

Gaussian Blending: Improved 3D Gaussian Splatting for Model Light-Weighting and Deep Learning-Based Performance Enhancement

Yeong-In Lee*, Jin-Nyeong Heo*, Ji-Hwan Moon**, Ha-Young Kim***

*Student, Graduate School of Information, Yonsei University, Seoul, Korea

**Student, Dept. of Artificial Intelligence, Yonsei University, Seoul, Korea

***Associate Professor, Graduate School of Information, Yonsei University, Seoul, Korea

[Abstract]

NVS (Novel View Synthesis) is a field in computer vision that reconstructs new views of a scene from a set of input views. Real-time rendering and high performance are essential for NVS technology to be effectively utilized in various applications. Recently, 3D-GS (3D Gaussian Splatting) has gained popularity due to its faster training and inference times compared to those of NeRF (Neural Radiance Fields)-based methodologies. However, since 3D-GS reconstructs a 3D (Three-Dimensional) scene by splitting and cloning (Density Control) Gaussian points, the number of Gaussian points continuously increases, causing the model to become heavier as training progresses. To address this issue, we propose two methodologies: 1) Gaussian blending, an improved density control methodology that removes unnecessary Gaussian points, and 2) a performance enhancement methodology using a depth estimation model to minimize the loss in representation caused by the blending of Gaussian points. Experiments on the Tanks and Temples Dataset show that the proposed methodologies reduce the number of Gaussian points by up to 4% while maintaining performance.

▶ **Key words:** Novel View Synthesis, 3D Gaussian Splatting, Computer Vision, NeRF, Light-Weighting, Deep Learning

[요약]

NVS는 여러 각도와 위치에서 수집한 이미지를 이용해 3차원 공간을 재현하는 연구 분야로, 증강 현실, 가상현실, 자율주행, 로봇 네비게이션 등에서 중요성이 커지고 있다. 최근 주목받는 3D-GS 방법론은 기존 NeRF 보다 고품질 장면 생성, 빠른 학습 시간, 실시간 렌더링이 가능하지만, Gaussian points의 밀도 조정 과정에서 전체 Gaussian points 수의 증가로 메모리 소모량 상승과 렌더링 속도가 저하되는 문제가 있다. 이를 개선하기 위해 본 논문에서는 불필요한 Gaussian points를 제거하여 메모리 효율성을 높이는 Gaussian blending 기법과 Gaussian points 감소로 인한 표현력 손실을 최소화하는 깊이 정보 반영 손실 함수를 제안하여 모델의 성능을 보완한다. 실험 결과, Tanks & Temples 벤치마크 데이터셋에서 성능을 유지하면서 Gaussian points 수를 최대 4% 감소시키는 효과를 확인하였다. 따라서 본 논문에서 제안한 방법론은 3D-GS 모델의 경량화 가능성을 실험적으로 증명하였다.

▶ **주제어:** 신규 뷰 합성, 3D 가우시안 스플래팅, 컴퓨터 비전, 신경 방사장, 경량화, 딥러닝

- First Author: Yeong-In Lee, Jin-Nyeong Heo, Ji-Hwan Moon, Corresponding Author: Ha-Young Kim
- *Yeong-In Lee (zeroin@yonsei.ac.kr), Graduate School of Information, Yonsei University
- *Jin-Nyeong Heo (wlssud132@yonsei.ac.kr), Graduate School of Information, Yonsei University
- **Ji-Hwan Moon (hoho@yonsei.ac.kr), Dept. of Artificial Intelligence, Yonsei University
- ***Ha-Young Kim (hayoung.kim@yonsei.ac.kr), Graduate School of Information, Yonsei University
- Received: 2024. 06. 11, Revised: 2024. 07. 23, Accepted: 2024. 07. 24.

I. Introduction

최근 딥러닝 기술의 발전은 컴퓨터 비전의 활발한 연구로 이어졌으며, 2차원 이미지를 활용한 연구 외에 비디오와 3차원 공간[1-3] 등으로 연구 대상이 확장되었다. 3차원 공간 관련 연구 중에서는 특정 공간이나 객체를 여러 각도와 위치에서 2차원 이미지로 수집하고, 수집된 2차원 이미지를 기반으로 연속적인 3차원 공간(Scene)을 재현하는 연구 분야인 NVS 연구가 활발히 진행되고 있다[4]. NVS 연구는 학습 데이터에 존재하지 않는 새로운 위치와 각도에서도 사실적인(Photorealistic) 결과물을 생성할 수 있어 3차원 정보를 필요로 하는 증강현실(Augmented Reality)과 가상현실(Virtual Reality), 자율주행, 로봇 네비게이션 등의 산업에서 많은 관심을 받고 있다[5-7].

NVS 분야의 대표적인 방법론인 NeRF[8]는 ray tracing 기법과 인공신경망을 이용하여 3차원 공간을 재현한다[9]. Ray tracing 기법은 2차원 이미지의 한 픽셀(Pixel)을 표현하기 위해 3차원 공간을 특정 각도로 광선(Ray)을 투사하여 광선 위 표본들의 색상과 밀도로 해당 픽셀의 색상을 계산하는 과정을 의미한다. 이때 광선 위 표본들의 색상과 밀도를 계산하기 위해 다층 신경 회로망(Multi-layer Perceptron)을 이용한다[8]. 그러나 NeRF의 인공신경망의 계산 복잡성 문제, 광선 위에 다량의 표본 추출로 인한 높은 연산량과 그로 인해 실시간 렌더링이 어렵다는 문제가 존재한다[4,9-11]. 이러한 문제를 해결하기 위해 다양한 후속 연구[12-14]가 진행되었는데, 최근 3D-GS[15] 방법론이 NeRF의 한계를 극복하였다. 3D-GS는 SfM(Structure from Motion)[16] 방법론을 이용하여 2차원 이미지로 3차원 포인트 클라우드(Point Cloud)를 생성하고, 이를 중심 좌표로 하는 3D Gaussian을 생성하여 3차원 공간을 재현하는 방법론이다. 3D-GS는 SfM을 통해 생성된 초기 3D Gaussian points에 래스터화(Rasterization)를 적용하여 3차원 데이터를 2차원 이미지로 변환하고, 변환한 이미지를 검증 이미지(Ground Truth Image)와 비교하며 포인트 클라우드의 밀도 조정(Density Control)과 함께 매개변수들을 갱신한다. 3D-GS의 경우, 딥러닝 모델과 ray tracing 기법을 사용하지 않기 때문에 NeRF에 비해 학습 시간이 짧으며 실시간 렌더링 속도가 빠르다는 장점이 있다. 그러나 3차원 공간을 표현하기 위해 수백만 개의 포인트 클라우드를 생성하기 때문에 포인트 클라우드 저장을 위한 메모리 소모량이 높으며, 생성된 포인트 클라우드 중 불필요한 Gaussian point가 존재하여[17,18] 3차원 공간을 효율적으로 표현하지 못하는 한계가 존재한다.

