Check for updates
특집논문 (Special Paper)
방송공학회논문지 제29권 제6호, 2024년 11월 (JBE Vol.29, No.6, November 2024)
https://doi.org/10.5909/JBE.2024.29.6.915
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

몰입형 3차원 영상의 고효율 저장 및 전송을 위한 동적 3D Gaussian Splatting 모델의 압축

최 재 열[a)], 김 영 규[b)], 정 종 범[c)], 박 준 형[b)], 류 은 석[b)‡]

Compression of Dynamic 3D Gaussian Splatting for Efficient Storage and Transmission of Immersive Video

Jaeyeol Choi[a)], Yeong-Gyu Kim[b)], Jong-Beom Jeong[c)], Jun-Hyeong Park[b)], and Eun-Seok Ryu[b)‡]

요 약

3D gaussian splatting (3DGS) 기술이 고품질의 3차원 재구성 및 고속의 자유 시점 렌더링을 가능하게 함에 따라 이를 응용한 기법이 급속도로 개발되고 있다. 그 중 동적 장면 재구성을 제공하는 4D gaussian splatting (4D-GS)은 위치 및 시간 입력값을 4차원 신경 복셀로 부호화한 후, 이를 보간하여 모든 시간대를 포괄하는 가능한 표준 3DGS 모델에 입력 가능한 변형값을 추론하는 과정을 거친다. 이러한 4D-GS 모델로부터 영상을 렌더링 하기 위해서는 고차원의 feature 정보와 인공신경망, 그리고 표준 3DGS 모델이 공통적으로 필요하기에 전체 파일 크기가 상대적으로 커 저장 및 전송에 비효율적이다. 이에 본 연구에서는 4D-GS를 구성하는 4차원 신경 복셀에 대해 2차원 평면으로 분해한 후 이에 양자화를 적용하였고, 동일 복셀 내 다수 채널을 구성하는 평면 간 시간축으로 병합한 후 동영상 코덱을 적용하는 방식으로 압축하였다. 또 다른 구성 요소인 표준 3DGS 집합에 대해서는 기여도 점수 기반으로 제거하는 기법을 도입하였다. 실험 결과 압축을 통해 렌더링 품질을 최소화하여 모델의 전체 비트율을 42.6% 감소시킬 수 있는 것이 확인되었다. 논문에서 제시한 기법을 활용하여 높은 저장 및 전송 효율을 갖는 자유 시점 비디오 시스템을 개발할 수 있을 것으로 기대된다.

Abstract

As 3D gaussian splatting (3DGS) method enables high-quality 3D reconstruction and fast free-viewpoint rendering, various techniques utilizing 3D gaussians are rapidly being developed. Among these, 4D gaussian splatting (4D-GS) supports dynamic scene reconstruction by encoding positional and temporal inputs into 4D neural voxels. These voxels are interpolated to infer deformation values, which serve as inputs for canonical 3DGS fields covering all timestamps. However, rendering images from 4D-GS model requires feature voxel, neural networks, and canonical 3DGS model, resulting in a large file size that is inefficient for storage and transmission. In this study, quantization is applied to the 4D neural voxels of the 4D-GS after decomposing them into multiple 2-dimensional planes. These planes are then concatenated along the temporal axis across multiple channels within the same voxel, followed by video codec encoding. Additionally, gaussian pruning method to remove gaussians of canonical 3DGS fields based on contribution scores are conducted. Experimental results demonstrated a 42.6% reduction of overall bitrate while minimizing rendering quality degradation. The proposed methods are expected to contribute to the development of free-viewpoint video systems with high efficiency.

Keyword : 3D gaussian splatting, Free-viewpoint video, Artificial intelligence, Metaverse, Virtual reality

Ⅰ. 서 론

최근 가상현실 및 증강현실 기술이 차세대 미디어 기술로 주목받음에 따라 3차원 몰입형 미디어 표현 방식에 대한 연구가 활발히 진행되고 있다. 현재 가상/증강 현실 서비스 중 가장 많이 사용되는 게임 또는 산업용 콘텐츠의 구현을 위해서는 Unity 또는 Unreal Engine 등의 게임 엔진을 활용한 컴퓨터 그래픽스 기반 기술이 주로 활용되고 있다. 이를 활용하여 사용자의 움직임과 바라보는 방향에 대응하는, 즉 six degrees of freedom (6DoF)의 장면 렌더링을 지원하는 콘텐츠를 제작할 수 있다. 하지만 실시간으로 공연 또는 스포츠 이벤트의 6DoF 콘텐츠를 사용자에게 제공하기 위해서는 게임 엔진을 거치지 않는 동영상 처리 기반의 영상 송출 시스템이 필요한데, 이를 위해 전통적으로 6DoF 시점에 해당하는 중간 뷰를 실시간으로 합성하는 novel view synthesis 기반의 시스템이 사용되어 왔다[1]. 클라이언트 레벨에서의 중간 뷰 합성을 위해서는 다중 뷰 동영상을 전송해야 하는데, 이때 파일 크기가 큰 다중 뷰 동영상을 압축하기 위해 뷰 간의 중복된 영역을 제거하고 고유의 영역의 조각끼리 조합하여 아틀라스를 생성한 후 디코딩 과정에서 복원하는 MPEG immersive video (MIV)[2]가 대두되었다. 하지만 이미지로부터 깊이 기반 합성을 통해 새로운 이미지를 만드는 방법은 제공할 수 있는 뷰포인트 범위가 제한된다는 한계가 있다. 이에 따라 3차원 데이터를 직접적으로 처리하여 사용자가 원하는 시점에서의 영상을 생성하는 volume rendering (볼륨 렌더링) 기술이 주목받고 있다. 3차원 공간 내 좌표의 색상 및 밀도 값을 인공신경망을 통해 학습시키는 neural radiance fields (NeRF)[3] 기술이 대표적인 예이다. NeRF 모델은 미분 가능한 렌더링 파이프라인을 통해 출력된 이미지와 원본 이미지 사이의 오차를 사용하여 엔드-투-엔드의 학습할 수 있다는 점과 바라보는 방향에 따라 다른 밝기의 색상을 출력하는 non-lambertian 효과를 지원한다는 점에서 화제를 얻었고, NeRF를 기반으로 동적 장면을 표현하는 여러 모델도 등장하였다.

하지만 NeRF 기반 기법에는 한계점이 존재하는데, 영상을 렌더링하는데 시간이 비교적 길게 소요되어 실시간 시스템의 구현이 어렵다는 것이다. 이를 해결하고자 다수의 3차원 가우시안 확률분포를 사용하여 3차원 공간을 표현하는 3D gaussian splatting (3DGS)[4] 기법이 등장했다. NeRF와 마찬가지로 고품질의 미분 가능한 볼륨 렌더링을 가능하게 함과 동시에, NeRF 기반 방식과 비교하여 렌더링 속도가 1080p 기준 100fps 이상으로 매우 빠르다는 장점이 있다. 하지만 인공신경망을 통해 암시적으로 정보를 표현하는 것이 아닌 명시적으로 장면 당 10만 개 이상의 가우시안 분포를 사용하여 표현하기에 파일 크기와 메모리 사용량이 증가한다는 단점이 존재한다. 이에 더해 동적 장면을 표현하는 3DGS 모델은 시간에 따라 변화하는 장면의 정보를 포함하기에 파일 크기가 더욱 커지는 상황이 발생한다. 이를 해결하기 위해 본 연구에서는 동적 장면을 표현하는 3DGS 모델을 압축하기 위한 연구를 동적 3DGS 모델 중 하나인 4D Gaussian Splatting (4D-GS)[5]을 기반으로 진행하였다. 구체적으로, 4D-GS 모델은 공간을 나타내는 x, y, z축과 함께 시간축 t를 가지는 4차원 공간을 xy, yz, zx, xt, yt, zt 6 종류의 임베딩으로 분해하여 표현한다. 이를 통해 입력된 4차원 좌표를 표준 3DGS 필드의 변형 좌표로 변환하여 쿼리를 진행하게 된다. 본 연구에서는 4D-GS에 사용된 4차원 신경 복셀을 동영상 코덱을 사용하여 부호화함과 동시에 표준 3DGS 필드에 가우시안 제거 (프루닝) 기법을 적용하여 전반적인 모델 크기를 감축하였다. 학습 시

a) 성균관대학교 인공지능융합학과(Department of Applied Artificial Intelligence, Sungkyunkwan University)
b) 성균관대학교 실감미디어공학과(Department of Immersive Media Engineering, Sungkyunkwan University)
c) 성균관대학교 컴퓨터교육학과(Department of Computer Science Education, Sungkyunkwan University)
‡ Corresponding Author : 류은석(Eun-Seok Ryu)
E-mail: esryu@skku.edu
Tel: +82-2-760-0677
ORCID: https://orcid.org/0000-0003-4894-6105
※ 이 논문의 연구 결과 중 일부는 한국방송·미디어공학회 2024년 하계학술대회에서 발표한 바 있음.
※ This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(RS-2024-00436936) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation).
※This research was supported by Global Standardization and Commercialization of Copyright Technology Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism (Project Name: Development of International Standards for CT XR Content Copyright Protection Technologies, Project Number: RS-2024-00439789, Contribution Rate: 50%).
· Manuscript September 5, 2024; Revised October 18, 2024; Accepted October 21, 2024.

