

일반논문 (Regular Paper)

방송공학회논문지 제30권 제5호, 2025년 9월 (JBE Vol.30, No.5, September 2025)

<https://doi.org/10.5909/JBE.2025.30.5.757>

ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

동적 메쉬의 움직임 벡터 예측을 위한 적응적 예측자 결정 방법

김 영 육^{a)}, 변 주 형^{a)}, 심 동 규^{a)†}

Adaptive Motion Vector Predictor Selection Method for Dynamic Mesh Compression

Youngwook Kim^{a)}, Joohyung Byeon^{a)}, and Donggyu Sim^{a)†}

요 약

본 논문에서는 동적 메쉬 압축 과정에서 베이스 메쉬의 움직임 벡터 예측 효율을 높이기 위한 적응적 예측자 결정 방법을 제안한다. 기존 방법은 예측자 선정 시 스kip 모드로 복원되어 움직임 벡터의 오차가 존재할 수 있는 예측자가 포함되는 한계가 있는 반면, 제안하는 방법은 현재 정점과 인접 정점 간의 위치 관계를 고려하여 예측자를 결정하고, 이 과정에서 스kip 모드로 복원되거나 중복되는 예측자를 배제함으로써 예측 오차와 편향을 줄여 보다 효율적인 예측을 가능하게 한다. 실험 결과, MPEG V-DMC TMM v12.0 대비 LD 환경에서 시간 증가 없이 포인트 클라우드 기반 D1, D2 PSNR BD-rate가 각각 -0.09%, -0.12%, Luma는 -0.15%로 성능이 개선되었으며, 베이스 메쉬의 인터 프레임 매칭 조건에서 D1 -0.28%, D2 -0.23%, Luma -0.26%의 BD-rate 감소를 확인하였다.

Abstract

In this paper, we propose an adaptive predictor selection method to improve motion vector prediction for base mesh compression in dynamic mesh processing. Unlike conventional methods, which may include predictors with errors due to skip mode decoding, the proposed method selects predictors based on spatial relationships between the current and neighboring vertices. Predictors decoded by skip mode or duplicates are excluded to reduce prediction errors and bias. Experimental results show that, compared to MPEG V-DMC TMM v12.0 under LD conditions, the proposed method improves compression efficiency without increasing processing time. Specifically, point cloud-based D1 and D2 PSNR BD-rate reductions of -0.09% and -0.12%, and Luma PSNR BD-rate reduction of -0.15% are achieved. When inter-frame base mesh matching is applied, BD-rate reductions reach -0.28% for D1, -0.23% for D2, and -0.26% for Luma.

Keyword : V-DMC, Mesh Compression, Motion Vector, Triangle centroid

a) 광운대학교 컴퓨터공학과(Kwangwoon University)

† Corresponding Author : 심동규(Donggyu Sim)

E-mail: dgsim@kw.ac.kr

Tel: +82-2-941-6470

ORCID: <https://orcid.org/0000-0002-2794-9932>

※ 이 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원-학·석사연계ICT핵심인재양성의 지원(IITP-2025-RS-2022-00156215)과 2024년도 우수연구자지원사업 지원을 받아 수행된 연구임

· Manuscript July 20, 2025; Revised August 27, 2025; Accepted August, 27, 2025.