이에 본 연구에서는 3D-GS 방법론의 한계점을 개선하여 NVS의 성능을 유지하고 불필요한 Gaussian points를 제거하여 메모리 사용량 감소와 실시간 렌더링 속도 개선이 가능한 방법론을 제시하며, 그 기여점은 다음과 같다.

- 3D-GS 방법론의 밀도 조정 과정을 개선하여 신규 뷰 합성을 위해 생성되는 불필요한 Gaussian을 조정하는 Gaussian blending 기법으로 메모리 소모량을 낮춰 기존 연구의 문제를 해결한다.
- 또한, 줄어든 포인트 클라우드로 인한 정보 손실을 방지하기 위해 딥러닝 기반 깊이 추정기(Depth Estimator)[19]를 활용하여 Gaussian point의 위치를 면밀하게 조정함으로써 표현력 손실을 최소화하는 방법론을 제안한다. 3D-GS는 깊이 정보가 없는 2차원 이미지를 기반으로 학습이 진행되므로 깊이 정보 오차를 손실 함수에 추가하여 3차원 정보를 반영함으로써 성능이 개선된다.
- 최종적으로 본 논문에서는 제안한 두 방법론을 함께 적용하여, 3D-GS 방법론을 개선하는 새로운 방향을 제시한다.

본 논문의 2장에서 NVS 관련 연구 동향을 분석하고, 3장에서 3D-GS 방법론을 적용한 경량화 방법 및 성능 보완 방법론을 제안한다. 4장에서는 사용된 벤치마크 데이터셋과 실험 결과를 제시하며, 5장에서 결론과 함께 향후 연구 방향을 명시한다.

II. Related works

2.1 Neural Radiance Field

인공신경망과 ray tracing 기법을 통해 3차원 공간을 재현하는 NeRF 방법론은 광선 위에서 표본으로 추출된 점들의 색상과 밀도를 계산하기 위해 인공신경망에 표본의 좌표와 그 점을 바라보는 각도를 입력값으로 넣는다. NeRF의 인공신경망은 모든 신경망이 서로 연결된 다층 신경 회로망이며, 이 신경망은 3차원 좌표(x, y, z)와 카메라의 각도 정보인 방위각(θ)과 고도각(ϕ)을 입력으로 받아 해당 좌표의 색상과 밀도를 출력하도록 설계되어 있다. 이러한 방식으로 NeRF는 특정 각도에서 3차원 공간을 바라봤을 때 이미지 평면에 투영되는 장면을 재현함으로써 3차원 공간을 생성한다. 그러나 NeRF는 NVS의 선행 방법론[20]에 비해 메모리 측면에서 효율적이라는 특징이 있으나, 3차원 공간을 재현하기 위해 다층 신경 회로망 기반의 학습과 다량의 광선과 광선 위의 표본을 추출하기 때문에

학습 시간과 렌더링 시간이 길어 실시간 장면 재현이 어렵다는 단점이 있다[21].

이러한 NeRF 방법론의 한계를 극복하기 위해 후속 연구들이 진행되었으며, Zip-NeRF[21]에서는 Mip-NeRF의 cone tracing 기법을 사용하여 ray tracing을 적용했을 때 물체의 크기가 고려되지 않아 발생할 수 있는 왜곡 현상을 감소시킨다. 또한, NeRF 기반 방법론의 긴 학습 시간을 줄이기 위해 cone tracing으로 얻은 표본들을 연산 효율성이 높은 iNGP[22]의 grid 기반 encoding을 적용하여 학습 시간을 감소시킨다. 그러나 Zip-NeRF는 학습 시간 및 추론 시간 단축에도 불구하고 실시간 렌더링이 어렵다는 문제가 여전히 존재한다. 또 다른 연구인 Tri-MipRF[23]는 Mip-NeRF의 cone tracing를 이용하여 성능 개선하였으며, cone tracing 과정에서 다변량 Gaussian을 이용하여 표본을 추출하는 기존 방법 대신 구체를 이용하여 표본을 추출하고, 추출된 표본을 3차원 좌표의 서로 다른 조합(xy, yz, xz)으로 encoding 한 후 다층 신경 회로망의 입력값으로 넣어준다. 더불어 사물의 표면을 mesh 형태로 표현하여 사물 표면에서만 표본을 추출하여 렌더링을 진행함으로써 추론 시간을 단축시킨다. 그러나 배경을 제거한 후 사물만 재현하여 실세계를 모두 표현하지는 못한다는 한계가 존재한다.

2.2 3D Gaussian Splatting

앞서 언급한 NeRF 방법론의 문제를 개선한 3D-GS[15]는 3D Gaussian points의 매개변수(좌표, 공분산, 색상, 불투명도)를 갱신하며 비등방 Gaussian을 최적화하는 3차원 공간 재현 방법론이다. 먼저, 초기 3D Gaussian points의 중심 좌표는 2차원 학습 이미지에 SfM[16]을 적용하여 생성된 3차원 포인트 클라우드의 좌표이며, 다른 매개변수(공분산, 색상, 불투명도)는 임의값으로 정의한다. 초기화가 완료된 3D Gaussian points는 2차원 이미지를 기반으로 학습이 진행되기 때문에 3차원 공간에서 카메라 위치 정보를 활용하여 2차원 공간에 투영(Projection)된다. 3D-GS 방법론은 Gaussian points를 벡터 그래픽 형식으로 정의하기 때문에 타일 기반 래스터화를 이용하여 2D Gaussian points를 효율적으로 이미지 형식으로 변환한다. 이후 3D-GS 방법론은 래스터화를 거쳐 생성된 이미지와 검증 이미지의 오차를 손실 함수를 통해 계산하고, 역전파(Backpropagation)를 이용하여 3D Gaussian points의 매개변수(좌표, 공분산, 색상, 불투명도)를 갱신한다. 그 후, 밀도 조정 과정을 통해 Gaussian points를 분할 및 복제하여 표현력이 풍부한 3차원 공간을 재현한

다. 이러한 3D-GS의 학습 과정은 GPU(Graphic Processing Unit)에 친화적으로 설계되어 빠른 학습과 실시간 렌더링이 가능하다는 장점이 있지만, 밀도 조정 과정에서 포인트 클라우드의 밀도가 높아져 메모리 소모량이 높아지는 문제가 발생한다.

