 연구들은 SVD의 강건성을 극대화하기 위해 입자 군집 최적화(PSO; Particle Swarm Optimization)[19], 마이크로 유전 알고리즘(Micro-GA; Micro-Genetic Algorithm)[20], Jaya 알고리즘[21]과 같은 최적화 기법을 도입하여 최적의 스케일링 팩터를 탐색하고 있다. 그러나 SVD는 $O(N^3)$ 의 높은 연산 복잡도를 요구하므로 실시간 처리를 위해서는 최적화가 필수적이다.

2-3 화질 평가 지표

워터마킹 알고리즘의 비가시성을 정량적으로 평가하기 위해, 본 연구에서는 널리 통용되는 두 가지 객관적 평가지표인 PSNR(Peak Signal-to-Noise Ratio)과 SSIM(Structural Similarity Index Measure)을 사용한다. PSNR은 픽셀의 절대적 오차인 평균 제곱 오차(MSE; Mean Square Error)만을 반영하므로, 이미지의 구조적 왜곡을 인지하는 인간 시각 특성을 완벽하게 대변하지 못한다[22]. 이에 반해 Wang 등[23]이 제안한 SSIM은 이미지의 휘도, 명암비, 구조 정보를 종합적으로 고려하여 인간이 체감하는 화질을 더욱 정확하게 수치화한다. 최근 연구에서는 FSIM(Feature Similarity Index Measure) 등 다양한 지표가 제안되고 있으나[24], SSIM은 여전히 워터마킹의 비가시성과 강건성 간의 트레이드오프를 평가하는 가장 신뢰성 높은 지표로 활용되고 있다 [25].

1) PSNR

PSNR은 원본 영상과 처리된 영상 간의 픽셀 단위 오차를 기반으로 화질 손실량을 측정하는 가장 일반적인 지표이다. 이는 평균 제곱 오차를 이용하여 식 (1)과 같이 정의된다.

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2 \quad (1)$$

$$PSNR = 10 \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \quad (2)$$

여기서 I 와 K 는 각각 원본 및 워터마킹된 영상, m , n 은 영상의 해상도, MAX_I 는 픽셀이 가질 수 있는 최대값을 의미한다. PSNR 값이 높을수록 원본과의 물리적 차이가 적음을 의미하며, 일반적으로 35dB 이상이면 시각적으로 구분이 어려운 수준으로 간주된다. 그러나 PSNR은 픽셀의 절대적 오차만을 반영하므로, 인간의 시각적 인지 특성을 완벽하게

대변하지 못하는 한계가 있다.

2) SSIM

SSIM은 인간 시각 시스템이 이미지의 구조적 정보 변화에 민감하다는 점에 착안하여 고안된 지표이다. SSIM은 영상의 휘도, 명암비, 그리고 구조의 세 가지 요소를 종합하여 식 (3) 과 같이 산출된다.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (3)$$

여기서 μ 는 평균 휘도, σ 는 표준편차, σ_{xy} 는 공분산을 나타내며, C_1, C_2 는 연산 과정에서 분모가 0이 되는 불안정성을 방지하기 위한 상수이다. SSIM은 -1부터 1까지의 값을 가진다. 1에 가까울수록 원본과 구조적으로 동일함을 의미한다. 일반적인 영상 품질 평가 기준에 따르면, SSIM 0.95 이상은 원본과 시각적 차이를 인지하기 어려운 매우 우수 등급으로, 0.90 이상 0.95 미만은 미세한 차이가 존재하나 허용 가능한 우수 등급으로 분류된다. 따라서 본 연구에서는 SSIM 0.90을 비가시성 만족을 위한 최소 임계값으로 설정한다.

2-4 GPU 아키텍처와 메모리 병목

현대의 GPU는 수천 개의 코어를 활용하여 대규모 데이터를 병렬 처리하는 SIMT(Single Instruction Multiple Threads) 구조를 가진다. GPU 컴퓨팅의 성능을 결정짓는 핵심 요소는 연산 속도보다 데이터 이동 속도, 즉 메모리 대역폭이다[5]. Mei와 Chu[26]는 마이크로벤치마킹을 통해 GPU 메모리 계층의 캐시 구조와 접근 지연시간을 분석하였으며, 전역 메모리 접근 비용이 성능 저하의 주원인임을 규명하였다.

또한, Volkov[27]는 메모리 지연(Latency)을 숨기기 위한 점유율(Occupancy)과 레지스터 사용량의 관계를 정립하였다. 기존의 워터마킹 연구들은 대형 블록(8x8 이상)이나 전역 변환을 수행함에 따라 빈번한 전역 메모리 접근을 유발하였고, 이는 메모리 I/O 지연으로 인한 심각한 성능 저하의 원인이 되었다. 특히 워프(Warp) 내의 스레드들이 연속되지 않은 메모리 주소에 접근할 때 발생하는 비병합 메모리 접근은 유효 대역폭을 급격히 떨어뜨린다. 따라서 고해상도 영상의 실시간 워터마킹을 위해서는 데이터를 레지스터나 공유 메모리와 같은 온칩 고속 메모리에 상주시키고, 전역 메모리 접근을 최소화하는 타일링 기법과 같은 메모리 최적화 전략이 필수적으로 요구된다.

III. 제안하는 GPU 기반 4x4 One-Shot 워터마킹

본 장에서는 고해상도 비디오 및 게임 렌더링 파이프라인과 같은 실시간 환경에 최적화된 4x4 블록 기반 One-Shot 워터마킹 시스템을 제안한다. 제안하는 기법은 GPU의 메모리 계층 구조를 고려한 레지스터 타일링 전략을 통해 오프칩 메모리 접근을 최소화하며, 복잡한 행렬 연산을 근사화한 Implicit SVD를 도입하여 실시간성을 확보한다.

3-1 One-Shot 파이프라인 아키텍처

기존의 워터마킹 시스템은 영상 캡처, CPU 메모리 복사, 워터마킹, 인코딩의 다단계 과정을 거치며, 이 과정에서 PCIe(Peripheral Component Interconnect Express) 버스를 통한 데이터 전송 오버헤드가 발생한다. 반면, 제안하는 One-Shot 파이프라인은 렌더링 결과가 저장된 GPU 텍스처 메모리 상에서 컴퓨트 셰이더(Compute Shader)를 통해 워터마킹을 즉시 수행한다. 전체 프로세스는 단일 커널 내에서 이루어지며, 동기화 비용이 발생하는 공유 메모리를 거치지 않고, 각 스레드가 할당받은 픽셀 데이터를 레지스터에서 변환, 삽입, 역변환까지 일괄 처리하는 구조를 가진다. 이를 통해 외부 메모리 대역폭 의존도를 획기적으로 낮추어 게임이나 비디오 렌더링 성능에 미치는 영향을 최소화한다.

3-2 4x4 레지스터 타일링

GPU 병렬 처리의 성능을 저해하는 가장 큰 요인은 느린 전역 메모리 접근과 뱅크 충돌이다. 기존 8x8 블록 기반 기법은 블록 내 64개의 픽셀 데이터를 처리하기 위해 공유 메모리를 캐시로 활용해야 했으나, 이는 스레드 간 동기화 오버헤드를 유발한다. 본 연구에서는 이를 해결하기 위해 4x4 마이크로 블록 구조를 채택하고, 블록 내 16개의 픽셀 데이터를 GPU 스레드의 레지스터 파일에 직접 매핑하는 레지스터 타일링 기법을 적용하였다. 4x4 블록은 최신 GPU 아키텍처의 레지스터 대역폭 내에서 충분히 수용 가능하므로 로드, 변환, 임베딩, 역변환, 저장을 수행하는 일련의 과정이 온칩에서 수행된다. 이는 오프칩 메모리 접근을 최초 1회의 읽기와 최후 1회의 쓰기로 제한하여 메모리 병목을 근본적으로 해소한다.

3-3 경량화된 주파수 변환 알고리즘

제안하는 파이프라인은 4x4 레지스터 구조에 맞춰 최적화된 변환 알고리즘을 사용한다.

1) Implicit SVD

전통적인 SVD는 식 (4)와 같다.

$$A = U\Sigma V^T \quad (4)$$

수식 4를 분해하기 위해 $O(N^3)$ 의 복잡도를 가지며 GPU에서 분기 발산을 유발한다. 본 연구에서는 워터마킹에 필요한 특이값 Σ 과 우측 특이벡터 V 만을 빠르게 추출하기 위해, A 대신 공분산 행렬 $A^T A$ 를 대각화하는 Implicit Jacobi 방식을 제안한다.