I. 서 론

최근 증강현실(Augmented Reality, AR), 가상현실(Virtual Reality, VR), 컴퓨터 그래픽, 게임, 디지털 콘텐츠 제작 등 다양한 응용 분야에서 3차원(3D) 데이터의 활용이 급격히 증가하고 있다. 이에 따라 3차원 데이터의 효율적인 생성, 표현, 처리 및 전송 기술에 대한 연구가 매우 활발히 진행되고 있다. 그 중 3차원 데이터를 효과적으로 표현하는 대표적인 방식으로 메쉬 구조가 널리 사용된다. 메쉬는 3차원 객체의 표면을 다각형, 주로 삼각형 또는 사각형으로 근사하여 표현함으로써, 원본의 복잡한 형상을 효율적으로 재현할 수 있다. 메쉬 데이터는 각 정점의 위치 좌표와 같은 기하 정보, 정점 간 연결 관계를 나타내는 토폴로지 정보, 표면의 시각적 특성을 저장하는 2차원 이미지 형태의 텍스처 맵, 그리고 메쉬 표면과 텍스처 맵 간의 매핑 관계를 정의하는 텍스처 좌표 등으로 구성된다. 이러한 메쉬 데이터는 높은 표현력과 효율성으로 인해, 실사 기반 3D 스캔, 모션 캡처, 실시간 렌더링, 객체 인식, 의료 영상 등 다양한 분야에서 핵심적인 역할을 담당한다. 특히, 실시간 상호작용이 중요한 AR/VR 환경이나 온라인 게임, 원격 협업 시스템 등에서는 3차원 객체의 움직임이나 형태 변화, 표면 속성 등이 시간에 따라 동적으로 변화하는 경우가 많다. 메쉬를 구성하는 요소(정점 위치, 토폴로지, 텍스처 등) 중 하나 이상이 시간에 따라 변하는 경우를 동적 메쉬(dynamic mesh)라고 하며^[1], 이들은 정적 메쉬(static mesh)에 비해 훨씬 더 많은 데이터가 발생되므로 효율적인 압축 및 전송 기법이 요구된다.

동적 메쉬 데이터는 2차원 이미지인 텍스처 맵과 더불어 기하 정보를 함께 포함하는 데이터 형식으로써 실시간으로 처리하거나 효율적으로 저장하기 위해서는 압축 기술의 적용이 필수적이다. 이러한 기술적 요구에 대응하여, 국제 표준화 기구인 ISO/IEC JTC1 SC29 WG7 MPEG의 3DGH(3D Graphics and Haptics Coding) 서브 그룹에서는 3차원 동적 메쉬 데이터의 효율적 압축을 목표로 V-DMC(Video-based Dynamic Mesh Coding) 표준화 작업을 추진하고 있다. 2021년 10월 동적 메쉬 압축 기술에 대한 제안 요청(Call for Proposals, CfP)을 시작으로^[2], 관련된 여러 압축 기술들이 지속적으로 제안되고 성능 평가와 논의가

이루어졌으며, 2025년 7월에는 해당 표준이 FDIS(Final Draft International Standard) 단계에 진입하였고, 2025년 10월에 IS(International Standard) 발행을 목표로 표준화가 진행되고 있다^[3]. V-DMC 표준화 과정에서는 입력 메쉬를 단순화하여 생성한 베이스 메쉬의 부호화, 복원된 베이스 메쉬를 기반으로 단순화한 정점 및 표면 정보를 다시 정밀한 해상도로 복원시키는 세분화(subdivision) 방법과 세분화된 메쉬가 원본 메쉬와 유사한 기하정보로 복원하기 위한 디테일 정보를 표현하는 정점별 변위(displacement) 벡터의 부호화가 중점적으로 논의되었다. 특히, V-DMC 기술에서는 베이스 메쉬를 부호화하는 방법으로써 참조 베이스 메쉬와 현재 베이스 메쉬의 토폴로지가 동일한 일대일 대응 관계인 경우 대응되는 정점 간의 차이값을 부호화하여 현재 베이스 메쉬를 복원하는 움직임 벡터 부호화 방법이 활발히 논의되었다. 이러한 움직임 벡터 부호화 방법은 동적 메쉬의 시간적 연속성을 효과적으로 활용함으로써, 이전에 부호화된 베이스 메쉬를 참조하여 데이터 중복을 줄이고 압축 성능을 크게 향상시킨다.