3D-GS의 높은 메모리 소모량 문제를 완화하기 위한 후속 연구들[17,18,24]이 진행되었다. 그중 Compressed 3DGS[24]는 매개변수의 민감도(Parameter Sensitivity)를 측정하여 민감도가 낮은 매개변수(Scale, Rotation, Color Coefficient)를 codebook을 이용하여 메모리 소모량을 감소시켰다. 여기서 매개변수의 민감도는 매개변수 값의 변화량이 렌더링된 이미지에 영향을 미치는 정도를 의미하며, 민감도가 높을수록 작은 변화에도 렌더링된 이미지가 달라진다. Compressed 3DGS[24]는 민감도가 낮은 매개변수를 codebook에 저장함으로써 메모리 소모량을 낮추고 성능 하락을 최소화하였다. 그러나 이 방법론은 Gaussian points를 직접적으로 줄이지 않는다는 한계가 존재한다. Compact 3D[18] 또한 codebook을 이용하여 유사한 형태의 Gaussian을 codebook으로 정의함으로써 메모리 소모량을 낮췄으며, 전체 매개변수의 약 81%를 차지하는 색상 관련 매개변수는 iNGP[22]에서 사용한 grid 기반 방법론을 활용하여 연산량과 메모리 소모량을 줄인다. 또한 3차원 공간 구현에 기여도가 낮은 Gaussian points를 선별하기 위해 Gaussian의 크기와 불투명도가 낮은 points에 mask를 적용하여 기여도를 판단한 후, 기여도가 낮은 point는 제거함으로써 모델을 경량화한다. 하지만 grid 기반 방법론을 적용하여 연산량이 증가함에 따라 특정 데이터셋에서는 FPS(Frames Per Second)가 감소하는 결과로 이어졌다. Efficient GS[17]는 밀도 조정 과정 개선 및 불필요한 Gaussian을 선별하여 모델을 경량화한다. 기존 밀도 조정 방법론에서는 Gaussian이 영향을 미치는 모든 픽셀에 대한 위치 기울기(Position Gradient) 벡터의 합으로 밀도 조정 대상을 선별한다. 하지만 이러한 방식은 Gaussian이 영향을 미치는 각 픽셀에서의 기울기 값은 크지만, 위치 기울기의 서로 다른 방향으로 인해 모든 픽셀에서의 기울기 값의 합이 0에 가까워져 밀도 조정 대상에 포함되지 않는 문제가 발생한다. 따라서 Efficient GS[17]에서는 각 기울기 벡터의 노름(Norm)을 기준으로 Gaussian이 영향을 미치는 모든 픽셀에서의 위치 기울기를 계산하여 밀도 조정 대상을 재선별하였다. 또한 2차원 이미지를 생성하는 과정에서 반영 비율이 낮은 Gaussian을 기여도 낮은 Gaussian으로 판단하여 제거함으로써 모델의 크기를 줄인다. 그러나 Efficient GS[17]에서 제안한

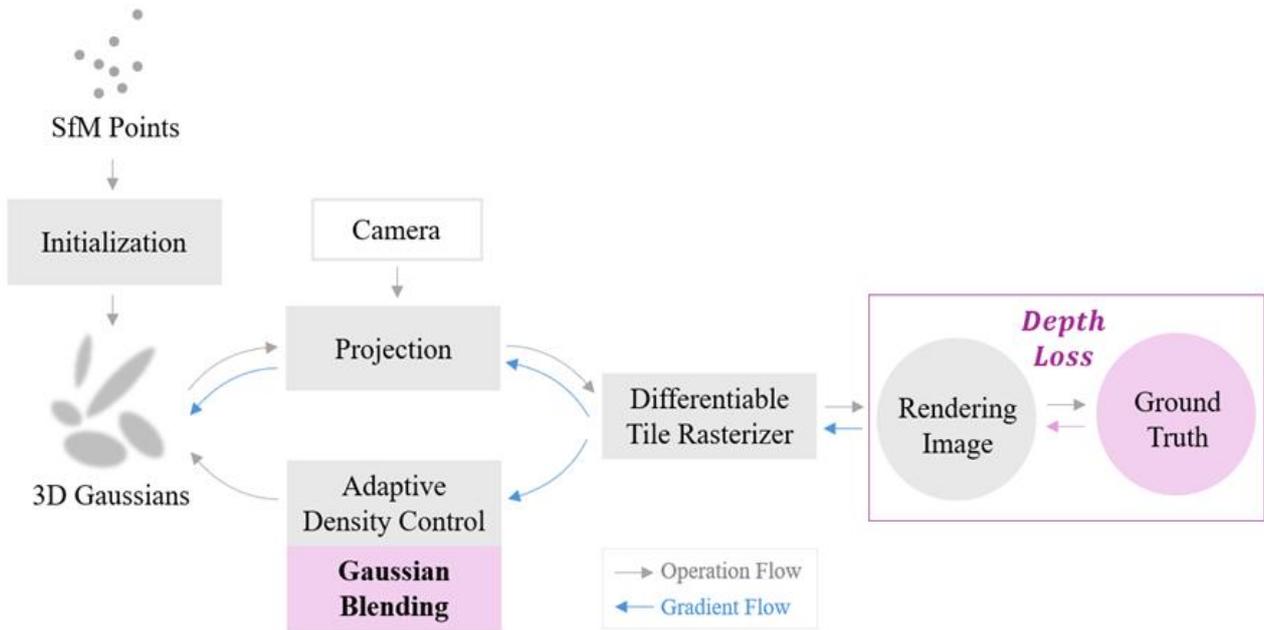


Fig. 1. Proposed 3D-GS architecture

방법론들을 학습 과정에서 적용하는 시점에 따라 모델 성능의 변화가 크다는 한계가 존재한다.