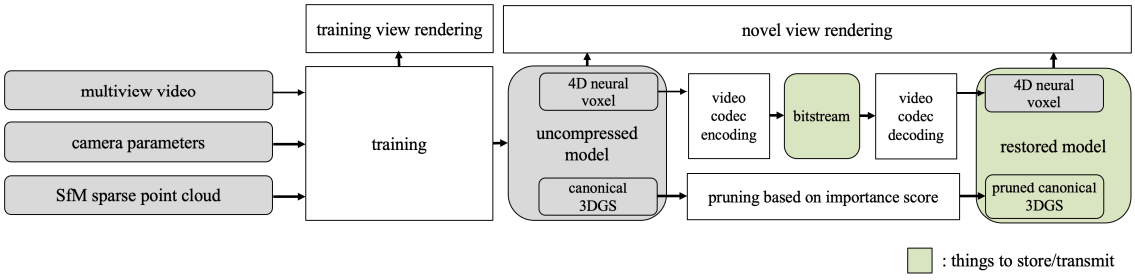


그림 1. 제안하는 동적 3DGS 모델의 훈련 및 압축 파이프라인
Fig. 1. Proposed pipeline of training and compressing dynamic 3DGS model

점부터 압축 알고리즘을 적용하여 학습 완료된 기존 3DGS 모델을 사용할 수 없는 기존 3DGS 압축 방법론과 달리, 본 논문에서 소개하는 접근법은 그림 1과 같이, 이미 학습 완료된 3DGS 기반 모델에 압축 모듈을 선택적으로 적용할 수 있다는 점에서 장점을 갖는다. 또한 실험 결과 비트율 및 품질 관점에서 기존 볼륨 렌더링 기반 공간 표현 모델 대비 우수한 성능을 나타낸다.

본 논문의 2장은 배경 연구로 볼륨 렌더링 기반의 동적 장면 표현 모델의 사례와 이에 대한 압축 연구를 소개한다. 3장에서는 4차원 복셀의 부호화 기법과 표준 가우시안 프루닝 기법에 대한 설명과 구현 내용을 설명하고, 4장에서는 실험 결과를 나타내고 타 기법과 비교한 결과에 대해 고찰한다. 이어서 5장에서는 결론을 맺으며 논문을 마무리한다.

II. 배경 연구

본 연구의 이해를 위한 배경 연구를 세 절로 나누어 설명한다. 2-1절에서는 대표적인 기계학습 기반 볼륨 렌더링 모델인 NeRF와 3DGS 기술에 대해 설명하고, 2-2절에서는 그 중 시간에 따라 변화하거나 움직이는 장면을 표현할 수 있는 동적 볼륨 렌더링 모델을 소개한다. 이어서 2-3절에서는 3DGS의 압축에 관련된 기존 연구를 소개한다.

1. Neural Radiance Fields 와 3D Gaussian Splatting

포인트 클라우드 또는 메쉬를 사용하던 기존의 3차원 모델링 방식과 다르게, NeRF[3]는 인공신경망을 사용하여 공간 내 점에서의 색상 및 밀도값을 학습한다. 제한된 시점에서 관찰된 2차원 색상 이미지와 카메라 매개변수만을 입력으로 받아 3차원 재구성을 실시하고 새로운 시점에서의 이미지를 예측할 수 있다는 점에서 강점을 갖는다. 이어서 기본 NeRF를 발전시킨 후속 연구가 다수 등장했다. 예를 들어, Instant-NGP[6]는 입력 좌표에 대한 다해상도 해시 인코딩을 도입하여 인공신경망을 간소화함과 동시에 학습 속도를 향상하였다. NeRF-W[7]는 가변 조명 조건에서 획득된 이미지를 통해서도 안정적인 3D 모델링을 가능하게 하였다. 또한 Mip-NeRF360[8]은 렌더링 과정에서 해상도를 다르게 처리함으로써 360도 장면에 대한 고품질의 복원을 구현하였다.

NeRF와 같이 암시적 신경망에 의존하여 3차원 장면을 렌더링하는 방식은 계산 비용이 높으며 실시간 처리에 제약이 있다. 이러한 한계를 극복하기 위해 공간 정보를 비교적 명시적으로 모델링하는 방법론이 등장했다. 대표적으로 NSVF[9]는 희소한 복셀로 장면을 표현하며 학습 과정에서 점차적으로 복셀을 잘라 나가는 식으로 메모리 사용량을 줄이고 효율성을 높였다. Plenoxel[10]은 암시적 신경망을 사용하지 않고 희소 복셀에 위치한 구면 조화 함수 (spherical harmonics)의 계수를 학습시킨다. 한편, 3DGS[4]는 다수의 타원체 모양의 3차원 가우시안 분포 모델을 사용하여 3차원 장면을 재구성한다. 이때 개별 가우시안의 위치, 회전, 스케일, 색상, 투명도가 명시적으로 모델링된다. 3DGS의 렌더링 알고리즘은 가우시안의 2D 투영과 알파 블렌딩을 통해 진행되기 때문에 GPU를 사용한 실시간 처리가 가능하다는 장점이 있다. 3DGS 또한 다양한 후속 연구가 등장했다. 대표적으로 3DGS 모델로부터 메쉬를 추출하는 알고리즘을 고안한 SuGaR[11], 2DGS[12] 등이 있다.

2. 동적 장면 표현을 위한 Radiance Fields 모델

시간에 따라 변화하는 동적 장면을 표현하는 모델을 개발하는 것은 NeRF와 3DGS를 비롯한 radiance fields 방법론의 주요 응용 분야 중 하나이다. 동적 장면을 표현하기 위해 고안된 radiance fields 모델을 5가지 접근법으로 구분할 수 있다[13]. 첫 번째는 매 프레임의 장면을 개별적으로 처리하여 훈련하는 것이다[14,15,16]. 재생 시점에 맞는 일부 프레임만 전송하더라도 클라이언트에서 재구성을 할 수 있기에 스트리밍에 용이하다는 장점이 있으나, 프레임에 비례하여 파일의 크기가 커지고 프레임 간 일관성 보존이 어렵다는 단점이 있다. 두 번째는 위치와 시간축으로 구성된 4차원 공간에 대해 다수의 저차원 자료구조로 분해하여 표현하는 방법이다[17,18]. 이때 차원 축소의 효과로 고차원 공간을 사용하는 것에 비해 파일 크기 감소의 효과를 얻을 수 있다. 세 번째 접근법은 모든 시간대의 3차원 좌표를 단일 시간대의 표준 (canonical) 필드의 3차원 좌표로 대응시킨 후 표준 3DGS 필드에서 쿼리하는 방법이다[19,20]. 비교적 적은 저장 공간을 사용하여 동적 공간을 모델링할 수 있다는 장점이 있지만, 표준 3DGS에 대응시킬 수 없는, 즉 기준으로 정한 프레임에 존재하지 않는 시각적 요소에 대해 표현이 불가능하다는 한계가 존재한다. 네 번째는 human pose 등의 템플릿을 사전에 입력으로 하여 제한된 조건에서 물체의 움직임을 모델링하는 방식이며[21], 다섯 번째는 시간에 따른 다수의 점 (또는 가우시안 분포)의 변화를 개별적으로 모델링하는 접근법이다[13,22].

본 연구에서 압축 기법을 적용하기 위해 baseline으로 사용한 4D-GS[5]는 앞서 서술 접근법 중 두 번째와 세 번째 접근에 기반한 동적 장면 모델링 기법이다. 4D-GS는 그림 2와 같이 spatial-temporal structure 인코더와 gaussian deformation 디코더 부분으로 구성된다. 인코더는 입력값으로 주어진 기하적 요소 $(x,\ y,\ z)$와 시간적 요소 (t) 좌표에 대한 최종 임베딩을 구하는 과정을 수행하는데, 구체적으로 입력 요소를 다수 해상도 레벨로 구성된 xy, yz, zx, xt, yt, zt의 학습 가능한 신경 복셀로 매핑한다. 파라미터로써 학습되는 6개의 신경 복셀 내 대응값은 아마다르 곱셈 (hadamard product) 및 연결 (concatenation) 과정이 수행된 후 다중 퍼셉트론에 투과하여 최종적인 임베딩을 나타내게 된다. 이후 gaussian deformation 디코더에서는 임베딩을 입력으로 하여 표준 3DGS 집합에서의 가우시안과의 위치, 회전, 스케일의 차분값을 추론하는 신경망으로 구성된다. 이와 같이 4D-GS는 인코더-디코더 구조로써 효과적으로 동적 3차원 공간을 모델링하여 고품질의 영상을 렌더링할 수 있다는 장점이 있다. 하지만 학습 후 자유 시점 렌더링을 위해서 4차원 신경 복셀, 인공신경망, 표준 3DGS 필드 등 많은 요소를 저장해야 하기에 파일 크기가 상대적으로 증

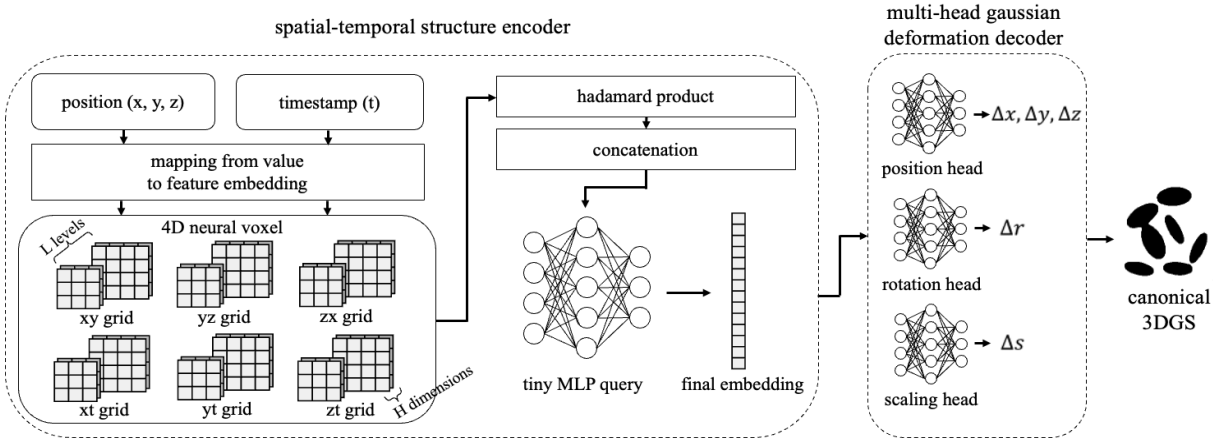


그림 2. 4D-GS 구조도
Fig. 2. Overall pipeline of 4D-GS

가한다는 단점이 있다.