구체적으로, 4x4 블록에 대해 U 행렬을 명시적으로 계산하지 않고도 대칭 야코비 회전만으로 고유값 분해를 수행하기 위해 식 (5)의 관계를 이용한다.

$$A^T A = V\Sigma^2 V^T \quad (5)$$

특히, 4회의 고정된 반복만으로 수렴하도록 루프를 풀어서 구현하여 연산 효율을 최적화하였다. 이후 변조된 특이값 행렬을 Σ' 라 할 때, 복원 행렬 M 은 식 (6)과 같이 정의된다.

$$M = V(\Sigma' \Sigma^{-1}) V^T \quad (6)$$

최종적으로 M 을 원본 픽셀 행렬 A 에 우측 곱셈하는 방식을 통해, 기존의 복잡한 재구성 과정을 $O(N^2)$ 복잡도로 단축하였다.

2) 행렬 기반 고속 DCT/DWT

DCT와 DWT는 4x4 블록의 직교 변환 특성을 활용하여 행렬 곱셈 형태로 최적화하였다. 본 연구에서는 루프 오버헤드를 제거하기 위해 4x4 변환 행렬을 레지스터에 상수로 적재하고 행렬 연산을 수행한다.

- **DCT:** JPEG 표준과 호환되는 DCT-II 기반의 변환 행렬을 사용하며, 부동소수점 연산 정밀도를 유지하면서도 GPU의 FMA(Fused Multiply-Add) 파이프라인을 최대한 활용할도록 곱셈 연산을 풀어서 구현하였다.

- **DWT:** Haar 웨이블릿 기저 함수를 4x4 행렬 형태로 재구성하여 적용한다. 이는 다해상도 분석을 단일 행렬 곱셈으로 처리하여, 공간적 특징과 주파수 특징을 동시에 추출한다.

3) 최적화된 4x4 DFT

일반적인 FFT(Fast Fourier Transform)는 버터플라이 연산 과정에서 동적 인덱싱과 비트 반전을 요구하여 분기 예측 실패를 유발할 수 있다. 본 제안 기법은 4x4 블록 크기에 특화된 Radix-4 알고리즘을 수식 형태로 전개하여 구현하였다. 복소수 연산을 위해 vec2 레지스터를 활용하며, 실수부와 허수부의 교차 연산을 GPU의 FMA 명령어로 최적화하여 단일 사이클 내에 처리한다. 이는 공유 메모리를 경유하지 않고 레지스터 내에서 즉시 변환을 수행하므로 메모리 지연 시간을 획기적으로 단축시킨다.

3-4 반복 부호(Repetition Code)

4x4 블록은 8x8 블록에 비해 하나의 블록이 담을 수 있는 정보량이 적다. 이를 보완하고 기하학적 공격에 대한 내성을 확보하기 위해, 본 시스템은 반복 부호 기법을 적용한다. 워터마크 시그널은 $M \times M$ 크기의 타일 단위로 이미지 전체에 반복적으로 타일링되어 삽입된다. 추출 과정에서는 손상된 블록들이 있더라도, 살아남은 블록들의 정보를 취합하여 다수결 투표 또는 평균값 필터링을 통해 원본 서명을 복원한다. 이를 통해 일부 영역이 잘려나가는 Crop 공격이나 국소적인 Salt&Pepper 공격에 대해 강력한 복원력을 제공한다. 하지만 반복 횟수가 늘어날수록 강건성은 높아지지만 삽입 가능한 총 데이터 양은 반비례하여 줄어드는 트레이드 오프가 존재한다. 본 연구에서는 강건성 확보를 최우선으로 하여 설계하였다.

3-5 워터마크 삽입 및 추출 알고리즘

1) 적응적 하이브리드 삽입

렌더링 된 이미지의 주파수 특성에 따라 삽입 강도를 조절하여 비가시성과 강건성을 동시에 확보하기 위해, 본 연구에서는 가법과 승법이 결합된 하이브리드 방식을 제안한다. 변환된 주파수 계수의 값, V 가 0에 가까운 평탄한 영역에서는 고정된 강도로 정보를 삽입하여 데이터 소실을 방지하고, 텍스처가 복잡한 영역에서는 계수의 크기에 비례하여 강도를 조절함으로써 시각적 왜곡을 은폐한다. 이러한 적응적 삽입 수식은 모든 변환 기법에 공통적으로 식 (7)과 같이 적용된다.

$$V' = \begin{cases} V + \alpha \cdot w & \text{if } |V| < \epsilon \text{ (Additive)} \\ V \cdot (1 + \alpha \cdot w) & \text{otherwise (Multiplicative)} \end{cases} \quad (7)$$

여기서 V 와 V' 는 각각 변환 전후의 계수, w 는 워터마크 비트, α 는 삽입 강도 계수, ϵ 은 0으로 간주할 임계값(10^{-5})이다. 본 시스템은 이 공통 수식을 기반으로, 각 변환 알고리즘의 주파수적 특성에 맞춰 다음과 같이 타겟 계수를 차별화하여 적용한다.

- **SVD:** 4x4 블록의 에너지가 집중되는 최대 특이값을 타겟으로 선정한다. 기하학적 변형 시 특이값들의 비율은 유지되는 성질을 이용하여, 최대 특이값에 워터마크를 삽입하고 나머지 특이값들과의 비율을 변경함으로써 강건성을 확보한다. 변조된 비율 행렬은 픽셀 공간에 다시 투영되어 밝기 변화로 적용된다.

- **DCT:** 특정 주파수 대역의 소실에 대비하기 위해, 워터마크 시그널을 중간 주파수 대역인 (1,2), (2,1), (1,1)의 세 지점에 분산하여 삽입한다.

- **DWT:** 인간 시각에 덜 민감하면서도 압축 효율이 높은 LH 대역의 (0,1) 계수를 타겟으로 한다.

• **DFT**: 회전 공격에 강한 내성을 갖기 위해 위상 정보를 보존한 채, 중간 주파수 대역의 크기 성분을 V 로 설정하여 변조한다.

2) 색차 보존 역변환

주파수 도메인에서 변경된 휘도 Y 를 다시 RGB 색상 공간으로 변환할 때, 단순한 덧셈 연산은 색상과 채도의 왜곡을 유발할 수 있다. 이를 방지하기 위해 본 파이프라인은 변경된 휘도 비율을 원본 RGB 픽셀에 곱하는 방식을 사용하여, 픽셀의 밝기만 변경하고 고유의 색상 정보는 보존한다. 최종 복원식은 식 (8)과 같다.

$$RGB_{out} = RGB_{in} \cdot \frac{Y_{watermarked}}{Y_{original}} \tag{8}$$

이 방식은 워터마킹으로 인한 픽셀 값 변화가 원본 색상 벡터의 방향을 유지하도록 강제하므로, 시각적으로 매우 자연스러운 결과를 보장한다.

IV. 실험 환경 및 결과 분석

본 장에서는 제안하는 컴퓨터 셰이더 기반의 One-Shot 워터마킹 프레임워크의 성능을 검증한다. 실험은 크게 시스템 효율성(속도), 비가시성(화질), 그리고 강건성 평가로 구성되며, 특히 기존 8x8 블록 및 전역 변환 방식 대비 제안하는 4x4 레지스터 최적화 기법의 유효성을 정량적으로 분석하는데 중점을 둔다.

4-1 실험 환경 및 구성

표 1. 실험 하드웨어 사양 및 소프트웨어 환경

Table 1. Experimental hardware specifications and software environment

Component	Specification / Version Information
CPU	I7-13700
GPU	RTX 4060
RAM	DDR4 32GB
Storage Device	SSD
Operating System	Window 11
Graphics API	OpenGL 4.3
Development Tools	Visual Studio 2026 Community

1) 하드웨어 및 소프트웨어 환경

본 연구의 모든 실험은 실시간 렌더링 파이프라인과 유사한 환경을 구축하기 위해 고성능 데스크톱 PC 환경에서 수행되었다. 워터마킹 알고리즘의 구현 및 성능 측정은 OpenGL 4.3 규격의 컴퓨터 셰이더를 사용하여 GPU 가속을 통해 이루어

졌다. 상세한 하드웨어 및 소프트웨어 사양은 표 1과 같다.

2) 비교군 설정 및 구현

객관적인 성능 검증을 위해 본 논문에서 제안하는 최적화 기법과 기존의 표준 기법을 동일한 프레임워크 내에서 구현하였다.