본 논문에서는 선행 연구의 메모리 소모량 감소를 위한 Gaussian 제거 방법 대신 유사한 형태의 Gaussian을 선별하여 이를 단일 Gaussian으로 융합함으로써 표현력 손실을 최소화하는 방법론을 제안한다.

III. The Proposed Scheme

본 논문에서는 3D-GS 방법론을 기본 모델로 하여 불필요한 Gaussian points를 제거할 수 있는 개선된 밀도 조정 방법론인 Gaussian blending과 Gaussian points 감소로 인한 표현력 손실을 최소화하기 위해 깊이 추정기 모델이 적용된 성능 보완 방법론을 제안한다. Fig. 1에서 제안하는 2가지 방법론이 적용된 3D-GS 모델 구조를 확인할 수 있다.

3.1 Gaussian Blending

밀도 조정 기존 3D-GS 방법론은 밀도 조정 과정을 거치면서 3차원 공간의 표현력을 강화시킨다[15]. 밀도 조정은 Gaussian points가 재현해야 하는 공간의 기하학(Geometry) 정보에 따라 Gaussian points를 복제(Clone) 혹은 분할(Split)하는 작업을 의미한다. Gaussian points가 표현해야 하는 장면에 비해 크기가 작다면

(Under-reconstruction) 동일한 point를 복제하고, 표현하고자 하는 장면에 비해 크다면(Over-reconstruction) 2개의 작은 points로 분할한다. 이러한 밀도 조정 과정을 모든 Gaussian points에 대해 진행하여 밀도가 낮은 초기 포인트 클라우드의 밀도를 높여 모델의 표현력을 향상시킨다[15]. 추가적으로 밀도 조정 과정에서 불투명도(Opacity)가 임계값(0.0002) 보다 낮은 points를 선별한 후 해당 points를 제거한다. 3D-GS 방법론은 밀도 조정 과정을 통해 3차원 공간을 사실적으로 표현하지만, 유사한 Gaussian points를 선별하는 과정 없이 분할 및 복제를 반복하여 points의 수를 지속적으로 증가시킨다. 이는 하나의 point로 재현할 수 있는 공간을 여러 개의 points로 재현하는 중복 문제를 야기한다.

Gaussian blending 기존 밀도 조정 과정의 Gaussian points 중복 문제를 해결하고자 Gaussian blending 방법론을 제안한다. 단일 Gaussian points로 표현할 수 있는 공간을 여러 points로 표현하게 되면, 최종 모델의 크기가 증가하여 메모리 소모량 및 연산량 증가로 인한 실시간 렌더링 속도가 감소하는 문제가 발생한다.

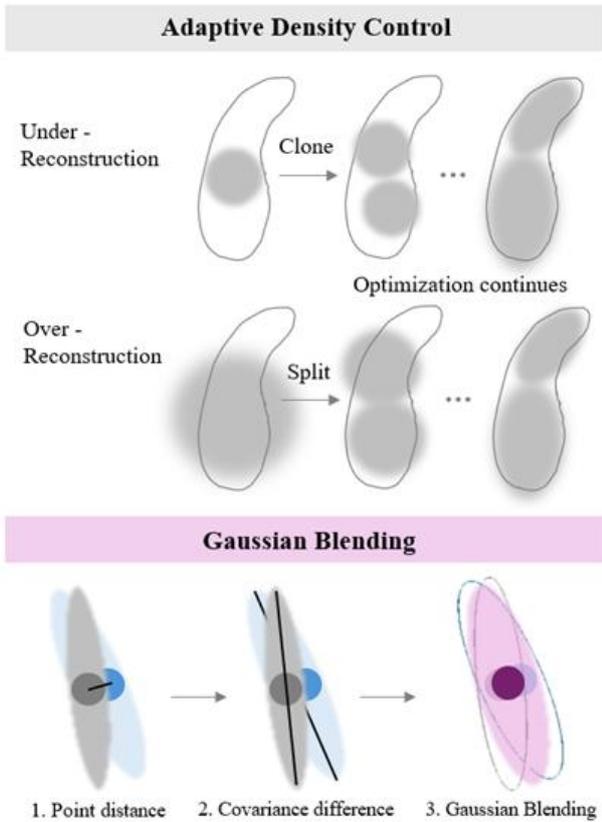


Fig. 2. Gaussian Blending

언급한 문제를 해결하기 위해 제안한 방법론은 Fig. 2에서 확인할 수 있다. Fig. 2의 ‘Adaptive Density Control’은 기존 3D-GS 방법론의 밀도 조정 과정이며, 아래의 ‘Gaussian Blending’은 본 논문에서 제안하는 개선된 밀도 조정 과정이다. Gaussian blending 기법은 기존 밀도 조정 과정을 거친 이후 Fig. 2의 Gaussian points처럼 서로 인접하고 유사한 형태의 points를 선별한 후, 선별한 points의 매개변수(좌표, 공분산, 색상, 불투명도)의 평균값으로 새로운 Gaussian을 생성한다. 이후 융합 대상이었던 유사한 points는 제거한다. 생성된 Gaussian points는 인접하고 유사한 Gaussian points의 모든 정보를 반영하기 때문에 정보 손실을 최소화할 수 있으며, 기여도 낮은 points를 제거하는 선행 연구[17,18]와는 다르게 유사한 Gaussian points를 평균값으로 융합한다는 점에서 points의 정보를 유지하며 모델의 크기를 줄일 수 있다는 장점이 있다.

유사한 Gaussian points를 융합하기 위해 본 방법론에서는 points 사이의 거리를 우선 계산한다. 모든 points의 거리를 단일 행렬 연산으로 계산할 수 있지만, 행렬 연산을 위해서 전체 points 수의 제곱의 메모리 공간이 필요하므로 단일 point와 그 외의 points의 거리를 계산하여 메

모리 초과 문제를 예방하였다. 이와 같은 거리 계산 연산을 point 별로 순차적으로 진행하여 모든 points 사이의 거리를 계산한다. 또한 융합의 대상이 되는 Gaussian points는 서로 인접해야 하므로 Gaussian의 모양을 정의하는 공분산 비교는 인접하다고 판단된 points에 대해서만 비교함으로써 불필요한 연산을 줄였다. 인접한 point라고 판단하는 임계값은 실험을 통해 0.001로 설정하였으며, 공분산 값 또한 실험적으로 0.005로 설정하였다. 이처럼 Gaussian blending 과정에서 불필요한 연산을 제거하여 계산량과 메모리 사용량을 감소시킬 수 있으며, 제안한 방법론은 선행연구[17,18]처럼 새로운 매개변수를 정의하지 않고 points의 수를 줄일 수 있다.