3. 3D Gaussian Splatting의 압축

특정 장면을 표현하기 위해 수십만 개의 3차원 가우시안 분포를 명시적으로 모델링하는 3DGS 모델의 특성상 고용량의 파일 크기는 저장 및 전송에 있어 큰 장벽이 된다. 이를 해결하고자 3DGS를 압축하기 위한 다양한 연구들이 등장했다. 대표적으로 3DGS 압축 방법을 두 가지로 분류할 수 있다[23]. 첫 번째는 고품질의 뷰 합성에 있어 영향력이 크지 않은 불필요한 가우시안의 개수를 감소시키는 접근이며, 두 번째는 개별 가우시안의 속성을 나타내는 데이터의 크기를 감소시키는 접근이다. 같은 가우시안이라도 바라보는 방향에 따라 다른 색상을 출력하기 위해 구면 조화 함수 (spherical harmonics)를 사용하는데, 이때 3개의 degree를 도입할 경우 각 가우시안마다 48개의 파라미터를 저장하여 큰 공간을 차지하게 된다. 대부분의 선행 연구에서는 두 접근법을 동시에 적용하여 3DGS를 압축했다. LightGaussian[24]은 가우시안을 프루닝 (pruning) 하기 위해 각 가우시안이 training view에 렌더링 된 히트 (hit) 횟수, 불투명도, 3D 볼륨을 종합하여 global significance score를 산출하였다. 또한 구면 조화 함수의 계수를 줄이기 위해 knowledge distillation 기법을 사용하여 낮은 degree를 갖는 student 모델을 학습시켰다. 한편 [25]에서는 불필요한 가우시안의 제거를 위해 학습 가능한 볼륨 마스크를 사용했으며 view-dependent한 색상 표현을 위해 사용된 spherical harmonics를 Instant-NGP[6]로 대체하는 방식으로 압축을 진행했다. HAC[26]는 무질서하고 비구조적인 가우시안 필드의 압축을 위해 구조화된 이진 해시 그리드를 도입하였다.

본 연구에서는 [27]을 참고하여 4D-GS 모델의 신경 복셀에 대해 비디오 코덱을 적용하여 부호화하고 [24]를 참고하여 표준 3DGS 필드에 대한 프루닝을 진행하는 파이프라인을 구축했다. 기존 연구는 정적인 장면을 표현하는 3DGS 모델을 압축하는 것에 불과하며 동적 3DGS 모델의 압축 연구는 크게 진전되지 않은 상태라는 점에서도 본 연구에 의의가 있다.

Ⅲ. 4차원 복셀 인코딩과 가우시안 필드의 프루닝 기법

기본 설정값으로 학습이 진행된 4D-GS[5]의 결과물 구성 요소별 파일 크기를 Neural 3D video 데이터셋[28] 중 coffee_martini 시퀀스를 예시로 살펴보면, 전체 파일 크기 41.89MB 중 4차원 신경 복셀 (그림 2의 4D neural voxel)은 9.47MB, 표준 가우시안 필드 (그림 2의 canonical 3DGS)는 30.43MB를 차지한다. 메타데이터, 인공신경망 등 나머지 데이터가 차지하는 1.99MB를 제외하면 두 가지 요소는 전체 모델의 용량 중 95.23%를 차지한다. 이에 본 연구에서는 4D-GS의 가장 큰 용량을 차지하는 두 가지 요소인 4차원 신경 복셀과 표준 가우시안 필드 각각에 대해 압축하기 위한 기법을 개별적으로 제시한다. 이 둘은 모듈화되어 구현되어 있으며, 선택적으로 또는 동시에 적용 가능하다. 첫 번째 압축 기법은 4차원 신경 복셀의 양자화 및 비디오 코덱을 사용한 압축이며, 두 번째는 표준 3DGS 필드의 기여도 점수 기반 프루닝 기법이다. 각각의 방법론에 대해 3-1절과 3-2절에 상세히 서술한다.

1. 4차원 신경 복셀의 양자화 및 비디오 코덱을 사용한 부호화/복호화 파이프라인

그림 2에 묘사된 4D-GS[5] 모델의 spatial-temporal structure encoder를 구성하는 4차원 신경 복셀은, L을 해상도 레벨의 개수라고 하였을 때 $6 \times L$개의 3차원 복셀 그리드로 구성된다. 이는 입력값 x, y, z, t, 즉 4개의 요소로 조합할 수 있는 경우의 수가 ${}_4C_2 = 6$이고 각의 조합마다 해상도 레벨 개수인 L개의 복셀이 존재하기 때문이다. 각 3차원 그리드는 H의 채널을 갖는 2차원 plane들로 구성되며, plane의 높이와 너비를 h, w이라고 할 때 최종적으로 32비트 소수 값으로 구성된 신경 복셀은 shape이 [$6 \times L$, H, h, w]인 텐서 구조를 갖는다고 해석할 수 있다.

그림 2의 spatial-temporal structure encoder와 multi-head gaussian deformation decoder의 파라미터는 4D-GS 학습 과정에서 동시에 업데이트된다. 결과적으로 4D-GS 학습을 마치면 4차원 신경 복셀은 공간상의 좌표 (x, y, z) 및 시간 (t)에 기반한 feature 임베딩을 나타내게 된다. 4차원 신

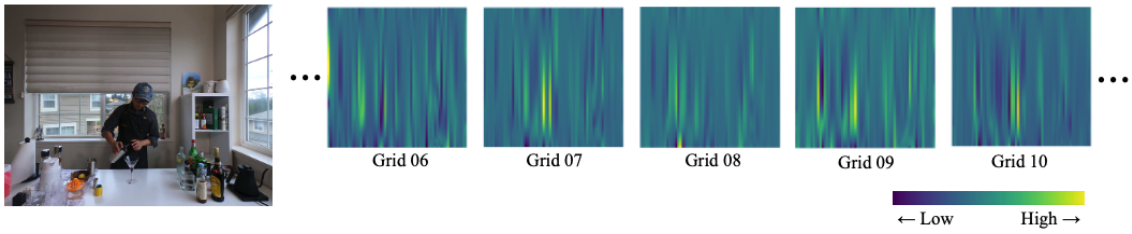


그림 3. (좌) Neural 3d video 데이터셋[28]의 coffee_matini 시퀀스 (우) coffee_martini 시퀀스로 학습된 4차원 신경 복셀 중 일부 평면의 시각화

Fig. 3. (left) coffee_martini sequence of neural 3d video[28] dataset (right) visualized partial feature planes of 4d neural voxel trained from coffee_martini

경 복셀을 2차원 그리드 단위로 분해하여 시각화한 결과는 그림 3의 우측 자료와 같다. 제시된 자료는 matplotlib을 통해 값의 고저를 색상으로 표현한 것이며, 데이터는 단일 채널의 32비트 자료형이다. 그림 3을 통해 4차원 신경 복셀의 두 가지 특징을 살펴볼 수 있는데, 첫 번째는 특정 그리드 내에서 인접한 위치에는 유사한 값이 분포한다는 것이다. 임베딩 값은 대체로 연속된 값의 분포를 갖는 것을 확인할 수 있다. 두 번째는 특정 3차원 복셀 내 인접한 2차원 평면 간 유사한 위치에는 유사한 값의 분포가 나타난다는 것이다. 그림 3에서 한 복셀 내 여러 채널의 그리드에서 동일 좌표에 유사한 값이 분포되어 있는 것으로 이를 설명할 수 있다. 본 연구에서는 이러한 특징을 비디오 코덱의 화면 내 예측 (intra prediction)과 화면 간 예측 (inter prediction)을 적용하는 방식으로 활용하여 신경 복셀을 압축하고자 한다.

32비트 소수 자료형의 4차원 텐서에 비디오 코덱을 적용하기 위해서는 전처리 작업이 필요하다. 수행한 첫 번째 전처리 작업은 2차원 평면으로 분해하는 것이다. 해상도 레벨 수 (L)와 복셀의 채널 수 (H)가 학습 이전에 사전 설정되었을 때, 4개의 차원을 갖는 텐서 자료구조를 먼저 $6 \times L$ 개의 3차원 복셀로 분해하였고, 이를 H개의 채널별 각각 하나의 평면으로 분해하였다. 두 번째 전처리 작업은 양자화이다. 비트수를 32비트에서 16비트 또는 8비트로 감소하였으며, 최댓값과 최솟값에 기반하여 정수 형태로 표현하였다. 기존 값을 x라고 할 때 양자화된 값 $\hat{x}$을 만들기 위해 다음과 같은 수식을 적용한다:

$$\hat{x} = \frac{2^n - 1}{M - m} \times (x - m) \tag{1}$$

n은 목표 비트수를 의미하며, m은 기존 임베딩의 최솟값, M은 기존 임베딩의 최댓값을 의미한다. 이와 같이 양자화를 수행함으로써, 신경 복셀의 임베딩값을 정수화하여 비디오 코덱의 원활한 적용을 가능하게 함과 동시에 양자화 자체로써 데이터의 저장 용량 크기를 감소시킬 수 있는 효과를 얻을 수 있다.