본 논문에서는 베이스 메쉬의 기하 정보를 부·복호화하는 과정에서, 움직임 벡터 예측 시 현재 정점과 인접 정점 간의 공간적 위치 관계, 움직임 벡터의 복원 모드, 그리고 중복 여부를 모두 고려하여 적응적으로 예측자를 결정하는 방법을 제안한다. 기존의 예측자 선정 방식은 최대 예측자 개수만큼 순차적으로 예측자를 선택하다가, 최대 예측자 개수에 도달하면 새롭게 탐색된 예측자를 마지막 예측자와 교체하는 방식을 사용한다. 반면, 제안하는 방법은 현재 정점과 인접 정점 간의 거리 정보를 활용하여 공간적으로 가까운 정점을 우선적으로 예측자로 선택한다. 또한, 예측자가 한 방향으로 편향되는 것을 방지하기 위해 삼각형의 무게 중심을 활용하여 다양한 방향에서 예측자가 고르게 선정되도록 한다. 그리고, 기존 방법 대비 움직임 벡터의 오차가 존재할 수 있는 스kip 모드로 복원된 예측자와 편향된 결과가 발생할 수 있는 중복되는 예측자가 선택되지 않도록 한다. 이러한 적응적 예측자 결정 방법의 적용을 통해 예측값 결정에서 발생할 수 있는 편향과 예측 오차를 효과적으로 감소시키며, 공간적으로 유사한 움직임 벡터를 기반으로 한 예측을 통해 압축 성능의 향상을 기존 방법과의 비교 실험을 통해 입증한다.

본 논문의 구성은 다음과 같다. 2장에서는 V-DMC 기술과 선행 연구인 베이스 메쉬의 움직임 벡터 부·복호화 방법에 대해 설명한다. 3장에서는 제안하는 예측자 후보 리스트 구성 방법과 부호화 모드 및 중복 여부에 기반한 예측자 결정 방법에 대해 설명한다. 4장에서는 제안하는 방법의 성능을 기존 방법과 비교하여 평가한다. 마지막으로, 5장에서는 본 논문에 대한 결론을 맺는다.

II. 관련 기술

1. Video-based Dynamic Mesh Coding

V-DMC는 원본 메쉬의 다수개의 정점을 단순화하여 정점 수를 줄인 베이스 메쉬를 생성한 뒤, 이 베이스 메쉬를 세분화하고 원본 메쉬와의 피팅 과정을 거쳐 얻은 변위 벡터를 활용해 동적 메쉬의 압축 및 복원을 수행하는 표준 기술이다. 부호화 과정에서 수행되는 일련의 단계들은 전처리 과정, 베이스 메쉬 부호화, 변위 벡터 부호화 그리고 텍스처 맵 부호화 과정으로 이뤄지며, 그림 1과 같은 구조를 가진다.

전처리 과정에서는 메쉬 단순화, 아틀라스 매개변수화, 세분화 표면 피팅 과정이 이뤄진다. 메쉬 단순화 과정에서

는 입력 메쉬의 정점 개수를 줄여 베이스 메쉬를 생성한다. 이후 아틀라스 매개변수화 과정에서 단순화 과정을 통해 생성된 베이스 메쉬에 맞게 텍스처 좌표를 재계산하고, 마지막으로 세분화 표면 피팅 과정에서는 메쉬 세분화 기술을 사용하여 베이스 메쉬를 세분화하고, 원본 메쉬와 유사한 기하정보를 가지도록 세분화된 각 정점에서 피팅하는 과정이 수행된다.

베이스 메쉬 부호화 과정에서는 베이스 메쉬의 예측 모드에 따라 정적 메쉬 부호화 또는 움직임 벡터 부호화가 수행된다. 예측 모드가 화면 내 예측(Intra prediction)으로 결정된 경우, 정적 메쉬 부호화가 수행된다. 이 과정에서는 베이스 메쉬의 정점 위치 정보, 정점 간 연결 정보, 그리고 정점과 텍스처 맵 간의 매핑 정보를 EdgeBreaker^[4] 기법을 이용하여 부호화된다. 반면, 예측 모드가 화면 간 예측(Inter prediction)으로 결정된 경우, 움직임 벡터 부호화가 수행된다. 다만, 이 과정은 다음 조건들이 모두 충족될 때에만 적용된다. 참조 베이스 메쉬와 현재 베이스 메쉬의 정점 수가 동일하고, 정점 간의 연결 관계가 일치하며, 정점과 텍스처 맵 간의 매핑 정보가 동일해야 한다. 이러한 조건이 모두 만족될 경우, 두 베이스 메쉬는 일대일 대응 관계에 있다고 정의하여 움직임 벡터 부호화 과정이 수행될 수 있다. 움직임 벡터 부호화 과정에서는 참조 베이스 메쉬와 동일한 연결 정보와 텍스처 맵 매핑 정보를 재사용하고, 정점별 위치