• **제안 기법**: 4x4 마이크로 블록 기반의 레지스터 타일링을 적용한 Implicit SVD, 행렬 기반 고속 DCT/DWT, 최적화된 DFT를 구현하였으며, 반복 부호를 통합하여 신호 수용력을 보완하였다.

• **비교군**: 대조군으로는 저자들의 선행 연구[6]에서 실시간 화질 성능이 검증된 8x8 블록 기반의 DCT 및 DWT 알고리즘을 채택하였으며, 추가적으로 Block SVD (Explicit 4x4)와 Cooley-Tukey 알고리즘 기반의 전역 변환 DFT를 구현하여 제안 기법과의 성능 차이를 다각도로 비교하였다.

3) 실험 변인 및 절차

실험은 FHD(1920x1080)와 4K UHD(3840x2160) 해상도에서 진행되었으며, 삽입 강도인 Alpha는 0.1~1.5 범위에서 조절하되 비가시성 비교 시에는 Alpha가 1일 때, PSNR 40dB 수준으로 통제하였다. 성능 평가는 glQueryCounter를 이용한 GPU 커널 실행 시간(ms), 원본 대비 PSNR 및 SSIM, 그리고 공격 후 추출된 비트 오류율(BER; Bit Error Rate)을 측정하였다. 공격 시뮬레이션은 GPU 메모리 상에서 셰이더를 통해 즉시 수행되었으며, 공격 유형은 기하학적 공격(Crop, Rotation, Resize) 및 신호 처리 공격(Noise, Quantization, Gamma, Brightness)을 포함한다. 실험에 쓰인 원본 이미지와 공격받은 이미지들은 그림 1과 같다.

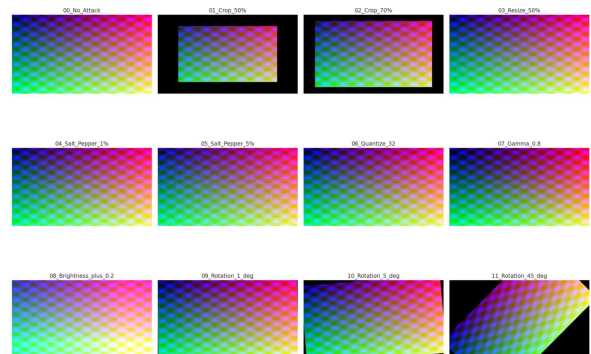


그림 1. 공격받은 이미지들
Fig. 1. Attacked images

4-2 연산 성능 및 시스템 효율성 분석

제안하는 4x4 블록 기반 One-Shot 아키텍처가 실제 GPU 연산 속도에 미치는 영향을 분석하기 위해, FHD(1920x1080)와 4K(3840x2160) 해상도에서 각 알고

리즘의 GPU 커널 실행 시간을 측정하였다. FHD 해상도에서의 실험 결과 그래프는 그림 2와 같으며 4K 해상도에서의 실험 결과 그래프는 그림 3과 같다.

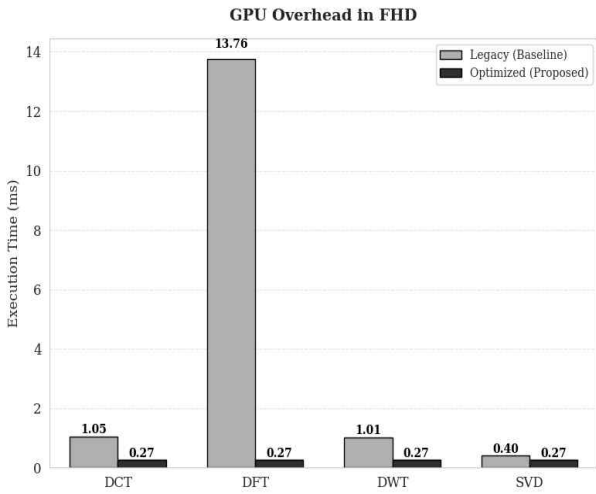


그림 2. FHD 화질에서 각 기법들의 GPU 처리 시간
Fig. 2. GPU processing time for each method at FHD resolution

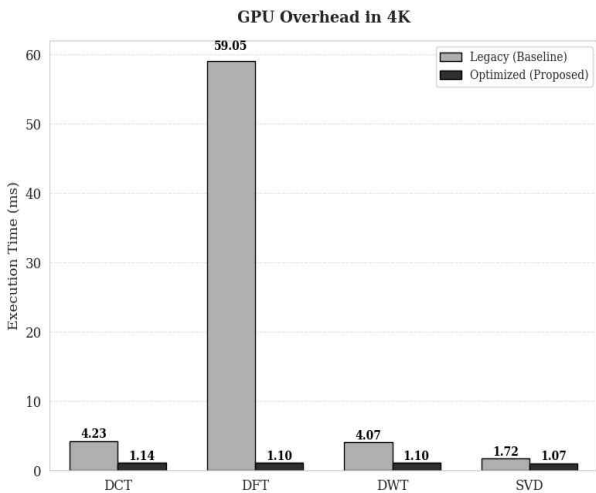


그림 3. 4K 화질에서 각 기법들의 GPU 처리 시간
Fig. 3. GPU processing time for each method at 4K resolution

1) 4x4 블록 기반 레지스터 최적화에 따른 가속 효과

선행 연구[6]를 포함하여 기존 연구에서 주로 사용된 8x8 블록 기반의 DCT 및 DWT, 그리고 전역 변환 기반의 DFT는 GPU의 공유 메모리 한계를 초과하거나 잦은 전역 메모리 접근을 유발하여 병렬 처리 효율을 저하시키는 원인이 된다. 이에 반해, 본 연구에서 제안하는 4x4 마이크로 블록 구조는 블록 내의 모든 연산 데이터를 GPU 스레드의 레지스터 공간에 할당함으로써 메모리 I/O 지연을 획기적으로 단축시켰다.

FHD 해상도에서 DCT는 기존 8x8 구현의 1.05ms에서 제안하는 4x4 구현을 통해 0.27ms로 약 3.85배의 속도 향상을 보였으며, DWT 또한 1.01ms에서 0.27ms로 약 3.82배 단축되었다. DFT의 경우 13.76ms에서 0.27ms로 약 51.7배의 성능 향상을 기록하였다.

4K 해상도에서도 이러한 가속 효과는 일관되게 나타났다. DCT는 4.23ms에서 1.14ms로, DWT는 4.07ms에서 1.10ms로 단축되어 약 3.7배의 성능 향상을 유지하였다. 특히 DFT는 전역 변환의 높은 연산 복잡도로 인해 59.05ms가 소요되었으나, 제안하는 4x4 블록 기반 처리를 통해 1.10ms로 단축, 약 53.8배의 압도적인 가속 성능을 보였다.

2) Implicit SVD 알고리즘의 연산 효율성

SVD 알고리즘의 경우, 동일한 4x4 블록 환경에서 행렬 분해 방식에 따른 성능 차이를 검증하였다. 베이스 라인인 Explicit SVD(4x4)는 행렬을 직접 분해하는 정석적인 과정을 거치며 FHD 기준 0.40ms, 4K 기준 1.72ms가 소요되었다. 반면 제안하는 Implicit SVD 기법은 FHD 0.27ms, 4K 1.07ms를 기록하였다. 이는 해상도가 증가할수록 Explicit 방식의 연산 부담이 가중되는 반면, Implicit 방식은 경량화된 연산 구조를 통해 더욱 효율적인 처리가 가능함을 시사한다. 4K 해상도 기준 Implicit SVD는 기존 SVD 대비 약 1.61배 빠른 속도를 보였으며, 이는 제안하는 알고리즘 최적화가 고 해상도 환경에서 더욱 유의미한 성능 향상을 제공함을 입증한다. 결과적으로 제안하는 모든 4x4 기반 최적화 알고리즘은 4K 해상도에서도 1.2ms 미만의 처리 속도를 달성하였다.

4-3 비가시성 분석

본 절에서는 제안하는 워터마킹 기법의 비가시성을 객관적으로 검증하기 위해, 모든 비교 알고리즘의 PSNR을 워터마크 삽입 강도 1 기준 약 40dB 수준으로 통제된 상태에서 워터마크 삽입 강도를 조절하고 SSIM 변화를 분석하였다. 이는 물리적인 오차인 PSNR를 동일하게 맞추었을 때, 인간의 시각적 만족도인 SSIM가 어떻게 달라지는지를 비교함으로써 알고리즘의 화질 보존 효율성을 평가하기 위함이다. 4K 해상도에서 0부터 1.5까지의 삽입 강도를 가졌을 때의 PSNR 그래프는 그림 4과 SSIM의 그래프는 그림 5와 같다. 또한 FHD 해상도에서 0부터 1.5까지의 삽입 강도를 가졌을 때의 PSNR 그래프는 그림 6, SSIM의 그래프는 그림 7과 같다.