그림 4는 4차원 신경 복셀의 부호화/복호화 절차와 본 연구에서 수행한 실험의 구조도를 나타낸다. 전반적으로 4차원 신경 복셀을 다수의 2차원 평면으로 분할한 후 양자화 및 비디오 코덱을 사용하여 부호화하는 과정을 거친다. 이후 비디오 코덱 디코딩 및 역양자화 과정을 거쳐 원본과 유사한 4차원 신경 복셀로 복원하게 되며, 각 단계의 효과성을 파악하기 위해 실험을 3가지로 나누어 진행한다. 실험 (A)는 양자화만 진행하는 경우이며, 실험 (B)는 양자화 및 화면 내 예측을 통한 압축을 진행하는 경우이며, 실험 (C)는 양자화 이후 복셀에 있는 평면을 시간축으로 연결하여 화면 내 예측과 화면 간 예측을 동시에 사용한 압축을 진행하는 경우이다. 특정 복셀 내에 존재하는 H개의 feature 평면을 개별 단위의 프레임으로 처리하여 첫 프레임을 I 프레임, 나머지 프레임으로 P 프레임으로 두는 low delay 모드로 인코딩하였다. 참고로, 비디오 코덱의 low delay 모드에서 I 프레임은 타 프레임에 대한 참조 없이 독자적으로 부호화되는 프레임이며, P 프레임은 이전 프레임들을 참조하는 화면 간 예측이 일어나는 프레임이다. 실험 (C)의 경우에 실험 (B)와 비교하여 두 가지 이점을 갖는다. 첫 번째는 inter coding 도입으로 인한 압축률 향상이며, 두 번째는 클라이언트의 디코딩 단계에서의 필요한 디코더 수 감소 효과이다. 물론 임베딩을 평면적으로 연결하더라도 디코더

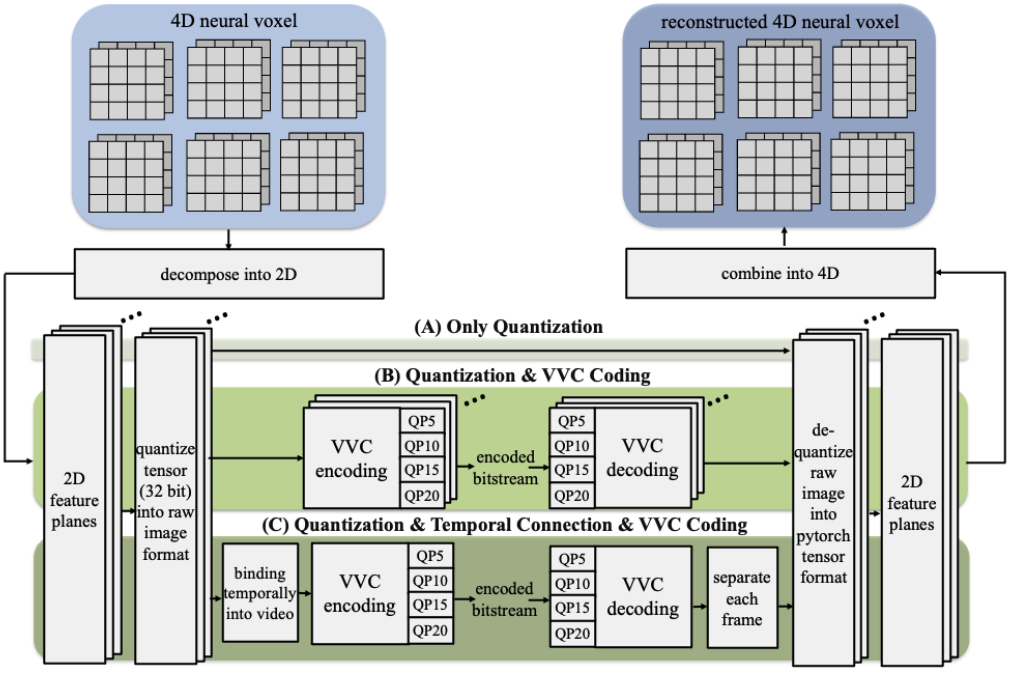


그림 4. 4차원 신경 복셀의 부호화/복호화 실험 파이프라인
Fig. 4. Pipeline of encoding/decoding procedure of 4-dimensional neural voxel

수 감소를 달성할 수 있지만, 임베딩 간 spatial 연결을 수행한 경우에는 실험 결과 압축률 향상에 큰 효과가 없는 것으로 나타났기에 temporal 연결을 사용하였다. 본 연구에서는 기본 4D-GS 모델과 함께 실험 (A), (B), (C) 각각을 비교하여 각 요소별로 압축률 향상의 어떠한 영향을 미치는지 파악하기 위였다. 이를 위해 각 실험 조건마다 압축 시 비트스트림 비트율을 측정하고, 복원 후 4D-GS 모델에 다시 신경 복셀로 삽입하여 해당 조건에서 4D-GS 렌더링 품질을 비교하였다. 동영상 부호화시 입력으로 YUV400 데이터 포맷을 지정하였으며, 비디오 코덱으로는 VVC 표준[29]의 reference software인 VTM[30]을 사용하였고 QP 값을 조정하여 다양한 압축 조건에서 실험하여 진행했다.

2. 표준 가우시안 필드의 프루닝

LightGaussian[24]에 소개된 가우시안 프루닝 기법은 정적 3DGS 필드에서 뷰 렌더링에 기여하는 빈도가 적거나 중요도가 낮은 가우시안을 제거하는 압축 방식이다. 이론상 동적 3DGS 모델인 4D-GS의 구성 요소 중 하나인 표준 3DGS에도 가우시안 프루닝을 적용하여 압축을 시도해볼 수 있다. 하지만 4D-GS[5]는 동적 장면을 표현하는 모델로써, 모든 시간대의 가우시안에 대해 표준 시간의 3DGS에서의 위치, 회전, 스케일에 대응하는 변화값을 추론하는 과정이 포함되기에 절차가 상이하다. 이에 본 연구에서는 4D-GS 내 gaussian deformation decoder를 거친 변형값에 대해 가우시안 프루닝을 적용할 수 있는 파이프라인을 구성 및 구현하였다. 학습 과정부터 압축 알고리즘을 반영하도록 수정해야 하는 타 기법과 달리 학습 완료된 모델에 압축 조건을 선택하여 유동적으로 적용할 수 있도록 모듈화하여 구현하였다. 따라서 3-1절에서 설명한 4차원 신경 복셀의 부호화/복호화 기법과 독립적으로 적용 가능하다.

그림 5의 순서도는 본 연구에 적용한 점수 기반 가우시안 프루닝 기법을 설명한다. 크게 세 가지 방법으로 분류할 수 있는데, 그림 5의 하단을 살펴보면 세 방법 모두 마지막에 옵션으로 설정한 프루닝 비율에 맞추어 전체 가우시안을 포함하는 배열에 대해 마스킹을 하는 절차를 거친다. 프루닝을 수행하는 기준 중 첫 번째는 투명도에 기반한 가우시안 제거 방법 (그림 5의 opacity mode)이다. 기본 3DGS 알

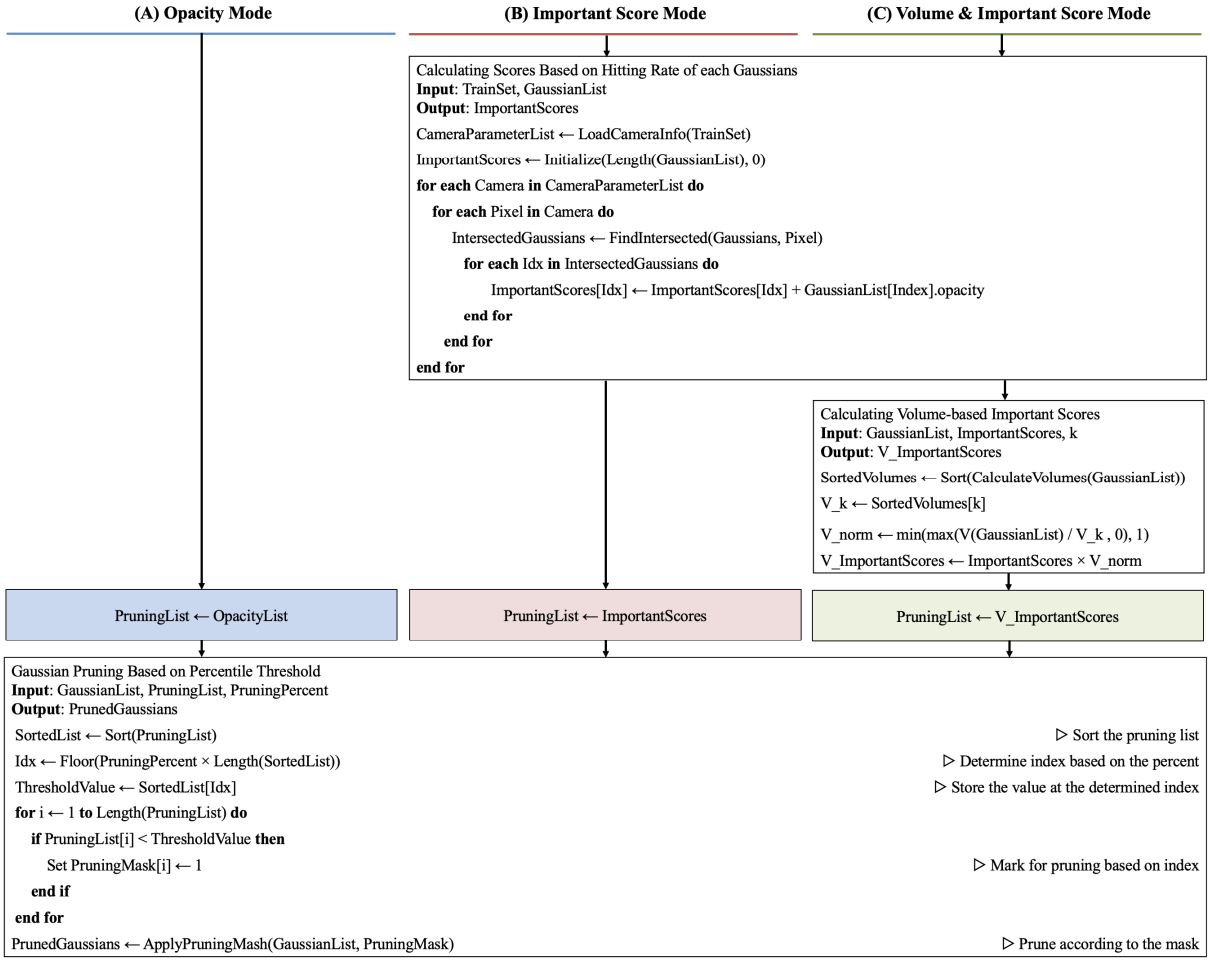


그림 5. 가우시안 필드의 점수 기반 프루닝 기법의 알고리즘 및 세 가지 옵션
Fig. 5. Algorithms of the score-based gaussian pruning technique and its three options