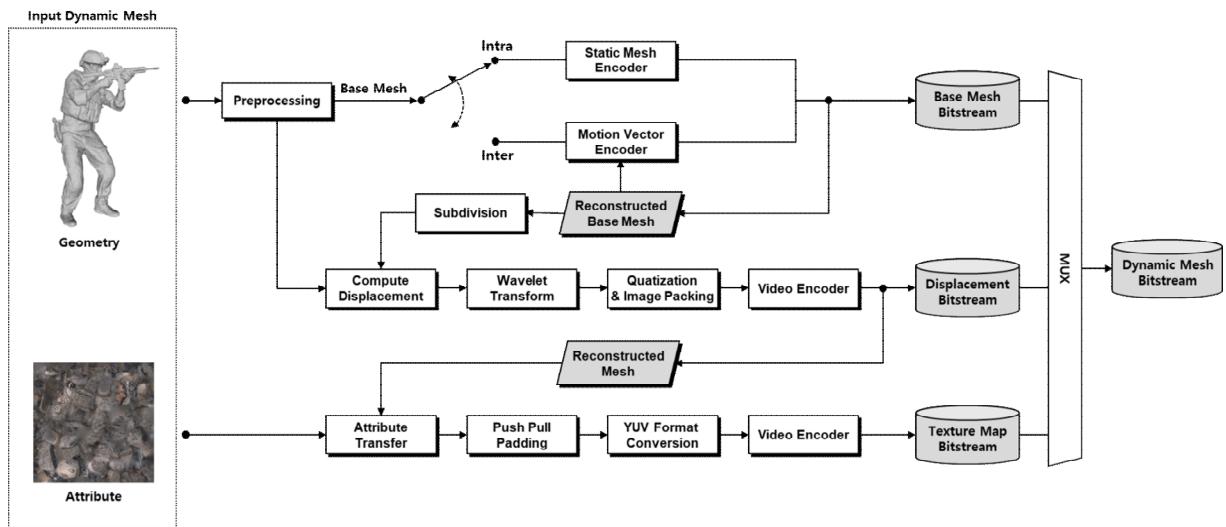


그림 1. V-DMC 부호화기 블록도

Fig. 1. Block diagram of V-DMC encoder

차이에 해당하는 움직임 벡터만을 계산하여 부호화를 수행한다.

변위 벡터 부호화 과정에서는 베이스 메쉬의 부호화기와 복호화기를 통해 복원된 베이스 메쉬에 세분화 과정을 적용하여 세분화된 정점들을 생성하고, 이 세분화된 정점별로 피팅 과정을 통해 전처리된 메쉬와 위치 차이를 계산하여 변위 벡터를 계산한다. 변위 벡터는 에너지 집중도를 높이기 위해 Normal, Tangential, Bi-tangential 방향으로 구성된 지역 좌표계로 변환된다. 변환된 변위 벡터는 세분화 레벨(Level of Detail, LoD)별로 중간 위치 세분화, Loop 기반 세분화^[5], 노말 벡터 기반 세분화^[6] 중 하나의 방식으로 세분화된다. LoD 단위로 생성된 변위 벡터에 대해서는 리프팅 기반 웨이블릿 변환이 수행되어 변환 계수가 생성된다. 생성된 변환 계수들은 양자화 과정을 거친 뒤, 2D 이미지 형태로 패킹되어 HEVC(High Efficiency Video Coding)^[7] 또는 VVC(Versatile Video Coding)^[8]와 같은 비디오 코덱을 통해 부호화된다. 텍스처 맵 부호화 과정에서는 원본 텍스처 맵을 복원된 메쉬에 맞게 보정하는 텍스처 전이 알고리즘을 수행한다. 이 과정을 통해 생성된 텍스처 맵은, 텍스처가 존재하는 영역과 존재하지 않는 영역 사이의 불연속성을 줄이기 위해, 푸쉬-풀 패딩^[9] 과정을 통해 다운샘플링과 업샘플링을 반복하며 빈 공간을 보간한다. 이후 패딩이 완료된 텍스처 맵은 YUV 포맷으로 변환되어, 비디오 코덱을 통해 부호화된다.