그러나, Gaussian blending 방법론을 사용하게 되면 기존 3D-GS 방법론으로 재현한 3차원 공간보다 Gaussian points가 줄어들기 때문에 비교적 정보가 손실된다. 이를 보완하고자 3.2에서 깊이 정보를 반영하는 깊이 손실 함수(Depth Loss Function)를 제안한다.

3.2 Depth Estimation

Computer Vision의 발전에 따라 시각 이미지와 관련된 여러 분야의 연구가 진행되었고, 그중 2차원 이미지를 이용하여 3차원 정보인 깊이를 추정하는 연구 또한 활발히 이루어졌다[19,25]. 깊이 정보가 존재하지 않는 2차원 이미지에서 깊이를 추정하는 depth estimation 분야는 저차원 데이터로 고차원 데이터를 얻을 수 있어 여러 3차원 연구 분야에서 활용 가능성이 높다[25]. 따라서 Gaussian blending으로 인해 감소하는 표현력을 보완하고자 깊이 손실 함수를 구성하였다. 기존 3D-GS 방법론은 재현된 3차원 공간을 2차원 이미지로 투영시킨 후 검증 이미지와 비교하며 매개변수를 갱신한다. 하지만 깊이 정보가 없는 2차원 이미지를 기반으로 3차원 공간을 재현하기 때문에 정밀한 묘사가 어렵다는 한계가 있다. 따라서 본 논문에서 제안하는 깊이 손실 함수는 깊이 정보를 기반으로 Gaussian points의 매개변수를 조정하기 위해 Transformer 기반의 암호기와 해독기(Encoder and Decoder)를 사용한 ZoeDepth[19]를 활용하여 검증 이미지와 재현된 이미지의 깊이 정보를 추정한다. 추정된 두 이미지의 깊이 값의 차이는 식 (1)을 통해 계산하고, 식 (1)의 $Depth_{gt}$ 와 $Depth_{pred}$ 는 각각 검증 이미지와 재현된 이미지의 추정된 깊이 정보를 의미한다. 식 (1)을 통해 계산된 오차는 두 이미지의 차이를 비교하는 SSIM[26] 연산인 식 (2)와 함께 최종 손실 함수인 식 (4)에 반영되어 매개변수를 갱신한다. 식 (2)는 두 이미지(x, y)의 상관관계

를 휘도(Luminance), 대비(Contrast), 구조(Structure) 측면에서 평가하는 식이며 μ , σ , c 는 각각 이미지 내 픽셀의 평균, 표준편차, 공분산을 의미한다[26]. 식 (2)에서 계산된 두 이미지의 유사도는 값이 높을수록 서로 유사함을 의미한다. 따라서 두 이미지의 오차를 학습하기 위해 식 (2)의 역수를 적용한 식 (3)을 통해 최종 손실 함수에 두 이미지의 오차를 반영하여 계산한다. 식 (1)과 (3)을 통해 계산된 손실값은 반영 비율을 의미하는 식 (4)의 λ_{D-SSIM} 는 0.2이며, λ_{depth} 는 1.0로 반영 정도가 정의된다. 그리고 식 (4)의 L_1 은 검증 이미지와 재현된 이미지의 L_1 노름으로 계산한 오차를 의미한다.

$$L_{depth} = |Depth_{gt} - Depth_{pred}| \quad (1)$$

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (2)$$

$$D-SSIM(x,y) = \frac{1 - SSIM(x,y)}{2} \quad (3)$$

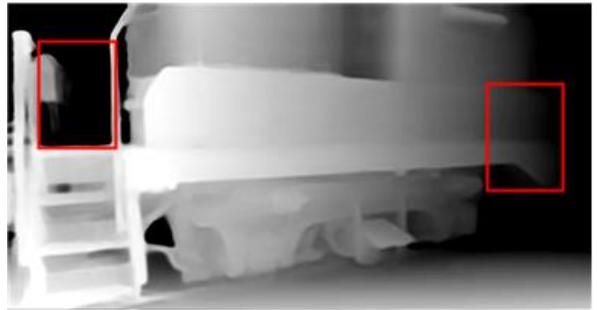
$$L = (1 - \lambda_{D-SSIM})L_1 + \lambda_{D-SSIM}L_{D-SSIM} + \lambda_{depth}L_{depth} \quad (4)$$

Fig. 3(a)는 검증 이미지, Fig. 3(b)는 검증 이미지의 깊이 정보를 시각화한 이미지, Fig. 3(c)는 재현된 이미지의 깊이 정보를 시각화한 이미지이며, 시각화된 이미지는 거리가 멀수록 어둡게 표현된다. Fig. 3(b)에서 어둡게 표현된 영역이 Fig. 3(c)에서는 밝게 표현되는 현상을 통해 Gaussian points가 잘못된 위치에서 공간을 재현하고 있음을 확인할 수 있다. 따라서 본 논문에서 제안한 깊이 손실 함수를 이용하여 매개변수를 갱신하면, 더 정밀하게 3차원 공간을 재현할 수 있다.

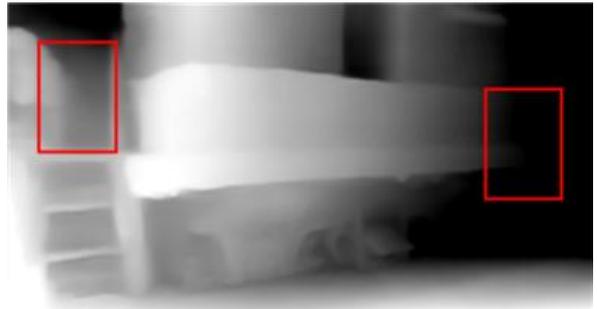
두 방법론을 함께 적용하면 Fig. 1의 전체 아키텍처에서 볼 수 있듯이, 먼저 제안한 Gaussian blending 방법론의 강점을 유지하면서 손실 함수 계산에 깊이 정보 오차를 더하여 성능을 보완할 수 있다. 이는 기존 3D-GS 방법론에서 학습되지 못한 고차원 깊이 정보를 통해 매개변수를 조정함으로써 명확한 3차원 공간 재현과 함께 메모리 효율을 개선할 수 있다. 따라서 본 논문에서는 제안한 두 가지 방법론을 함께 사용하여 기존 3D-GS 방법론 보다 효율적이며 정확한 3D Gaussian 조정이 가능하도록 개선하였다.