고리즘에서 일정 주기마다 투명도에 기반한 프루닝이 진행되며, 이를 사용할 경우 렌더링 과정에서 알파 블렌딩 수행 시 투명 상태에 가까워 연산 결과에 큰 영향을 주지 못하는 가우시안을 제거할 수 있다는 장점이 있다. 두 번째는 train set에서 출발한 광선에 교차하는 횟수와 투명도를 종합하여 제거할 가우시안을 결정하는 방식이다 (그림 5의 important-score mode). 실제로 렌더링되는 빈도를 반영하여 압축할 수 있다는 장점이 있다. 세 번째는 가우시안이 광선에 교차하는 빈도와 함께 투명도, 가우시안의 부피를 함께 고려하는 방식이다 (그림 5의 volume & important score mode). 두 번째와 세 번째 방식은 압축 알고리즘이 렌더링 과정을 포함한다. 따라서 LightGaussian 소프트웨어를 4D-GS에 그대로 적용할 수 없으며, 이에 본 연구에서는 spatial-temporal encoder와 gaussian deformation decoder를 거친 최종적인 가우시안의 변형 좌표, 변형 회전값, 변형 스케일값에 대해 추론하도록 변경하였다. 이후 그림 5의 세 가지 기법 각각에 대해 프루닝 비율을 조정해가며 개별적으로 압축 실험을 수행한 후 압축률과 품질 변화를 비교 분석하였다.

Ⅳ. 실험 결과

본 연구에서는 압축을 실시하지 않은 기본 4D-GS 모델 및 이를 통해 렌더링한 결과물을 대조군 (baseline)으로 설

표 1. Neural 3D video 데이터셋[28] 6종에 대한 4차원 신경 복셀 부호화 및 복원후 렌더링 실험 결과
Table 1. Experimental results of encoding and rendering of 4D neural voxels on six neural 3d video datasets[28]

		PSNR (dB) ↑	SSIM ↑	LPIPS ↓	4D neural voxel bitrate (Mbps) ↓	total bitrate (Mbps) ↓
baseline		31.41	0.9364	0.1492	7.57	33.83
(A) 8-bit quantization		31.11 (-0.96%)	0.9343 (-0.23%)	0.1500 (+0.59%)	1.89 (-75.01%)	28.14 (-16.79%)
(A) 16-bit quantization		31.41 (+0.00%)	0.9364 (+0.00%)	0.1491 (-0.00%)	3.78 (-50.02%)	30.04 (-11.20%)
(B) 16-bit quantization + VVC encoding	QP 5	31.41 (-0.02%)	0.9364 (-0.01%)	0.1492 (+0.01%)	0.52 (-93.08%)	26.78 (-20.84%)
	QP 10	31.39 (-0.06%)	0.9362 (-0.02%)	0.1492 (+0.05%)	0.42 (-94.48%)	26.67 (-21.15%)
	QP 15	31.36 (-0.16%)	0.9359 (-0.06%)	0.1494 (+0.15%)	0.33 (-95.66%)	26.58 (-21.42%)
	QP 20	31.24 (-0.55%)	0.9350 (-0.16%)	0.1498 (+0.46%)	0.25 (-96.74%)	26.50 (-21.66%)
(C) 16-bit quantization + temporal binding + VVC encoding	QP 5	31.36 (-0.16%)	0.9360 (-0.05%)	0.1493 (+0.11%)	0.34 (-95.46%)	26.60 (-21.37%)
	QP 10	31.27 (-0.45%)	0.9350 (-0.15%)	0.1496 (+0.33%)	0.26 (-96.58%)	26.51 (-21.62%)
	QP 15	31.05 (-1.15%)	0.9330 (-0.37%)	0.1506 (+0.96%)	0.19 (-97.53%)	26.44 (-21.83%)
	QP 20	30.56 (-2.70%)	0.9291 (-0.78%)	0.1527 (+2.41%)	0.13 (-98.20%)	26.39 (-22.00%)

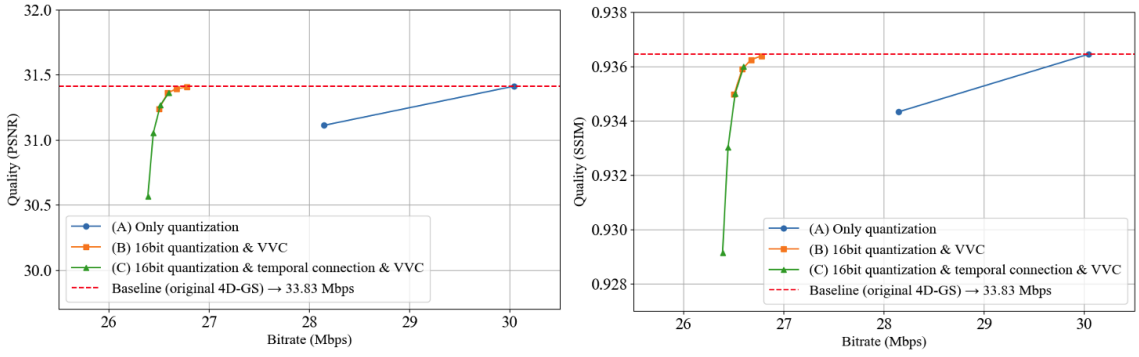


그림 6. 표 1의 결과를 RD-curve로 나타낸 모습. 점선은 압축을 실시하지 않은 기본 4D-GS 모델을 의미 (좌) PSNR (우) SSIM
Fig. 6. RD-curve representation of the results in Table 1. The dashed line represents the baseline 4D-GS model without compression (Left) PSNR (Right) SSIM

정하였다. 학습 횟수 (iteration)는 기본값인 14,000회로 통일하였으며, 해상도 레벨 (L)은 2로 설정하였고 신경 복셀의 채널 수 (H)는 공통적으로 16을 사용하였다. 실험에 사용한 데이터셋은 배경이 있는 동적 장면을 촬영한 실사 다중 뷰 동영상 데이터셋인 neural 3D video[28]이다. 총 6개의 시퀀스로 구성되며, 본 연구에서는 모든 시퀀스에 대해 실험한 후 이들 간의 평균값을 산출하여 비트율 대비 PSNR, SSIM, LPIPS에 대한 RD-curve를 나타내었다. 이때 PSNR, SSIM, LPIPS 지표는 학습되지 않은 뷰 (test view)에 대해 4D-GS 모델로 렌더링 한 후 원본 이미지와 비교하는 방식으로 측정되었다. 4-1절은 3-1절에서 소개한 4차원 신경 복셀의 부호화 및 복호화 과정에 대한 실험 결과를, 4-2절은 가우시안 프루닝 과정에 대한 실험 결과를 소개한다. 이후 4-3절에서 두 모듈을 동시에 적용하였을 때의 결과를 소개하고 타 방법론과 비교하며 마무리한다.

1. 4차원 신경 복셀 압축 모듈의 실험 결과

표 1은 4차원 신경 복셀에 대해 부호화 과정을 거친 후의 비트율과 복호화 과정까지 수행된 후의 4D-GS test view 렌더링 품질을 나타낸 것이고, 그림 6은 이를 RD-curve로 나타낸 결과이다. 결과물을 살펴보면, 4차원 신경 복셀에 대해 8비트로 양자화했을 경우 전체 모델의 비트율을 16.79% 감소시킬 수 있고 16비트로 양자화했을 경우 11.20% 감소시킬 수 있었다. 특히 16비트 양자화를 수행하였을 때 일부 지표 (PSNR, SSIM)의 경우 품질이 오히려 소폭 상승하는 경우도 발생했다. 이는 [31]에 따르면 정규화에 의한 일반화 성능 향상 및 노이즈 억제 효과 때문인 것으로 파악된다. 한편, 4차원 신경 복셀에 대한 16비트 양자화와 함께 VVC 부호화/복호화를 적용한 실험 (B)의 결과를 보면, QP 파라미터를 낮게 설정하였을 때 0.1~0.2% 가량의 PSNR, SSIM, LPIPS 품질 감소로 전체 모델 기준 21.37%의 비트율 감소를 달성했다. 양자화만 진행한 실험 (A)와 비교했을 때 초당 최대 916KB의 파일 크기를 절약할 수 있는 것이 확인되었다. 같은 복셀에 존재하는 다수 차원의 임베딩 평면을 시간축으로 연결하여 화면 간 예측을 적용한 실험 (C)의 결과를 파악했을 때, 모든 평면을 개별적으로 부호화한 실험 (B)와 비교하여 초당 비트율을 23KB 가량 감소시킬 수 있었다. 이때 QP를 5, 10에서는 1% 이하의 PSNR, SSIM 품질 저하와 함께 비트율을 21% 이상 감소시킬 수 있었으나, QP를 높게 설정하였을 경우 PSNR 지표가 30.56dB로 크게 감소하였다. 그림 6 RD-curve의 축적