1) PSNR 대비 SSIM 효율성 및 알고리즘별 계층 구조

실험 결과, 모든 알고리즘의 PSNR을 약 40dB로 동일하게 통제했음에도 불구하고, 인간이 체감하는 구조적 화질인 SSIM은 알고리즘의 동작 원리에 따라 뚜렷한 차이를 보였다.

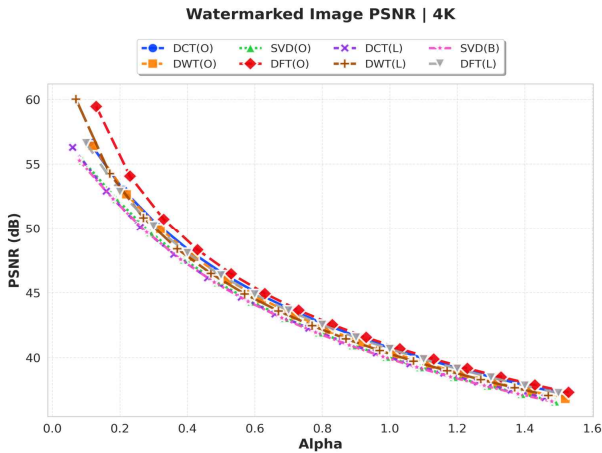


그림 4. 4K 해상도에서 삽입 강도에 따른 PSNR 변화
Fig. 4. PSNR variation according to embedding strength (alpha) at 4K resolution

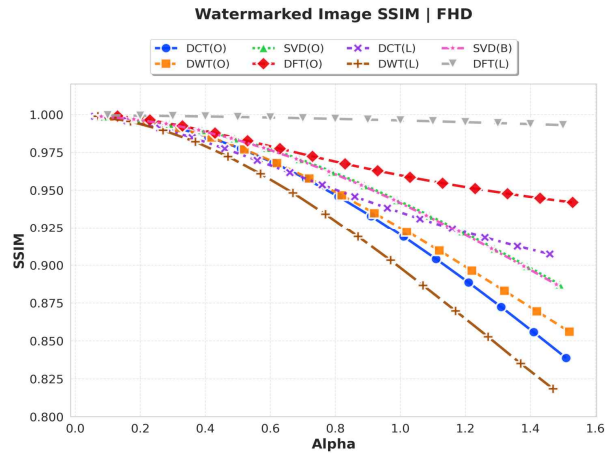


그림 7. FHD 해상도에서 삽입 강도에 따른 SSIM 변화
Fig. 7. SSIM variation according to embedding strength (alpha) at FHD resolution

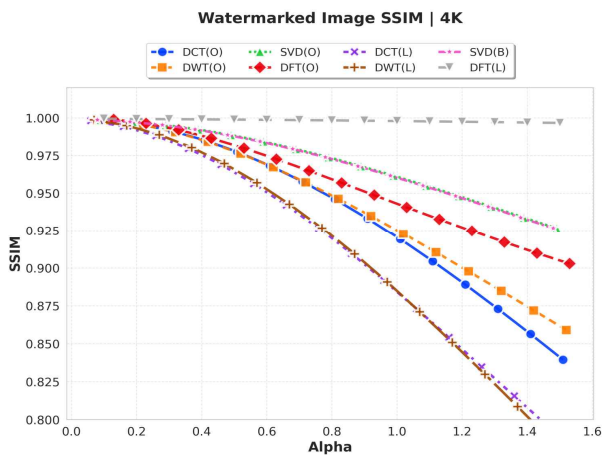


그림 5. 4K 해상도에서 삽입 강도에 따른 SSIM 변화
Fig. 5. SSIM variation according to embedding strength (alpha) at 4K resolution

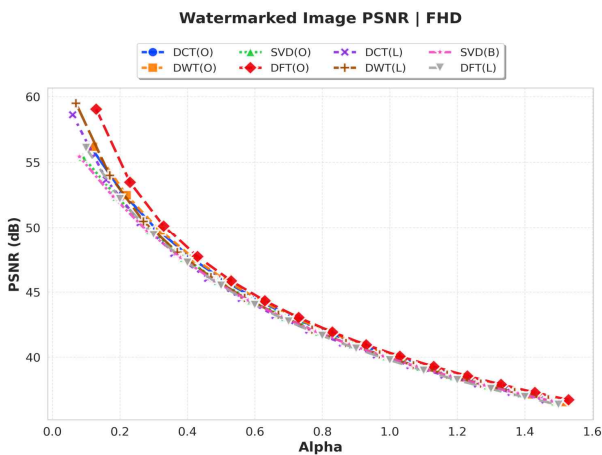


그림 6. FHD 해상도에서 삽입 강도(alpha)에 따른 PSNR 변화
Fig. 6. PSNR variation according to embedding strength (alpha) at FHD Resolution

DFT와 SVD는 SSIM을 0.96로 유지했다. DFT의 경우, 주파수 도메인 전체 혹은 블록 전체에 워터마크 에너지를 얇게 분산시키는 확산 대역 특성 덕분에, 픽셀 간의 급격한 변화를 유발하지 않아 구조적 유사도가 극도로 높게 유지되었다. SVD의 경우, 픽셀 값을 직접 수정하는 것이 아니라 이미지의 대수적 특성인 특이값을 미세 조정하기 때문에, 픽셀 간의 상관관계와 텍스처 구조가 온전하게 보존되었다. 주파수 계수를 양자화하여 조작하는 DWT와 DCT는 0.90~0.92 수준의 준수한 SSIM을 기록하였다.

4K DWT의 경우, 기존 8x8 Legacy 방식(SSIM 0.871)은 블록 경계면이 드러나며 SSIM이 급락했으나, 제안하는 Optimized DWT(SSIM 0.923)는 동일 PSNR에서도 더 높은 SSIM을 기록했다. 이는 마이크로 블록 구조가 인간 시각이 민감한 저주파 및 에지 영역의 왜곡을 최소화했음을 시사한다. 모든 기법이 PSNR 40dB 기준에서 SSIM 0.90 이상의 고품질 비가시성을 만족함을 확인하였다.

4-4 강건성 및 공격 유형별 성능 분석

본 절에서는 Crop, Resize, Rotation과 같은 기하학적 변환 공격과 Salt&Pepper, Quantize, Gamma, Brightness와 같은 신호 처리와 같은 7가지 공격 시나리오에 대한 제안 알고리즘의 내성을 심층 분석한다. 분석 결과는 신호 처리 공격과 기하학적 공격으로 나누어 기술한다.

해상도 및 기법에 따른 공격별 비트 오류율(BER) 측정 결과는 다음과 같다. 4K 해상도에서의 기존 기법인 Legacy과 제안 기법인 Optimized의 BER 성능 비교는 각각 그림 8과 그림 9와 같으며, FHD 해상도에서의 실험 결과 그래프는 그림 10과 그림 11과 같다.