V-DMC의 복호화 과정에서는 베이스 메쉬, 변위 벡터, 텍스처 맵에 해당하는 비트스트림을 복호화하여 기하 정보와 속성 정보를 복원하며, 그림 2와 같은 처리 구조를 따른다. 베이스 메쉬의 예측 모드가 화면 내 모드인 경우에는 정적 메쉬 복호화 과정을 수행하며, 화면 간 모드인 경우에는 움직임 벡터 복호화 과정을 통해 베이스 메쉬를 복원한다. 변위 벡터는 비디오 복호화기를 통해 2D로 패킹된 이미지를 복원한 후, 리프팅 기반 웨이블릿 역변환을 수행하여 복원한다. 복원된 변위 벡터는, 복원된 베이스 메쉬에 부호화기에서와 동일한 세분화 과정을 적용하여 생성된 세분화 정점에 적용되어 최종 기하 정보를 복원한다. 텍스처 맵은 비디오 복호화기를 통해 YUV 포맷으로 복원한 후, RGB 포맷으로 변환되어 최종 텍스처 맵을 복원한다.

2. 베이스 메쉬의 움직임 벡터 부호화 방법

베이스 메쉬 부호화 방식은 크게 정적 메쉬 부호화와 움직임 벡터 부호화 두 가지로 나뉜다. 정적 메쉬 부호화는 베이스 메쉬의 정점 위치 정보, 정점 간 연결 관계, 그리고 정점과 텍스처 맵 간의 매핑 좌표를 모두 압축 대상으로 포함하기 때문에, 전송해야 할 데이터 양이 많아지는 특성이 있다. 반면, 움직임 벡터 부호화는 참조 베이스 메쉬의 정점 위치에서 현재 베이스 메쉬의 정점 위치까지의 이동