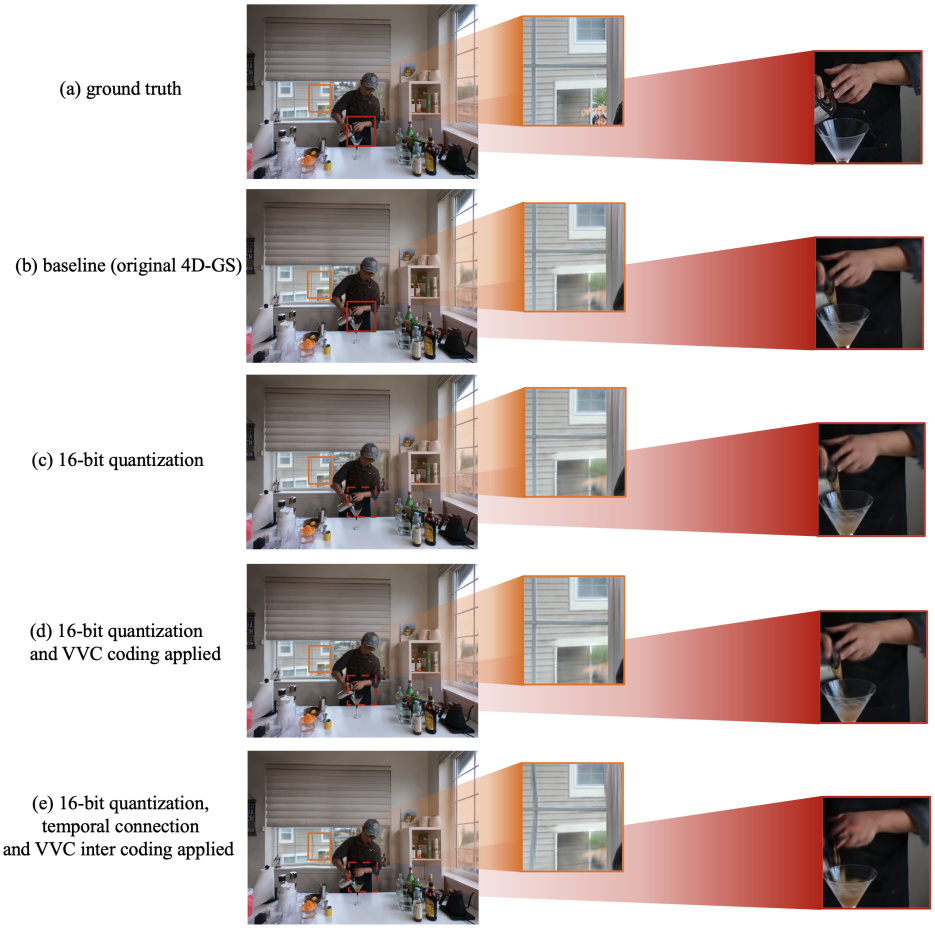


그림 7. 4차원 신경 복셀에 대한 인코딩 실험에 대한 주관적 품질 평가
Fig. 7. Subjective quality assessment of the 4D neural voxel encoding experiments

과 클라이언트의 동영상 디코더의 수를 감소시킬 수 있다는 점을 고려했을 때, 실험 (C)는 충분히 효용 가치가 있을 것으로 파악된다.

그림 7은 실험을 수행한 neural 3d video 데이터셋[28]중 하나인 coffee_martini 시퀀스 실험에 대한 주관적 품질 평가를 나타낸다. (a)는 원본 이미지를 나타내고 (b)는 baseline으로 사용한 무압축 4D-GS 모델로부터 렌더링한 결과를 의미한다. (c)는 16비트 양자화만 진행하였을 경우이며, (d)는 양자화와 함께 2차원 임베딩별 VVC intra 코딩을 적용했을 경우를 나타내며, (e)는 (d)와 유사하지만 복셀 내 존재하는 여러 채널의 임베딩 평면을 시간적으로 연결하여 VVC 코덱을 적용한 경우이다. 관찰 결과 test view의 배경에 존재하는 창문 문틀의 경우 양자화까지 적용한 (c) 실험의 경우까지는 원만히 보존되었으나 동영상 코덱을 적용한 (d), (e) 실험의 경우 4차원 신경 복셀 내 임베딩 값의 오차가 발생하여 복원 시 문틀 부분의 왜곡이 다소 발생하였다. 반면 모든 경우에서 물병 등 유리에 맺히는 빛 반사 등의 현상은 원본과 유사하게 복원하는 것으로 나타났다.

2. 표준 가우시안 필드 프루닝 모듈의 실험 결과

표 2는 4D-GS를 구성하는 표준 3DGS 필드에 대해 그림 5에 소개된 세 가지 프루닝 기법에 의한 실험을 적용한 후 비트율 변화 및 test view 렌더링시 품질 변화를 나타낸다. 두 번째 열에 기재된 pruning 수치는 가우시안을 제거한 비율을 의미한다. coffee_martini 시퀀스로 학습된 원본 4D-GS 모델의 표준 3DGS 필드에 122,685개의 가우시안이 존재하는데, 0.1 비율로 프루닝하였을 경우 10% 제거된 110,416개, 0.7 비율의 경우 70% 제거된 36,806개의 가우시안이 남게 된다. 그림 8은 표 2의 결과를 RD-curve로 나

표 2. Neural 3d video 데이터셋[28] 6종에 대해 실험한 4D-GS 가우시안 프루닝 후 비트율 변화 및 렌더링 실험결과
Table 2. Bitrate changes and rendering results after 4D-GS gaussian pruning on six neural 3d video datasets[28]

		PSNR (dB) ↑	SSIM ↑	LPIPS ↓	canonical 3DGS bitrate (Mbps) ↓	total bitrate (Mbps) ↓
baseline		31.41	0.9364	0.1492	24.02	33.83
(A) Opacity-based pruning	pruning 0.1	31.06 (-1.12%)	0.9345 (-0.20%)	0.1509 (+1.18%)	22.15 (-7.80%)	31.34 (-7.37%)
	pruning 0.3	30.96 (-1.43%)	0.9327 (-0.40%)	0.1526 (+2.99%)	17.29 (-28.01%)	26.48 (-21.72%)
	pruning 0.5	30.43 (-3.11%)	0.9221 (-1.53%)	0.1712 (+14.77%)	12.35 (-48.58%)	21.54 (-36.33%)
	pruning 0.7	28.11 (-10.51%)	0.8788 (-6.15%)	0.2375 (-69.14%)	7.41 (-69.14%)	16.60 (-50.93%)
(B) Important score-based pruning	pruning 0.1	31.06 (-1.30%)	0.9346 (-0.20%)	0.1508 (+1.11%)	22.15 (-7.80%)	31.34 (-7.37%)
	pruning 0.3	31.04 (-1.19%)	0.9342 (-0.24%)	0.1518 (+1.77%)	17.29 (-28.01%)	26.48 (-21.72%)
	pruning 0.5	30.78 (-2.01%)	0.9295 (-0.75%)	0.1611 (+8.01%)	12.35 (-48.58%)	21.54 (-36.33%)
	pruning 0.7	29.66 (-5.57%)	0.9118 (-2.63%)	0.1954 (+30.98%)	7.41 (-69.14%)	16.60 (-50.93%)
(C) Volume and important score-based pruning	pruning 0.1	31.07 (-1.10%)	0.9345 (-0.20%)	0.1508 (+1.10%)	22.15 (-7.80%)	31.34 (-7.37%)
	pruning 0.3	31.03 (-1.22%)	0.9341 (-0.25%)	0.1515 (+1.57%)	17.29 (-28.01%)	26.48 (-21.72%)
	pruning 0.5	30.73 (-2.16%)	0.9283 (-0.87%)	0.1609 (+7.87%)	12.35 (-48.58%)	21.54 (-36.33%)
	pruning 0.7	29.62 (-5.72%)	0.9071 (-3.14%)	0.1981 (+32.82%)	7.41 (-69.14%)	16.60 (-50.93%)

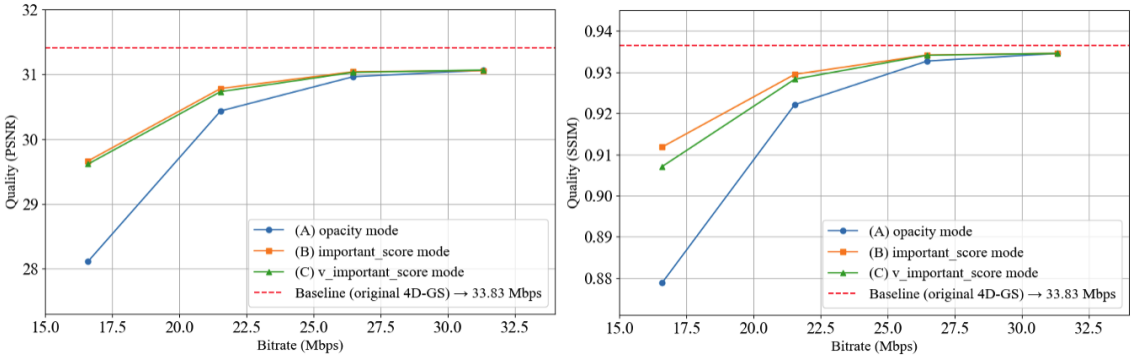


그림 8. 표 2의 결과를 RD-curve로 나타낸 모습. 점선은 프루닝을 실시하지 않은 기본 4D-GS 모델을 의미 (좌) PSNR (우) SSIM
Fig. 8. RD-curve representation of the results in Table 2. The dashed line represents the baseline 4D-GS model without pruning (Left) PSNR (Right) SSIM