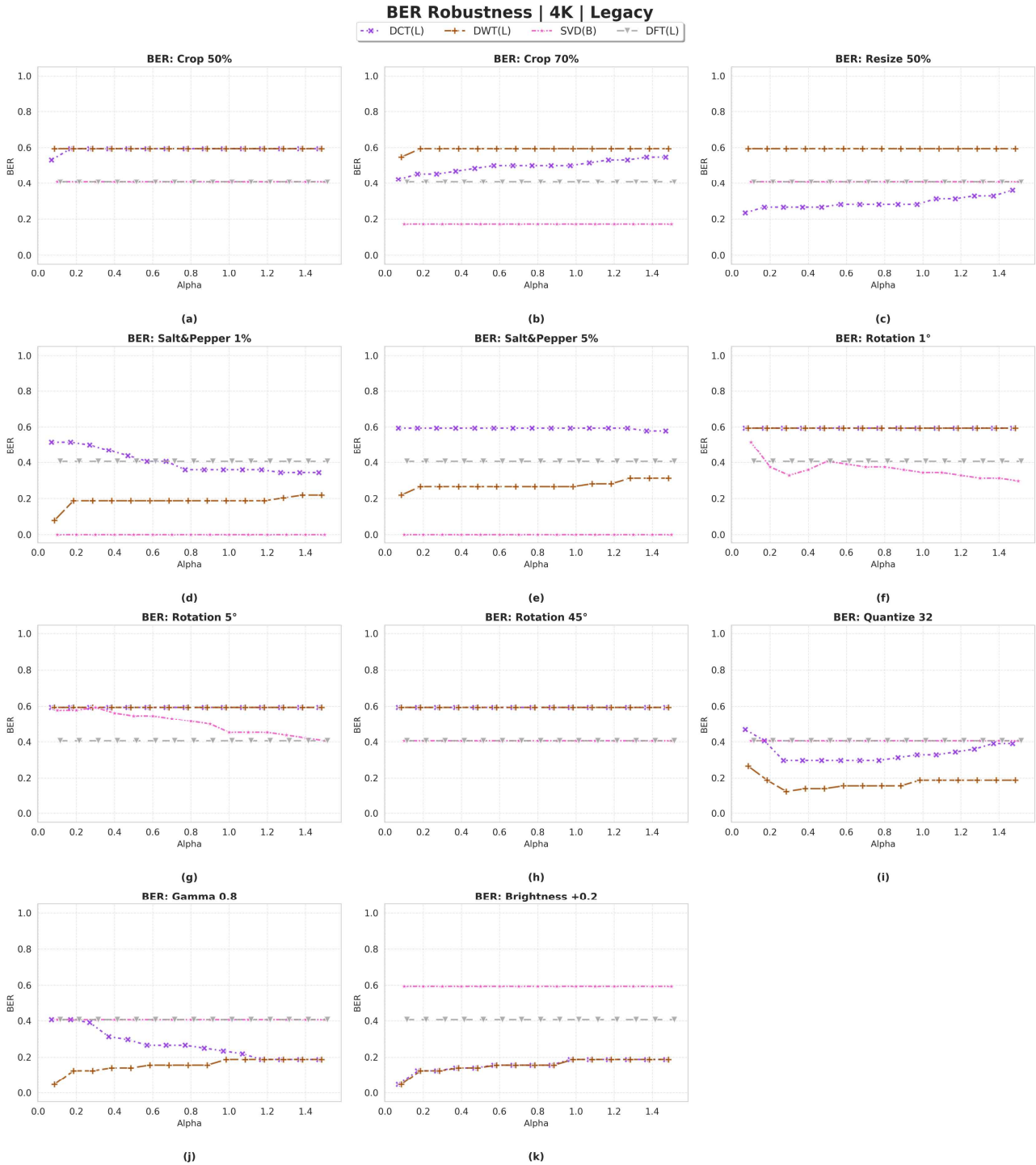


그림 8. 4K 해상도에서 기존 기법의 공격 유형별 BER 성능 비교: (a) Crop (50%), (b) Crop (70%), (c) Resize (50%), (d) Salt & Pepper 1%, (e) Salt & Pepper 5%, (f) Rotation (1°), (g) Rotation (5°), (h) Rotation (45°), (i) Quantization (32), (j) Gamma (0.8), (k) Brightness (+0.2)

Fig. 8. BER performance comparison of legacy methods under various attacks at 4K resolutions: (a) Crop (50%), (b) Crop (70%), (c) Resize (50%), (d) Salt & Pepper 1%, (e) Salt & Pepper 5%, (f) Rotation (1°), (g) Rotation (5°), (h) Rotation (45°), (i) Quantization (32), (j) Gamma (0.8), (k) Brightness (+0.2)

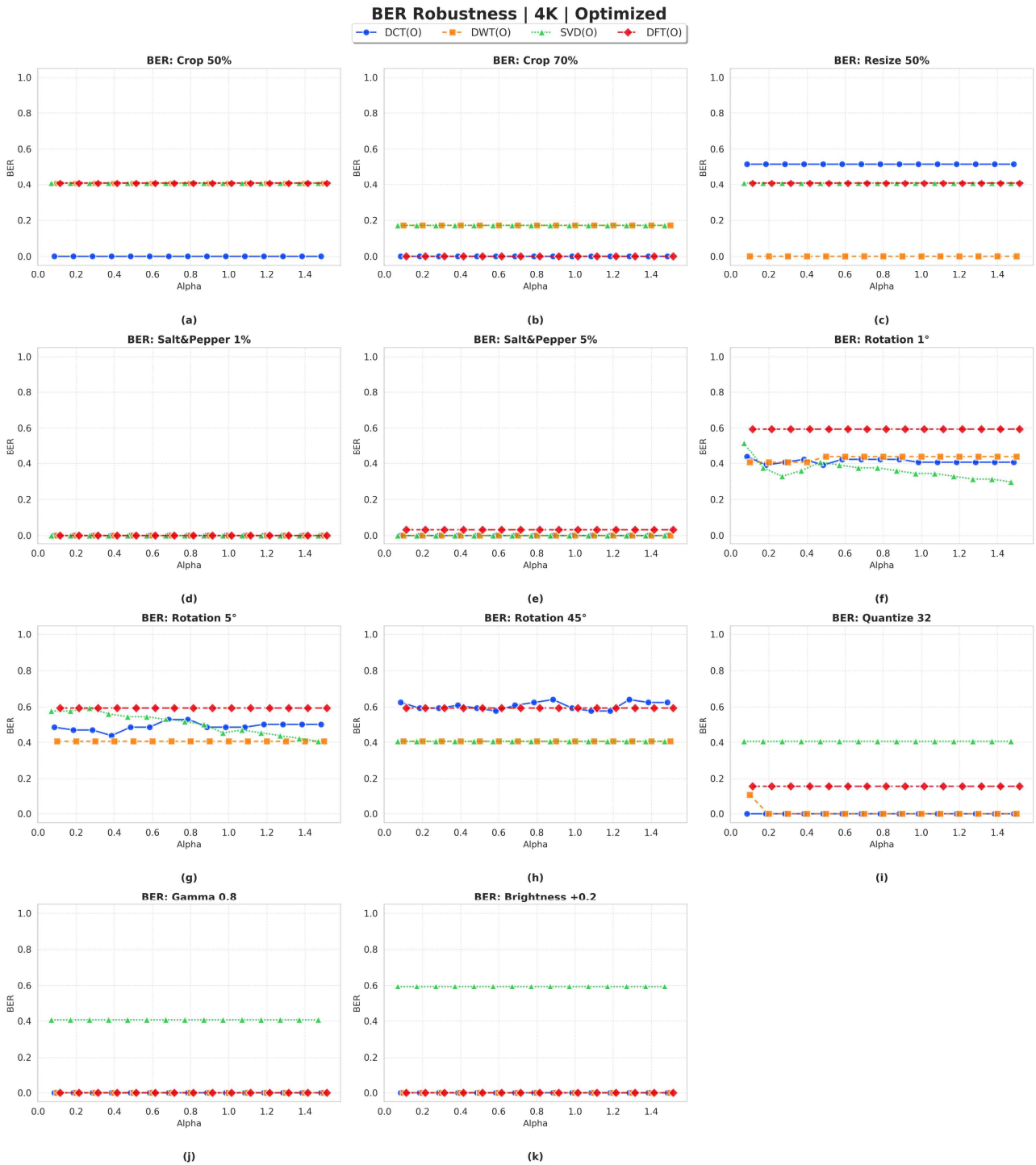


그림 9. 4K 해상도에서 제안 기법의 공격 유형별 BER 성능 비교: (a) Crop (50%), (b) Crop (70%), (c) Resize (50%), (d) Salt & Pepper 1%, (e) Salt & Pepper 5%, (f) Rotation (1°), (g) Rotation (5°), (h) Rotation (45°), (i) Quantization (32), (j) Gamma (0.8), (k) Brightness (+0.2)

Fig. 9. BER performance comparison of optimized methods under various attacks at 4K resolutions: (a) Crop (50%), (b) Crop (70%), (c) Resize (50%), (d) Salt & Pepper 1%, (e) Salt & Pepper 5%, (f) Rotation (1°), (g) Rotation (5°), (h) Rotation (45°), (i) Quantization (32), (j) Gamma (0.8), (k) Brightness (+0.2)