(a) Ground truth image



(b) Depth map of the ground truth image



(c) Depth map of the reconstructed image

Fig. 3. Images generated from a depth estimator

IV. Experimental Results

4.1 Dataset & Evaluation Metrics

실험에 사용한 데이터셋은 3차원 공간 구현을 위해 산업용 레이저 스캐너로 수집된 Tanks and Temples 벤치마크 이미지 데이터셋이다[27]. 학습 데이터셋과 검증 데이터셋은 3D-GS의 설정과 동일하게 구성하였다. Fig. 4에서 기차와 트럭 원본 이미지를 확인할 수 있으며, 전체 기차 이미지와 트럭 이미지는 각각 301장과 251장으로 구성되어 있다.

모든 실험은 RTX 3090 GPU 1대에서 실행한 결과이며, 실험 진행 시 사용한 하이퍼 매개변수(Hyper Parameter)는 3D-GS의 설정과 동일하다. 성능을 평가하기 위한 지표는 NVS 분야에서 일반적으로 사용되는 PSNR (Peak

Signal to Noise Ratio), SSIM (Structural Similarity Index Map), LPIPS (Learned Perceptual Image Patch Similarity) 등 3가지를 사용하였다[15]. PSNR, SSIM 평가지표는 높은 수치일수록, LPIPS는 낮은 수치일수록 고 성능을 의미한다. 추가적으로 최종 Gaussian points 수와 렌더링 속도(FPS)를 평가지표와 병기하여 모델의 경량화 성능을 평가하였고, 모델 평가는 7K iterations와 30K iterations에서 진행하였다.

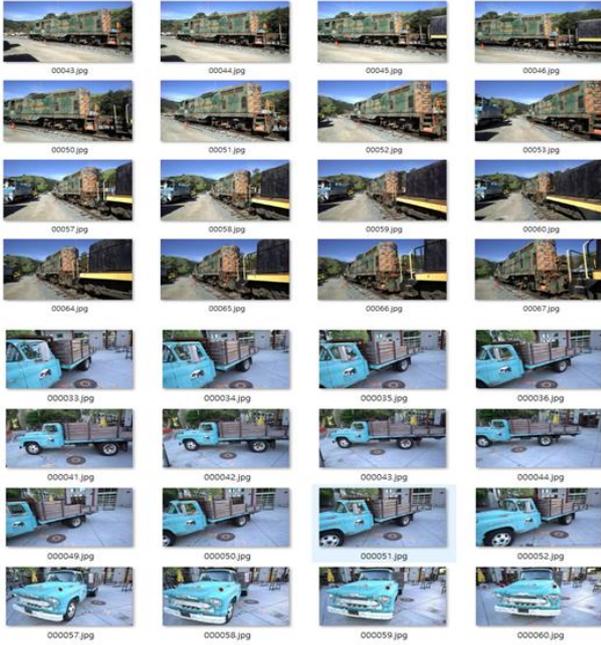


Fig. 4. Tanks and Temples dataset

4.2 Experiment Results

본 논문에서 제시한 방법론의 타당성을 확인하기 위해 기존 3D-GS 모델과 비교 실험을 진행하였다.

Fig. 5에서 초기 생성된 포인트 클라우드의 이미지를 확인할 수 있으며, 위의 이미지는 기차를, 아래는 트럭의 포인트 클라우드를 시각화한 이미지이다.

먼저, Gaussian blending 기법을 적용한 실험은 Table 1에서 확인할 수 있으며, 실험 결과 Gaussian point 수의 감소를 확인하였으나, Gaussian point의 제거로 인한 표현력 손실은 소폭의 성능 하락으로 이어졌다.

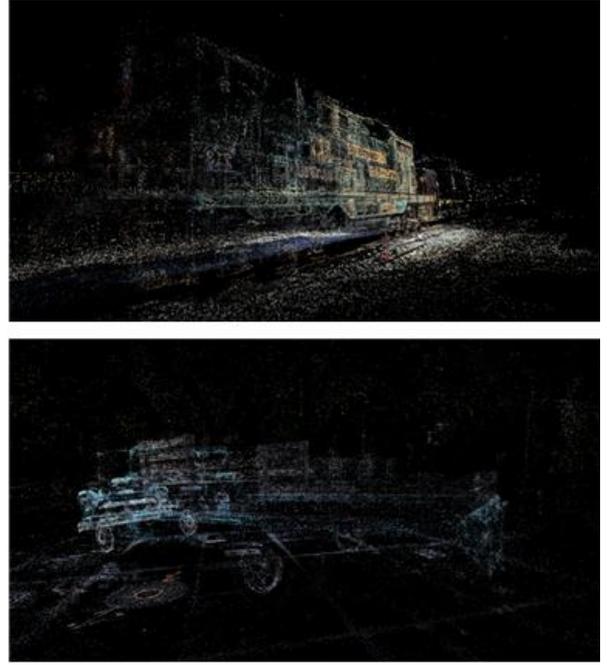


Fig. 5. Initial point clouds (Top=train / Bottom=truck)

Table 1. Experimental results on Gaussian blending

		Iteration 7K					
		Method	PSNR ↑	SSIM ↑	LPIPS ↓	Gaussians	FPS
train	Baseline	19.422	0.709	0.269	657,272	-	
	+ GB	19.337	0.707	0.273	651,002	-	
truck	Baseline	23.901	0.845	0.107	1,851,759	-	
	+ GB	23.887	0.846	0.106	1,849,414	-	
		Iteration 30K					
		Method	PSNR ↑	SSIM ↑	LPIPS ↓	Gaussians	FPS
train	Baseline	21.968	0.811	0.130	1,080,379	328	
	+ GB	21.950	0.811	0.130	1,068,565	332	
truck	Baseline	25.372	0.878	0.064	2,573,881	217	
	+ GB	25.373	0.878	0.065	2,489,281	225	