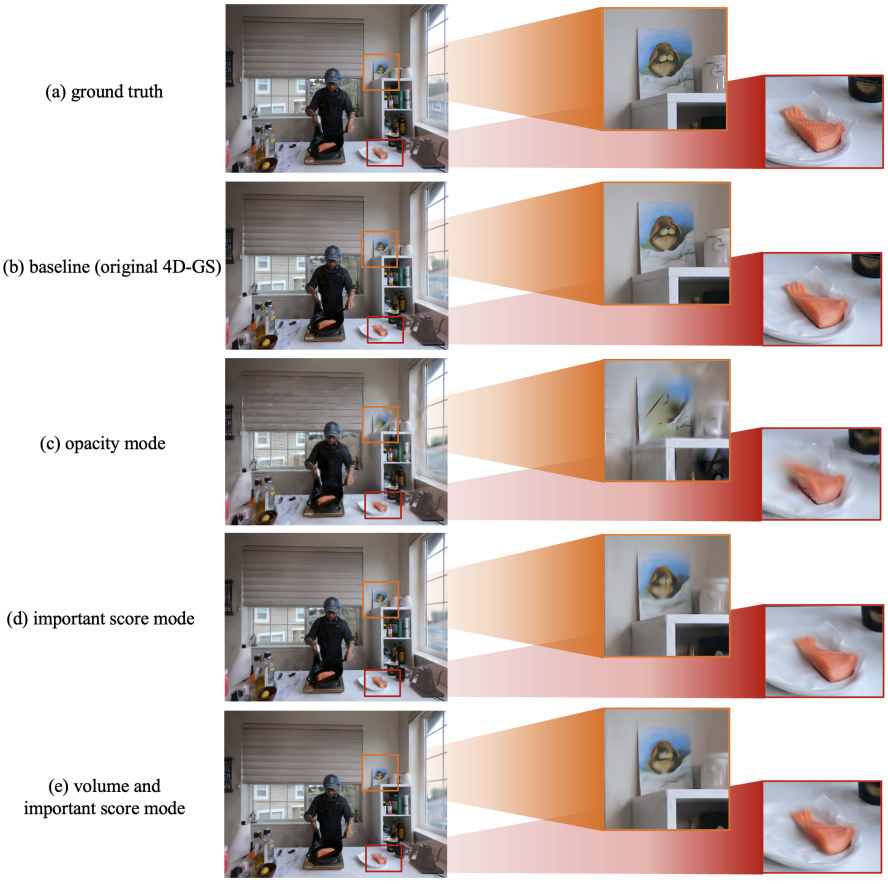


그림 9. 표준 가우시안 필드 프루닝 기법에 대한 주관적 품질 평가 (a) ground truth는 원본 (b) baseline은 무압축 4D-GS 모델을 의미함
Fig. 9. Subjective quality assessment of the canonical 3DGS network pruning technique. (a) ground truth refers to the original, and (b) baseline represents the uncompressed 4D-GS model

타낸 것이다. 전반적으로, 렌더링 파이프라인이 기본 3DGS와 다른 것에도 불구하고 4D-GS의 표준 3DGS 필드에 LightGaussiain 프루닝 기법을 적용하는 것이 효과가 있었음을 파악할 수 있다. 전체 모델의 파일 크기를 36.33% 감소하면서 PSNR 값과 SSIM 값을 원본과 2%대 차이로 유지할 수 있었다. 각 옵션에 대한 차이를 상세히 설명하면, 3-2절에서 소개한 important score 기반 프루닝 방식과 volume & important score 기반 프루닝 방식은 유사한 형태를 띄었으며, 압축률을 향상하였을 경우에는 important score 기반 프루닝 방식이 더욱 좋은 렌더링 성능을 나타내었다. opacity 기반 프루닝 방식은 대체로 낮은 렌더링 품질을 나타내었고, 이는 단순히 투명도만으로 가우시안을 제거하는 것이 test view에 넓은 영역을 나타내는 중요 표현을 저해하는 것이 원인인 것으로 파악된다. 결과적으로 4D-GS의 표준 3DGS 필드를 압축하는 것에 있어 train set에 투영되는 빈도 및 투명도를 동시에 고려하는 important score 기반 프루닝 방식이 가장 효과적인 것으로 파악된다.

그림 9는 flame_salmon 시퀀스에 대해 여러 조건에 대하여 표준 3DGS 필드의 프루닝을 실시한 후 렌더링된 test view에 대한 주관적 품질 평가를 진행한 결과이다. 이때 프루닝을 실시한 (c), (d), (e) 실험에서 프루닝 비율은 0.7로 설정한 상태이다. (c), (d), (e) 모두 장면 내 70%에 해당하는 가우시안을 제거했음에도 불구하고, 전체적인 구조 및 형태는 유지되었다. 하지만 opacity 기반 프루닝을 적용했을 경우 단순히 투명도가 낮은 가우시안을 제거하게 되므로써 그림 7의 액자 및 연어 부분 등 고주파를 나타내는 영역에서 왜곡이 나타났다. (d)와 (e)에서는 비록 디테일이 일부 훼손되긴 했으나 실제로 train set에 렌더링되는 빈도가 낮은 가우시안을 제거함으로써 주요 사물의 구조적인 특징을 유지할 수 있었다.

3. 두 모듈의 공통 적용 및 타 방법론과의 비교

본 절에서는 4-1절과 4-2절에서 개별적으로 효과가 있음을 파악한 4차원 신경 복셀 압축 기법과 표준 3DGS 필드 프루닝 기법을 동시에 적용하였을 때의 결과를 제시함과 동시에 타 논문에서 제시한 3차원 볼륨 렌더링 모델과 성능을 비교한다.

표 3. 제안하는 기법과 기존 4D-GS[5], K-Planes[17], TeTriRF[27]의 성능비교
Table 3. Performance comparison between the proposed method and original 4D-GS[5], K-Planes[17] and TeTriRF[27]

	PSNR (dB) ↑	SSIM ↑	LPIPS ↓	bitrate (Mbps) ↓
K-Planes[17]	31.39	0.9405	0.2117	129.36
TeTriRF[27]	28.71	0.8673	0.3209	5.15
4D-GS[5]	31.41	0.9364	0.1492	33.83
Ours (Low)	30.77	0.9293	0.1602	13.42
Ours (High)	31.10	0.9345	0.1516	19.43

표 3은 4차원 신경 복셀의 인코딩 및 가우시안 프루닝을 동시에 적용한 본 연구의 모델과 유사한 기능을 제공하는 radiance fields 모델인 K-Planes[17], TeTriRF[27]와 비교한 결과이다. 동일한 neural 3d video[28] 데이터셋을 통해 실험되었으며 비트율은 30fps의 프레임률을 갖는다는 조건 하에 측정되었다. 본 연구에서 제안하는 Ours (Low)는 시간적 연결을 수행한 VVC 인코딩을 사용하고 표준 가우시안 필드의 프루닝 비율을 0.5로 설정한 경우이다. 한편 Ours (High)는 개별 임베딩 평면에 대해 독립적으로 VVC 부호화/복호화를 적용하고 프루닝 비율은 0.3을 적용한 경우이다. 두 조건 모두 가우시안 프루닝 시에는 4-2절에서 가장 좋은 결과를 나타낸 important score 모드를 사용하였다. 결과를 살펴보면 압축을 수행하지 않은 4D-GS 모델과 K-Planes 모델이 가장 좋은 품질을 나타냈다. 하지만 K-Planes의 경우에는 129Mbps 이상의 높은 비트율을 나타내며 저장 및 전송에는 부적합한 것으로 파악된다. 5Mbps 대의 비트율로 압축률 측면에서 강점을 보인 TeTriRF 모델은 복원 품질이 PSNR 기준 29dB 미만으로 낮은 모습을 보였다. 본 연구에서 제안하는 4D-GS 압축 기법은 Low 조건과 High 조건 모두 렌더링 품질을 PSNR 기준 31dB 내외로 유지한 채 비트율을 13~20Mbps로 낮출 수 있다는 것에서 품질과 비트율 관점에서 모두 강점을 갖는 것으로 분석할 수 있다.

V. 결 론

본 논문에서는 4D-GS 모델의 구성 요소인 4차원 신경 복셀과 표준 3DGS 필드 각각에 대해 압축하는 기법을 제

시하였고 이를 모듈화하여 선택적으로 적용할 수 있도록 구현하였다. 4차원 신경 복셀의 경우 압축을 위해 2차원 평면화를 진행함과 동시에 정수 범위로 양자화를 진행하고 동영상 코덱을 적용하였다. 표준 3DGS 필드의 경우 투명도와 train view에 투영되는 빈도를 기준으로 가우시안을 제거하였다. 두 모듈을 동시에 적용하였을 경우 기본 4D-GS와 비교하여 품질의 PSNR 기준 0.3dB의 손실, SSIM 기준 0.002의 감소 폭을 동반하며 42.56%의 비트율 감소 효과를 얻을 수 있었다. 후속 연구로 동적 3D 가우시안 모델의 압축을 프레임 조각 단위로 부분적으로 구현하여 고효율의 실사 기반 동적 장면 전송 시스템 구축에 유용하게 활용될 수 있을 것으로 예상한다.

참 고 문 헌 (References)

[1] J. S. Yoon, K. Kim, O. Gallo, H. S. Park, and J. Kautz, "Novel view synthesis of dynamic scenes with globally coherent depths from a monocular camera," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5336 – 5345, 2020. doi: https://doi.org/10.1109/cvpr42600.2020.00538

[2] J. M. Boyce, R. Doré, A. Dziembowski, J. Fleureau, J. Jung, B. Kroon, B. Salahieh, V. K. M. Vadakital, and L. Yu, "MPEG immersive video coding standard," *Proceedings of the IEEE*, vol. 109, no. 9, pp. 1521 – 1536, 2021. doi: https://doi.org/10.1109/JPROC.2021.3062590

[3] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis," *European Conference on Computer Vision (ECCV)*, 2020. doi: https://doi.org/10.1145/3503250

[4] 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, 2023. doi: https://doi.org/10.1145/3592433

[5] G. Wu et al, "4d gaussian splatting for real-time dynamic scene rendering." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024. doi: https://doi.org/10.1109/cvpr52733.2024.01920

[6] T. Mü ller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics (ToG)*, vol. 41, no. 4, pp. 1 – 15, 2022. doi: https://doi.org/10.1145/3528223.3530127

[7] Ricardo Martin-Brualla et al. "Nerf in the wild: Neural radiance fields for unconstrained photo collections." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021. doi: https://doi.org/10.1109/cvpr46437.2021.00713

[8] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-nerf 360: Unbounded anti-aliased neural radiance fields," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5470 – 5479, 2022. doi: https://doi.org/10.1109/cvpr52688.2022.00539

[9] L. Liu, J. Gu, K. Zaw Lin, T.-S. Chua, and C. Theobalt, "Neural sparse voxel fields," *Advances in Neural Information Processing Systems*, vol. 33, pp. 15651 – 15663, 2020.