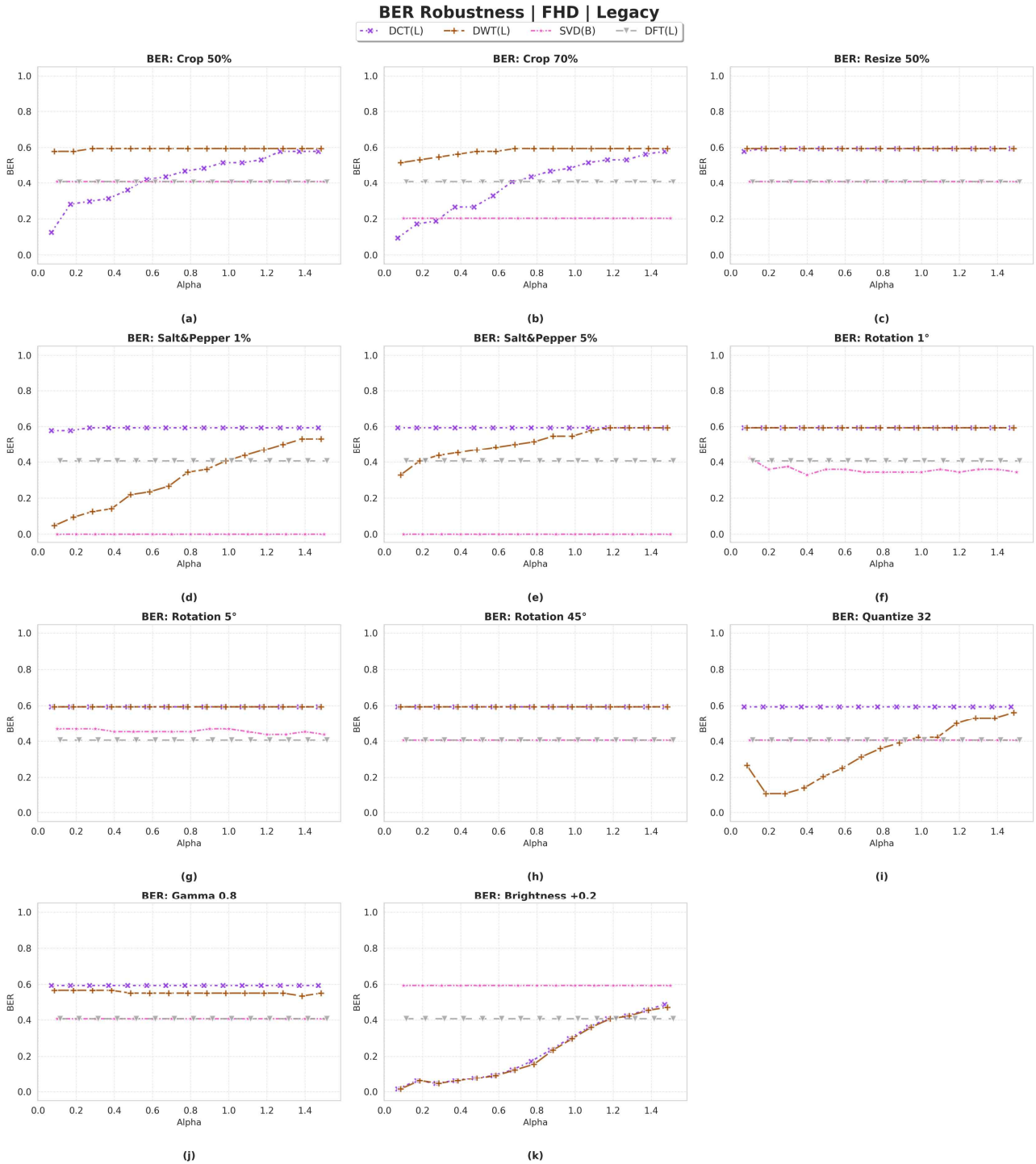


그림 10. FHD 해상도에서 기존 기법의 공격 유형별 BER 성능 비교: (a) Crop (50%), (b) Crop (70%), (c) Resize (50%), (d) Salt & Pepper 1%, (e) Salt & Pepper 5%, (f) Rotation (1°), (g) Rotation (5°), (h) Rotation (45°), (i) Quantization (32), (j) Gamma (0.8), (k) Brightness (+0.2)

Fig. 10. BER performance comparison of legacy methods under various attacks at FHD resolutions: (a) Crop (50%), (b) Crop (70%), (c) Resize (50%), (d) Salt & Pepper 1%, (e) Salt & Pepper 5%, (f) Rotation (1°), (g) Rotation (5°), (h) Rotation (45°), (i) Quantization (32), (j) Gamma (0.8), (k) Brightness (+0.2)

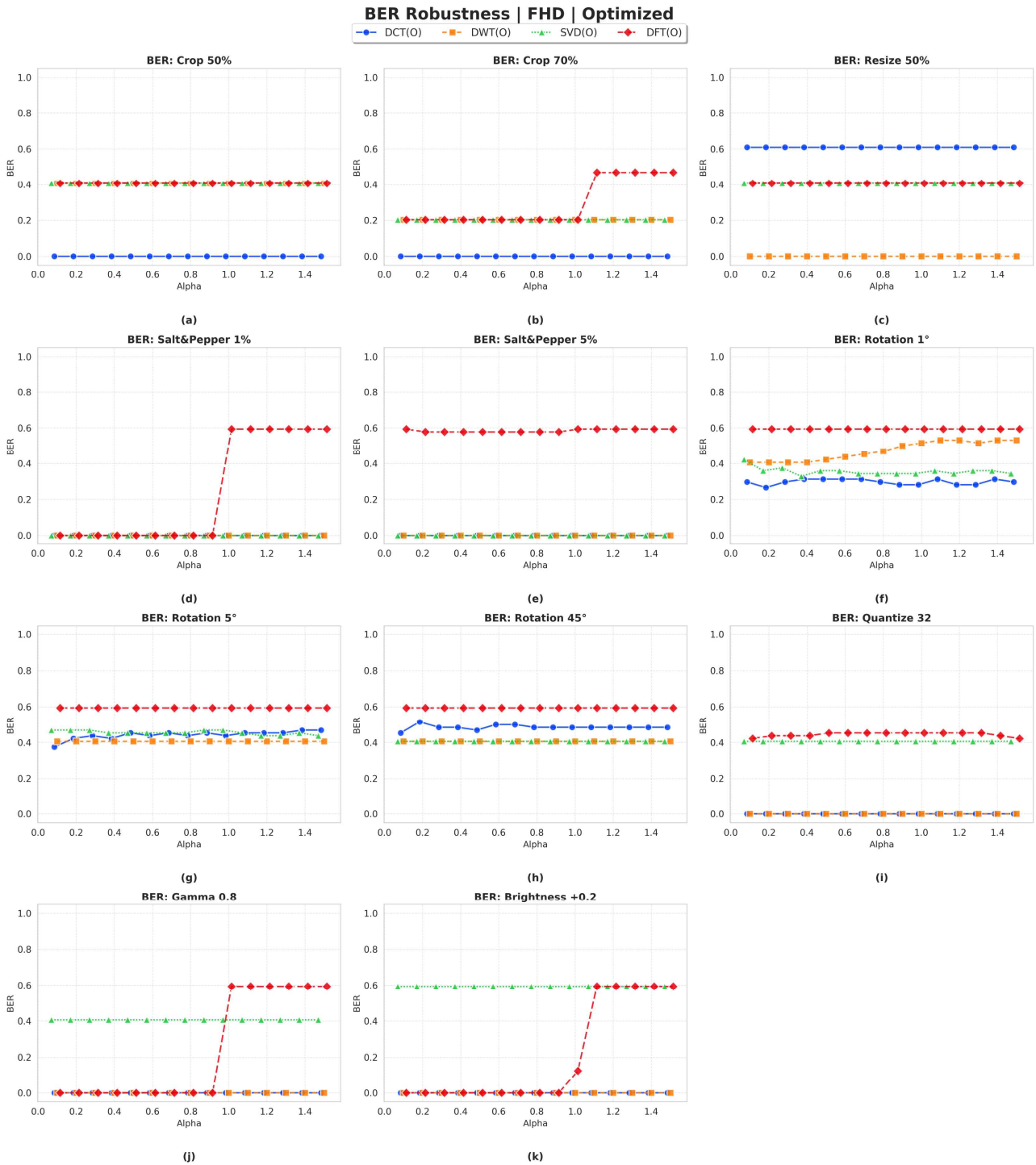


그림 11. FHD 해상도에서 제안 기법의 공격 유형별 BER 성능 비교: (a) Crop (50%), (b) Crop (70%), (c) Resize (50%), (d) Salt & Pepper 1%, (e) Salt & Pepper 5%, (f) Rotation (1°), (g) Rotation (5°), (h) Rotation (45°), (i) Quantization (32), (j) Gamma (0.8), (k) Brightness (+0.2)