Table 2. Experimental results on depth estimator

		Iteration 7K					
		Method	PSNR ↑	SSIM ↑	LPIPS ↓	Gaussians	FPS
train	Baseline	19.422	0.709	0.269	657,272	-	
	+ depth	19.448	0.708	0.270	657,512	-	
truck	Baseline	23.901	0.845	0.107	1,851,759	-	
	+ depth	23.867	0.846	0.106	1,844,374	-	
		Iteration 30K					
		Method	PSNR ↑	SSIM ↑	LPIPS ↓	Gaussians	FPS
train	Baseline	21.968	0.811	0.130	1,080,379	328	
	+ depth	22.020	0.812	0.130	1,080,197	328	
truck	Baseline	25.372	0.878	0.064	2,573,881	217	
	+ depth	25.403	0.879	0.064	2,580,689	216	

성능 하락을 보완하기 위해 깊이 추정기를 단독으로 적용하였을 때는 Table 2의 결과처럼 기존 방법론 보다 성능이 높아지는 것을 확인할 수 있다. 이는 Gaussian blending으로 감소하는 표현력을 깊이 추정기를 통해 보완할 수 있음을 의미하며, Table 3에서 이를 실험적으로 증명하였다.

Table 3. Experimental results on Gaussian blending and depth estimator

		Iteration 7K				
Method		PSNR ↑	SSIM ↑	LPIPS ↓	Gaussians	FPS
Train	Baseline	19.422	0.709	0.269	657,272	-
	GB+dep	19.495	0.709	0.270	648,385	-
Truck	Baseline	23.901	0.845	0.107	1,851,759	-
	GB+dep	23.817	0.844	0.108	1,810,952	-
		Iteration 30K				
Method		PSNR ↑	SSIM ↑	LPIPS ↓	Gaussians	FPS
Train	Baseline	21.968	0.811	0.130	1,080,379	328
	GB+dep	21.980	0.810	0.133	1,062,583	334
Truck	Baseline	25.372	0.878	0.064	2,573,881	217
	GB+dep	25.384	0.878	0.064	2,479,871	230

Gaussian blending 기법과 깊이 추정기를 함께 적용한 결과, Table 3의 결과처럼 Gaussian blending으로 낮아진 성능이 보완되었고, Gaussian points 수 또한 감소한 것을 확인할 수 있다.

Fig. 6에서 Fig. 6(a) 기차 이미지, Fig. 6(b) 트럭 이미지의 7K, 30K iterations에서 재현된 이미지를 순서대로 확인할 수 있다.



(a) Reconstructed train images



(b) Reconstructed truck images

Fig. 6. Images reconstructed after 7K, 30K iterations

본 논문에서 제시한 방법론은 Gaussian blending을 이용하여 중복된 Gaussian points를 단일 point로 융합하여 모델을 경량화하였으며, 깊이 추정기를 함께 활용하여 성능 하락을 최소화하였다.

IV. Conclusions

3차원 공간 구현의 활용도를 높이기 위해서 실시간 렌더링과 정확도는 필수적이다. 따라서 본 논문은 NVS 분야에서 활발히 연구되는 3D-GS 방법론의 밀도 조절을 개선한 두 가지 방법론을 제안하였다. 먼저 Gaussian blending 기법으로 중복되는 Gaussian points를 융합한 단일 point로 표현함으로써 모델을 경량화하였으며, 줄어든 Gaussian points로 인한 표현력 손실은 깊이 추정기 모델을 활용하여 검증 이미지와 재현된 이미지의 깊이 정보를 추정한 후, 두 이미지의 오차를 손실 함수에 반영하여 보완하였다. 따라서 본 연구는 생성되는 Gaussian points 수의 효율성에 대해 고찰하여 기존 3D-GS 연구의 새로운 개선 방향을 제시하였다는 의의가 있다. 그러나 본 연구에서 제안한 Gaussian blending 기법은 생성된 모든 points에 대한 거리값을 계산하여 선별하기 때문에, 인접한 points를 선별하는 최적의 방법으로 보기 어렵다. 이를 보완하기 위해 K-최근접 이웃 알고리즘(K-Nearest Neighbors)을 활용하여 Gaussian points 선별 기준을 새

롭게 정의하거나, GNN(Graph Neural Networks)를 활용하여 points 간의 관계성을 구성할 수 있도록 하여, 표현하고자 하는 공간에 대한 정보를 모델이 추가적으로 학습하도록 추후 연구 방향을 제시할 수 있다. 더불어 융합 대상인 points를 선별하는 기준에 Gaussian의 색상 정보를 추가하여 유사도 기준을 재정의함으로써 성능을 개선하는 연구 방향을 제시할 수 있다.

ACKNOWLEDGEMENT

This work was supported by Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea government (MOTIE)

(20202020800030, Development of Smart Hybrid Envelope Systems for Zero Energy Buildings through Holistic Performance Test and Evaluation Methods and Fields Verifications)