[10] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5501 – 5510, 2022. doi: https://doi.org/10.1109/cvpr52688.2022.00542

[11] A. Guédon and V. Lepetit, "Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5354 – 5363, 2024. doi: https://doi.org/10.1109/cvpr52733.2024.00512

[12] B. Huang, Z. Yu, A. Chen, A. Geiger, and S. Gao, "2D Gaussian Splatting for Geometrically Accurate Radiance Fields," *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers '24*, Denver CO USA: ACM, Jul. 2024, pp. 1 – 11. doi: 10.1145/3641519.3657428. doi: https://doi.org/10.1145/3641519.3657428

[13] J. Luiten, G. Kopanas, B. Leibe, and D. Ramanan, "Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis," *2024 International Conference on 3D Vision (3DV)*, IEEE, pp. 800 – 809, 2024. doi: https://doi.org/10.1109/3dv62453.2024.00044

[14] L. Li, Z. Shen, Z. Wang, L. Shen, and P. Tan, "Streaming radiance fields for 3d video synthesis," *Advances in Neural Information Processing Systems*, vol. 35, pp. 13485 – 13498, 2022.

[15] L. Wang et al., "Neural Residual Radiance Fields for Streamably Free-Viewpoint Videos," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 76 – 87, 2023. doi: https://doi.org/10.1109/cvpr52729.2023.00016

[16] Z. Yang, H. Yang, Z. Pan, and L. Zhang, "Real-time Photorealistic Dynamic Scene Representation and Rendering with 4D Gaussian Splatting," arXiv: arXiv:2310.10642. Accessed: Mar. 26, 2024. [Online]. Available: http://arxiv.org/abs/2310.10642, Feb. 22, 2024.

[17] S. Fridovich-Keil, G. Meanti, F. R. Warburg, B. Recht, and A. Kanazawa, "K-planes: Explicit radiance fields in space, time, and appearance," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12479 – 12488, 2023. doi: https://doi.org/10.1109/cvpr52729.2023.01201

[18] A. Cao and J. Johnson, "Hexplane: A fast representation for dynamic scenes," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 130 – 141. doi: https://doi.org/10.1109/cvpr52729.2023.00021

[19] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-nerf: Neural radiance fields for dynamic scenes," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern*

Recognition, pp. 10318 – 10327, 2021.
doi: https://doi.org/10.1109/cvpr46437.2021.01018

[20] K. Park et al., "Nerfies: Deformable neural radiance fields," *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5865 – 5874, 2021.
doi: https://doi.org/10.1109/iccv48922.2021.00581

[21] C.-Y. Weng, B. Curless, P. P. Srinivasan, J. T. Barron, and I. Kemelmacher-Shlizerman, "Humannerf: Free-viewpoint rendering of moving people from monocular video," *Proceedings of the IEEE/CVF conference on computer vision and pattern Recognition*, pp. 16210 – 16220, 2022.
doi: https://doi.org/10.1109/cvpr52688.2022.01573

[22] J. Abou-Chakra, F. Dayoub, and N. Sünderhauf, "Particlenerf: A particle-based encoding for online neural radiance fields," *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 5975 – 5984, 2024.
doi: https://doi.org/10.1109/wacv57701.2024.00587

[23] H. Li, S. Li, Y. Liao and L. Yu, "[INVR] Review of 3DGS compression methods," Standard ISO/IEC JTC1/SC29/WG4, MPEG/m67666, 2024.

[24] Z. Fan, K. Wang, K. Wen, Z. Zhu, D. Xu, and Z. Wang, "Light-Gaussian: Unbounded 3D Gaussian Compression with 15x Reduction and 200+ FPS," Mar. 29, 2024, arXiv: arXiv:2311.17245. Accessed: Apr. 16, 2024. [Online]. Available: http://arxiv.org/abs/2311.17245

[25] J. C. Lee, D. Rho, X. Sun, J. H. Ko, and E. Park, "Compact 3d gaussian representation for radiance field," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21719 – 21728, 2024.

[26] Y. Chen, Q. Wu, J. Cai, M. Harandi, and W. Lin, "HAC: Hash-grid Assisted Context for 3D Gaussian Splatting Compression," Apr. 02, 2024, arXiv: arXiv:2403.14530. Accessed: May 12, 2024. [Online]. Available: http://arxiv.org/abs/2403.14530

[27] M. Wu, Z. Wang, G. Kouros, and T. Tuytelaars, "TeTriRF: Temporal Tri-Plane Radiance Fields for Efficient Free-Viewpoint Video," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6487 – 6496, 2024.
doi: https://doi.org/10.1109/cvpr52733.2024.00620

[28] T. Li et al., "Neural 3d video synthesis from multi-view video," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5521 – 5531, 2022.
doi: https://doi.org/10.1109/cvpr52688.2022.00544

[29] B. Bross, et al. "Overview of the versatile video coding (VVC) standard and its applications." *IEEE Transactions on Circuits and Systems for Video Technology*, 31.10 pp. 3736-3764, 2021.
doi: https://doi.org/10.1109/tcsvt.2021.3101953

[30] VVCSoftware_VTM, https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM (accessed Sep. 4. 2024).

[31] K. Zhang, M. Yin, and Y.-X. Wang, "Why Quantization Improves Generalization: NTK of Binary Weight Neural Networks," Jun. 13, 2022, arXiv: arXiv:2206.05916. Accessed: Nov. 01, 2024. [Online]. Available: http://arxiv.org/abs/2206.05916

저 자 소 개

최 재 열

- 2018년 3월 ~ 2024년 2월 : 성균관대학교 컴퓨터교육학과 학사
- 2024년 3월 ~ 현재 : 성균관대학교 인공지능융합학과 석사과정
- 2023년 1월 ~ 2023년 2월 : 한국전자통신연구원 학생연구원
- ORCID : https://orcid.org/0009-0009-2923-1252
- 주관심분야 : 실감미디어, 인공지능, 그래픽스, 멀티미디어 통신 및 시스템

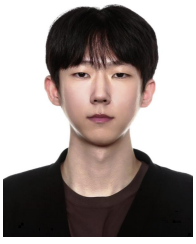
김 영 규

- 2016년 3월 ~ 2024년 8월 : 성균관대학교 중어중문학과 학사
- 2024년 9월 ~ 현재 : 성균관대학교 실감미디어공학과 석사과정
- ORCID : https://orcid.org/0009-0008-5470-3103
- 주관심분야 : 실감미디어, 인공지능, 볼류메트릭 비디오

저 자 소 개

정 종 범

- 2018년 8월 : 가천대학교 컴퓨터공학과 학사
- 2018년 9월 ~ 2019년 8월 : 가천대학교 컴퓨터공학과 석사과정
- 2019년 9월 ~ 현재 : 성균관대학교 컴퓨터교육학과 석박통합과정
- 2020년 1월 ~ 2020년 3월 : University of California, Santa Barbara 방문연구원
- 2021년 8월 ~ 2022년 1월 : Purdue University 방문연구원
- 2022년 9월 ~ 2023년 8월 : 성균관대학교 글로벌융합학부 강사
- 2023년 9월 ~ 2024년 8월 : 성균관대학교 실감미디어공학과 강사
- ORCID : https://orcid.org/0000-0002-7356-5753
- 주관심분야 : 멀티미디어 통신 및 시스템, 비디오 압축 표준, MPEG immersive video, video-based dynamic mesh coding



박 준 형

- 2018년 3월 ~ 2024년 2월 : 성균관대학교 영상학과 학사
- 2024년 3월 ~ 현재 : 성균관대학교 실감미디어공학과 석사과정
- 2023년 7월 ~ 2023년 8월 : 한국전자통신연구원 학생연구원
- ORCID : https://orcid.org/0009-0000-6524-1559
- 주관심분야 : 실감미디어, 인공지능, 그래픽스, 멀티미디어 통신 및 시스템

류 은 석

- 1999년 8월 : 고려대학교 컴퓨터학과 학사
- 2001년 8월 : 고려대학교 컴퓨터학과 석사
- 2008년 2월 : 고려대학교 컴퓨터학과 박사
- 2008년 3월 ~ 2008년 8월 : 고려대학교 연구교수
- 2008년 9월 ~ 2010년 12월 : 조지아공대 박사후과정
- 2011년 1월 ~ 2014년 2월 : InterDigital Labs Staff Engineer
- 2014년 3월 ~ 2015년 2월 : 삼성전자 수석연구원/파트장
- 2015년 3월 ~ 2019년 8월 : 가천대학교 컴퓨터공학과 조교수
- 2019년 9월 ~ 2023년 6월 : 성균관대학교 컴퓨터교육과 부교수
- 2023년 7월 ~ 현재 : 성균관대학교 실감미디어공학과 부교수
- ORCID : https://orcid.org/0000-0003-4894-6105
- 주관심분야 : 멀티미디어 통신 및 시스템, 비디오 코딩 및 국제 표준, HMD/VR 응용분야