Fig. 11. BER performance comparison of optimized methods under various attacks at FHD resolutions: (a) Crop (50%), (b) Crop (70%), (c) Resize (50%), (d) Salt & Pepper 1%, (e) Salt & Pepper 5%, (f) Rotation (1°), (g) Rotation (5°), (h) Rotation (45°), (i) Quantization (32), (j) Gamma (0.8), (k) Brightness (+0.2)

1) 신호 처리 및 화질 변환 공격

먼저 노이즈 및 양자화 공격에 대한 분석 결과, 제안하는 알고리즘들은 전반적으로 우수한 내성을 보였으나 특정 조건에서 명확한 한계점이 드러났다. 그림 9와 그림 11의 (d), (e)에 나타난 Salt & Pepper 공격(1%, 5%)과 (i)의 Quantization(32단계) 공격 시나리오에서, Optimized DCT와 DWT 기법은 Alpha 0.1 수준의 낮은 강도에서도 BER 0.00을 기록하며 완벽한 복원력을 입증했다. 이는 고주파 성분을 주로 손상시키는 해당 공격들의 특성상, 워터마크가 삽입된 중저주파 대역의 정보가 온전히 보존되었기 때문이다. 반면, FHD 해상도 결과를 나타내는 그림 11의 (d)와 (e)에서 DFT 기법은 Alpha 0.9까지 안정적인 성능을 유지하다가 Alpha 1.0 이상 구간에서 BER이 0.59로 급격히 치솟는 '성능 절벽' 현상을 보였다. 이는 과도한 워터마크 에너지가 픽셀의 표현 범위를 초과하며 발생한 클리핑 왜곡이 노이즈와 결합되어 디코더의 동기화를 무너뜨린 것으로 분석된다.

다음으로 밝기 및 감마 보정 공격에서는 알고리즘의 구조적 특성에 따라 성능이 극명하게 갈렸다. 그림 9와 그림 11의 (j) Gamma(0.8) 및 (k) Brightness(+0.2) 공격 결과를 살펴보면, DCT와 DWT는 차분 방식이나 고주파 계수 활용을 통해 전역적인 픽셀 값 변화에 영향을 받지 않고 BER 0.00을 유지함을 확인할 수 있다. 그러나 동일한 그래프 (j)와 (k)에서 SVD 기법들은 BER 0.40~0.59를 기록하며 사실상 정보 복원에 실패하였다. 이는 이미지 전체의 픽셀 값이 비선형적으로 이동함에 따라 특이값의 비율이 왜곡되고, 결과적으로 워터마크 추출을 위한 임계값 설정이 무력화되었기 때문이다. 따라서 SVD 기법을 실제 응용에 적용하기 위해서는 밝기 불변 전처리 과정이 선행되어야 함을 시사한다.

2) 기하학적 공격

기하학적 변환 중 그림 9와 그림 11의 (c) Resize (50%) 공격 결과를 살펴보면, Optimized DWT 기법의 독보적인 강건성이 확인된다. 픽셀 정보를 강제로 소실시키는 리사이즈 과정에서 DCT, DFT, SVD 등 대부분의 기법이 BER 0.3~0.5 수준의 높은 오류율을 보인 반면, DWT는 유일하게 모든 해상도에서 BER 0.00을 기록하며 생존하였다. 이는 웨이블릿 변환의 다해상도 분석 특성 덕분에, 저주파수 대역에 삽입된 워터마크 정보가 이미지 크기 축소 과정에서도 견고하게 유지된 결과이다.

하지만 Rotation 및 Crop 공격은 블록 기반 워터마크 기술의 태생적 한계를 드러냈다. 그림 8부터 그림 11까지의 (f), (g), (h) 그래프에서 나타나듯, 미세한 (f) 1도 회전만으로도 픽셀 그리드(Grid)가 어긋나면서 본 연구의 모든 4x4 블록 알고리즘이 BER 0.4~0.6 수준으로 붕괴되는 취약점을 보였다. (a) Crop 50%과 (b) Crop 70% 공격 또한 대부분의 기법에서 높은 오류율을 보였으나, 그림 9(4K Optimized)의 결과에서 Optimized DCT 기법은 4K 해상도의 공간적 여유와 반복 부호의 효과로 인해 일부 구간에서 정보 복원이 가능

함을 확인했다. 이는 전체 이미지가 손상되더라도 국소적인 블록 패턴이 보존된다면 반복 코드를 통해 전체 정보를 재구성할 수 있음을 의미한다.

V. 결 론

5-1 연구 요약 및 기여

본 연구는 저자들의 선행 연구[6]에서 상용 게임 엔진인 Unity를 통해 확인했던 워터마크의 적용 가능성을 넘어, 네이티브 OpenGL 환경에서의 정밀한 오버헤드 측정과 강건성 검증을 통해 고해상도 비디오 및 실시간 렌더링 환경에 최적화된 GPU 기반 4x4 블록 워터마크 파이프라인을 완성하였다. 제안 기법의 연산 효율성, 비가시성, 강건성을 종합적으로 검증한 주요 연구 성과는 다음과 같다.

1) 시스템 효율성 극대화

기존 8x8 및 전역 변환 방식의 메모리 병목을 4x4 레지스터 타일링 기법으로 해소하여, FHD 및 4K 해상도에서 60 FPS 실시간 처리를 만족하는 1.2ms 미만의 성능을 달성하였다. 특히 Optimized DFT는 전역 변환 대비 50배 이상의 가속을, Implicit SVD는 근사 연산을 통해 화질 저하 없이 1.6 배의 속도 향상을 입증하였다.

2) 비가시성 효율 역전 규명

PSNR 40dB 통제 환경에서 DFT와 SVD는 SSIM 0.96 이상의 탁월한 구조 보존력을 보였다. 또한 4K Optimized DWT와 Optimized DCT는 마이크로 블록 구조가 시각 민감 영역을 효과적으로 회피함으로써, 기존 8x8 방식보다 높은 SSIM을 기록하였다.

3) 공격 유형별 최적 기법 도출

4x4 DWT는 크기 변환에, 4x4 DCT는 노이즈 및 압축 공격에 강인하여 각각 웹 게시와 스트리밍 환경에 적합함을 확인하였다. 반면, Optimized DFT와 Implicit SVD는 구조적 비가시성은 우수하나 각각 클리핑과 밝기 변화에 대한 취약점이 존재함을 확인하였다.

5-2 한계 및 향후 과제

실험을 통해 확인된 알고리즘별 한계점과 개선 방향을 종합하면, 제안하는 블록 기반 기법들은 4x4의 작은 커널 크기로 인해 회전 및 절단 공격 시 발생하는 픽셀 그리드 불일치에 취약점을 보였다. 이를 극복하기 위해 향후 연구에서는 워터마크 추출 전 DFT 도메인에 삽입된 동기화 템플릿을 통해 왜곡된 각도를 보정하거나, 기하학적 불변 모멘트 기반의 특

정점 매칭을 결합하는 하이브리드 파이프라인에 대한 연구가 요구된다. 아울러 SVD 기법의 경우 연산 효율은 우수하나 단순 밝기 변화에도 성능이 저하되는 민감성을 가지므로, 안정성을 확보하기 위해 밝기 불변 전처리 모듈의 도입이 필수적이다.

또한 본 연구는 통제된 환경에서 알고리즘 고유의 성능을 측정하였으나, 실제 상용 환경에서는 순수 알고리즘의 연산 시간 외에도 GPU 메모리 대역폭 경합 등 다양한 변수가 존재하므로, 고사양 3D 씬 통합 환경에서의 안정성 검증 및 최적화 연구가 후속적으로 수행될 필요가 있다.

감사의 글

본 연구는 문화체육관광부 및 한국콘텐츠진흥원의 2025년도 문화체육관광 연구개발사업으로 수행되었음(연구개발과제명: 게임콘텐츠 저작권 관리 기술 개발 및 글로벌 인재양성, 연구개발과제번호: RS-2024-00396709, 기여율: 50%).

이 논문은 2025학년도 홍익대학교 학술연구진흥비에 의하여 지원되었음.