REFERENCES

- [1] Y-J. Lee, "A Design of Anomaly Behavior Detection System Based on Deep Learning Model Using CCTV", *Journal of Knowledge Information Technology and Systems(JKITS)*, Vol. 17, No. 2, pp. 183-191, April 2022. DOI: 10.34163/jkits.2022.17.2.001
- [2] J-N. Heo, Y-I. Lee, and H-Y. Kim, "Pothole Detection Using Deep Learning and Domain-based Image Preprocessing Methods", *Journal of Knowledge Information Technology and Systems(JKITS)*, Vol. 18, No. 5, pp. 1331-1343, October 2023. DOI: 10.34163/jkits.2023.18.5.029
- [3] D-H Kim, and H-H Choi, "A Research on Cylindrical Pill Bottle Recognition with YOLOv8 and ORB", *Journal of the Korea Society of Computer and Information(JKSCI)*, Vol. 29, No. 2, pp. 13-20, February 2024. DOI: 10.9708/jksci.2024.29.02.013
- [4] Zhang, Jiahui, et al., "FreGS: 3d gaussian splatting with progressive frequency regularization", In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21424-21433, Seattle WA, USA, 2024. DOI: 10.48550/arXiv.2403.06908
- [5] Qin, Minghan, et al., "Langsplat: 3d language gaussian splatting", In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20051-20060, Seattle WA, USA, 2024. DOI: 10.48550/arXiv.2312.16084
- [6] Zhou, Xiaoyu, et al., "Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes", In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21634-21643, Seattle WA, USA, 2024. DOI: 10.48550/arXiv.2312.07920
- [7] T. Chen, O. Shorinwa, W. Zeng, J. Bruno, P. Dames, and M. Schwager, "Splat-Nav: Safe Real-Time Robot Navigation in Gaussian Splatting Maps", *arXiv preprint arXiv:2403.02751*, 2024. DOI: 10.48550/arXiv.2403.02751
- [8] Mildenhall, Ben, et al., "Nerf: Representing scenes as neural radiance fields for view synthesis", *Communications of the ACM*, Vol. 65, No. 1, pp. 99-106, Online, 2021. DOI: 10.48550/arXiv.2003.08934
- [9] G. Chen, and W. Wang, "A survey on 3d gaussian splatting", *arXiv preprint arXiv:2401.03890*, 2024. DOI: 10.48550/arXiv.2401.03890
- [10] Malarz, Dawid, et al., "Gaussian Splitting Algorithm with Color and Opacity Depended on Viewing Direction", *arXiv preprint arXiv:2312.13729*, 2023. DOI: 10.48550/arXiv.2312.13729
- [11] Lu, Tao, et al., "Scaffold-gs: Structured 3d gaussians for view-adaptive rendering", In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20654-20664, Seattle WA, USA, 2024. DOI: 10.48550/arXiv.2312.00109
- [12] Yu, Alex, et al., "pixelnerf: Neural radiance fields from one or few images", In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4578-4587, Online, 2021. DOI: 10.48550/arXiv.2012.02190
- [13] Xu, Qiangeng, et al., "Point-nerf: Point-based neural radiance fields", In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5438-5448, New Orleans, Louisiana, USA, 2022. DOI: 10.48550/arXiv.2201.08845
- [14] Barron, Jonathan T., et al., "Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields", In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5855-5864, Online 2021. DOI: 10.48550/arXiv.2103.13415
- [15] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3D Gaussian Splatting for Real-Time Radiance Field Rendering", *ACM Transactions on Graphics*, Vol. 42, No. 4, August 2023. DOI: 10.48550/arXiv.2308.04079
- [16] J. L. Schonberger, and J-M. Frahm, "Structure-from-motion revisited", In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4104-4113, Las Vegas, Nevada, USA, 2016.
- [17] Liu, Wenkai, et al., "EfficientGS: Streamlining Gaussian Splatting for Large-Scale High-Resolution Scene Representation", *arXiv preprint arXiv:2404.12777*, 2024. DOI: 10.48550/arXiv.2404.12777

- [18] Lee, Joo Chan, et al., "Compact 3d gaussian representation for radiance field", In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 21719-21728, Seattle WA, USA, 2024. DOI: 10.48550/arXiv.2311.13681
- [19] Bhat, Shariq Farooq, et al., "Zoedepth: Zero-shot transfer by combining relative and metric depth", arXiv preprint arXiv:2302.12288, 2023. DOI: 10.48550/arXiv.2302.12288
- [20] Yu, Alex, et al., "Plenoxels: Radiance fields without neural networks", arXiv preprint arXiv:2112.05131 2.3, 2021. DOI: 10.48550/arXiv.2112.05131
- [21] Barron, Jonathan T., et al., "Zip-nerf: Anti-aliased grid-based neural radiance fields", In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 19697-19705, Vancouver, Canada, 2023. DOI: 10.48550/arXiv.2304.06706
- [22] Müller, Thomas, et al., "Instant neural graphics primitives with a multiresolution hash encoding", ACM Transactions on Graphics, Vol. 41, No. 4, pp. 1-15, 2022. DOI: 10.1145/3528223.3530127
- [23] Hu, Wenbo, et al., "Tri-miprf: Tri-mip representation for efficient anti-aliasing neural radiance fields", In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 19774-19783, Paris, France, 2023. DOI: 10.48550/arXiv.2307.11335
- [24] Niedermayr, Simon, Josef Stumpfegger, and Rüdiger Westermann, "Compressed 3d gaussian splatting for accelerated novel view synthesis", In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10349-10358, Seattle WA, USA, 2024. DOI: 10.48550/arXiv.2401.02436
- [25] Zhu, Shengjie, and Xiaoming Liu, "LightedDepth: Video depth estimation in light of limited inference view angles", In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5003-5012, Paris, France, 2023.
- [26] Wang, Zhou, et al., "Image quality assessment: from error visibility to structural similarity", IEEE Transactions on Image Processing, pp. 600-612, 2004. DOI: 10.1109/TIP.2003.819861
- [27] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun, "Tanks and temples: Benchmarking large-scale scene reconstruction", ACM Transactions on Graphics, Vol. 36, No. 4, 2017. DOI: 10.1145/3072959.3073599

Authors



Yeong-In Lee received the B.S. in the Department of Business Administration and Visual Media Design from the Sookmyung Women University, Seoul, Korea, in 2018. She is pursuing a master's degree at the Graduate School of Information, Yonsei University, Seoul, Korea since 2023. Her current research areas are deep Learning, computer vision, anomaly detection.



Jin-Nyeong Heo received the B.S. in the Department of Business Administration from Korea Aerospace University, Goyang, Korea, in 2023. He is pursuing a master's degree at the Graduate School of Information, Yonsei University, Seoul, in Korea since 2023. His current research interests include deep learning, computer vision, data analysis.



Ji-Hwan Moon received the B.S. in the Department of Computer Engineering, Gachon University, Seongnam, Korea, in 2023. He is currently pursuing M.S. degree in the Department of Artificial Intelligence, Yonsei University, Seoul, Korea since 2023. His current research interests include computer vision, generative model, federated learning, and MLOps.



Ha-Young Kim is an Associate Professor at Graduate School of Information, Yonsei University, Korea. She received her Ph.D. degree at Department of Mathematics, Purdue University, United States. From 2011 to 2016, she was a research staff member in Samsung Advanced Institute of Technology (SAIT) of Samsung Electronics, Korea, working on various recognition systems with deep learning. Her primary research areas are deep learning and computational Finance. She has published in leading journals, including Information Fusion, Applied Soft Computing, Expert Systems with Applications, Stochastic and Dynamics, Computers in Biology and Medicine, PLoS ONE, Automation in Construction, Journal of Computing in Civil Engineering and Annals of Finance. She is the inventor of 8 patents and 13 patent applications.