참고문헌

- [1] L. Cao, "Watermarking for AI Content Detection: A Review on Text, Visual, and Audio Modalities," arXiv:2504.03765, April 2025. <https://doi.org/10.48550/arXiv.2504.03765>
- [2] D. Trick, W. Berchtold, M. Schäfer, and M. Steinebach, "3D Watermarking in the Context of Video Games," in *Proceedings of the 2013 IEEE 15th International Workshop on Multimedia Signal Processing (MMSP)*, Pula, Italy, pp. 418-423, 2013. <https://doi.org/10.1109/MMSP.2013.6659325>
- [3] P. Khoshdani, Watermarking DDS Files, B.S. Thesis, Department of Electrical Engineering, Blekinge Institute of Technology, Karlskrona, Sweden, 2013.
- [4] K. Debattista, K. Bugeja, S. Spina, T. Bashford-Rogers, and V. Hulusic, "Frame Rate vs Resolution: A Subjective Evaluation of Spatiotemporal Perceived Quality Under Varying Computational Budgets," *Computer Graphics Forum*, Vol. 37, No. 1, pp. 363-374, February 2018. <https://doi.org/10.1111/cgf.13302>
- [5] N. Zhang, J. L. Wang, and Y. S. Chen, "Image Parallel Processing Based on GPU," in *Proceedings of the 2010 2nd International Conference on Advanced Computer Control*, Shenyang, China, pp. 367-370, 2010. <https://doi.org/10.1109/ICACC.2010.5486836>
- [6] S.-Y. Nam, J.-Y. Woo, J.-M. Park, and H.-Y. Kim, "Comparative Analysis of Image Quality Performance for Watermarking in Real-Time Rendering Environments," *Journal of Digital Contents Society*, Vol. 26, No. 7, pp. 1881-1893, July 2025. <https://doi.org/10.9728/dcs.2025.26.7.1881>
- [7] M. Kutter, F. D. Jordan, and F. Bossen, "Digital Signature of Color Images Using Amplitude Modulation," in *Proceedings of SPIE 3022, Storage and Retrieval for Image and Video Databases V*, San Jose: CA, pp. 518-526, February 1997. <https://doi.org/10.1117/12.263442>
- [8] S. Sur, R. G. Roy, and S. Chakraborty, "Digital Watermarking: A Technical Overview," *IOSR Journal of Electronics and Communication Engineering*, Vol. 10, No. 4, pp. 34-38, July 2015. <https://doi.org/10.9790/2834-10413438>
- [9] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoan, "Secure Spread Spectrum Watermarking for Multimedia," *IEEE Transactions on Image Processing*, Vol. 6, No. 12, pp. 1673-1687, December 1997. <https://doi.org/10.1109/83.650120>
- [10] J. F. Delaigle, C. DeVleeschouwer, B. Macq, and L. Langendijk "Human Visual System Features Enabling Watermarking," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, Lausanne, Switzerland, pp. 489-492, 2002. <https://doi.org/10.1109/ICME.2002.1035653>
- [11] A. Katharotiya, S. Patel, and M. Goyani, "Comparative Analysis between DCT & DWT Techniques of Image Compression," *Journal of Information Engineering and Applications*, Vol. 1, No. 2, pp. 9-17, 2011.
- [12] S. D. Lin, S.-C. Shie, and J. Y. Guo, "Improving the Robustness of DCT-Based Image Watermarking against JPEG Compression," *Computer Standards & Interfaces*, Vol. 32, No. 1-2, pp. 54-60, January 2010. <https://doi.org/10.1016/j.csi.2009.06.004>
- [13] S. Amirgholipour and A. Sharifi, "A Pre-Filtering Method to Improve Watermark Detection Rate in DCT Based Watermarking," *The International Arab Journal of Information Technology*, Vol. 11, No. 2, pp. 178-185, March 2014.
- [14] P. Porwik and A. Lisowska, "The Haar-Wavelet Transform in Digital Image Processing: Its Status and Achievements," *Machine Graphics & Vision*, Vol. 13, No. 1, pp. 79-97, 2004.
- [15] V. Kumar and D. Kumar, "Performance Evaluation of DWT Based Image Steganography," in *Proceedings of the 2010 IEEE 2nd International Advance Computing Conference (IACC)*, Patiala, India, pp. 223-228, February 2010. <https://doi.org/10.1109/IADCC.2010.5423005>

- [16] K. Deb, M. S. Al-Seraj, M. M. Hoque, and M. I. H. Sarkar, "Combined DWT-DCT Based Digital Image Watermarking Technique for Copyright Protection," in *Proceedings of the 2012 7th International Conference on Electrical and Computer Engineering (ICECE)*, Dhaka, Bangladesh, pp. 458-461, December 2012. <https://doi.org/10.1109/ICECE.2012.6471586>
- [17] J. Varghese, O. B. Hussain, S. Subash, and T. Abdul Razak, "An Effective Digital Image Watermarking Scheme Incorporating DCT, DFT and SVD Transformations," *PeerJ Computer Science*, Vol. 9, e1427, July 2023. <https://doi.org/10.7717/peerj-cs.1427>
- [18] X. Sun, J. Liu, J. Sun, Q. Zhang, and W. Ji, "A Robust Image Watermarking Scheme Based on the Relationship of SVD," in *Proceedings of the 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Harbin, China, pp. 731-734, 2008. <https://doi.org/10.1109/IIH-MSP.2008.190>
- [19] L. Zhang and D. Wei, "Dual DCT-DWT-SVD Digital Watermarking Algorithm Based on Particle Swarm Optimization," *Multimedia Tools and Applications*, Vol. 78, pp. 28003-28023, July 2019. <https://doi.org/10.1007/s11042-019-07902-9>
- [20] C.-C. Lai, H.-C. Huang, and C.-C. Tsai, "Image Watermarking Scheme Using Singular Value Decomposition and Micro-genetic Algorithm," in *Proceedings of the 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Harbin, China, pp. 469-472, 2008. <https://doi.org/10.1109/IIH-MSP.2008.168>
- [21] D. Awasthi and V. K. Srivastava, "LWT-DCT-SVD and DWT-DCT-SVD Based Watermarking Schemes with Their Performance Enhancement Using Jaya and Particle Swarm Optimization and Comparison of Results under Various Attacks," *Multimedia Tools and Applications*, Vol. 81, pp. 25075-25099, March 2022. <https://doi.org/10.1007/s11042-022-12456-4>
- [22] A. Hore and D. Ziou, "Image Quality Metrics: PSNR vs. SSIM," in *Proceedings of the 2010 20th International Conference on Pattern Recognition*, Istanbul, Turkey, pp. 2366-2369, August 2010. <https://ieeexplore.ieee.org/document/5596999>
- [23] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity," *IEEE Transactions on Image Processing*, Vol. 13, No. 4, pp. 600-612, April 2004. <https://doi.org/10.1109/TIP.2003.819861>
- [24] U. Sara, M. Akter, and M. S. Uddin, "Image Quality Assessment through FSIM, SSIM, MSE and PSNR-A Comparative Study," *Journal of Computer and Communications*, Vol. 7, No. 3, pp. 8-18, March 2019. <https://doi.org/10.4236/jcc.2019.73002>
- [25] S. Fazli and G. Khodaverdi, "Trade-off between Imperceptibility and Robustness of LSB Watermarking using SSIM Quality Metrics," in *Proceedings of the 2009 Second International Conference on Machine Vision*, Dubai, UAE, pp. 101-104, December 2009. <https://doi.org/10.1109/ICMV.2009.68>
- [26] X. Mei and X. Chu, "Dissecting GPU Memory Hierarchy Through Microbenchmarking," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 28, No. 1, pp. 72-86, 2017. <https://doi.org/10.1109/TPDS.2016.2549523>
- [27] V. Volkov, Understanding Latency Hiding on GPUs, Ph.D. Dissertation, University of California, Berkeley, 2016.

남시영 (Si-Young Nam)



2024년 : 홍익대학교 게임학부 게임소
프로그래밍학과 학사

2024년~현 재: 홍익대학교 일반대학원 게임학과 석사과정
※ 관심분야 : 정보보호(Personal Information), 블록체인(Blockchain), 워터마킹(Watermarking) 등

김혜영 (Hye-Young Kim)



1989년~1998년: (주) 현대전자 S/W연구소 책임연구원
2001년~2004년: 청강문화산업대학 컴퓨터 게임학과 전임교수
2005년~2006년: Wright State University PostDoctoral Fellow

2007년~현 재: 홍익대학교 게임학부 게임소프트웨어전공 교수
※ 관심분야 : 온라인 게임서버 및 부하분산, 디지털 콘텐츠 저작권 기술, 블록체인 및 Web3