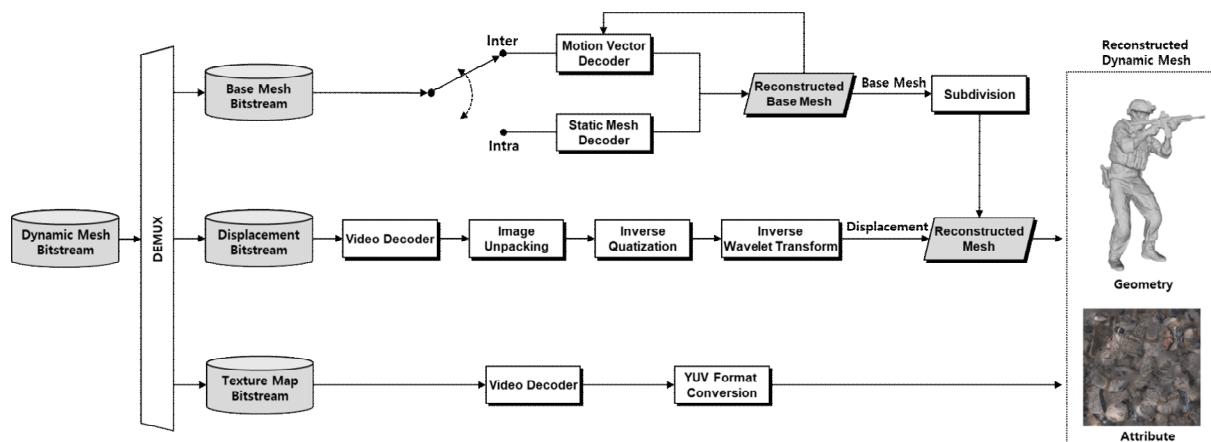


그림 2. V-DMC 복호화기 블록도
Fig. 2. Block diagram of V-DMC decoder

정보만을 전송하므로, 데이터 양을 크게 줄일 수 있다. 그러나 움직임 벡터 부호화를 수행하기 위해서는 현재 베이스 메쉬가 참조 베이스 메쉬와 동일한 구조를 갖는다는 제약 조건이 필요하다^[10]. 즉, 정점 수, 정점 간의 연결 관계, 텍스처 맵 매핑 좌표가 참조 메쉬와 동일해야 한다. 이 조건이 충족될 경우, 참조 베이스 메쉬의 구조적 정보를 그대로 활용하고 정점 위치 변화만을 움직임 벡터로 부호화하면 되기 때문에 베이스 메쉬를 효과적으로 압축할 수 있다. 이에 따라 움직임 벡터 부호화 과정에서는 각 정점의 움직임 벡터만을 압축 대상으로 하며, 해당 벡터는 참조 베이스 메쉬와 현재 베이스 메쉬의 정점 위치 간의 차분으로 산출되고, (x, y, z) 축 성분으로 구성되며 각 성분은 정수형으로 표현된다. 각 정점의 움직임 벡터는 정적 메쉬 부호화 과정에서 결정된 순서와 동일한 순서로 부호화가 수행된다.

움직임 벡터를 부호화하기에 앞서, 중복 정점 처리 과정이 먼저 수행된다. 중복 정점은 동일한 위치에 존재하면서 동일한 움직임 벡터를 가지는 정점으로 정의된다. 이러한 경우, 동일한 위치에 있는 정점은 움직임 벡터를 공유하여 복원할 수 있으므로, 중복 정점에 대해서는 움직임 벡터 부호화가 생략될 수 있다. 이러한 중복 정점은 플래그를 통해 복호화기에 중복 여부를 알리며, 모든 정점에 대해 플래그를 전송하는 것이 아니라, 동일한 위치에 존재하는 정점들에 대해서만 중복 여부 플래그를 전송한다. 플래그가 1로 설정된 경우, 동일한 위치에서 먼저 복원된 정점과 동일한

움직임 벡터를 갖는다는 의미이며, 복호화기는 해당 움직임 벡터를 사용하여 중복 정점의 움직임 벡터를 복원한다. 반면, 플래그가 0으로 설정된 경우에는 동일 위치의 먼저 복원된 정점과 움직임 벡터가 다르다는 뜻이며, 이 경우 해당 정점의 움직임 벡터는 별도로 시그널링되어 복원된다. 중복되는 정점을 제외한 나머지 정점들에 대해서는 움직임 벡터 부호화 과정이 수행되며, 이때 동적 메쉬의 움직임 특성을 고려하여 부호화를 수행한다. 동적 메쉬의 움직임 특성은 그림 3에서 볼 수 있듯이 크게 두 가지 영역으로 나눌 수 있는데, 하나는 참조 프레임 대비 움직임이 발생되는 영역이며, 다른 하나는 움직임이 거의 없는 정적인 영역이다. 먼저, 움직임이 거의 없는 영역의 경우, 해당 영역에서는 대부분의 움직임 벡터 계수가 0에 수렴하게 된다. 이러한 특성을 활용하여, 해당 정점들에 대해서는 복호화기에서 움직임 벡터를 영벡터로 유도하여 복원하는 스kip(skip) 모드를 통해 부호화를 수행한다^[11]. 스kip 모드는 부호화 생략에 따른 압축 효율 향상 효과를 제공한다. 반면, 움직임이 존재하는 영역에서는 각 정점별로 움직임 벡터가 발생되므로, 해당 정보를 전송해야 한다. 이때, 원본 움직임 벡터를 그대로 부호화할 수도 있으나, 일반적으로는 인접 정점의 복원된 움직임 벡터를 활용한 예측 기반 부호화 방식을 적용한다. 이는 일반적으로 현재 정점과 옛지로 연결된 인접 정점들이 공간적으로 유사한 움직임을 보이는 특성을 활용한 것이다^[12]. 이러한 공간적 상관성을 바탕으로, 현재 정점

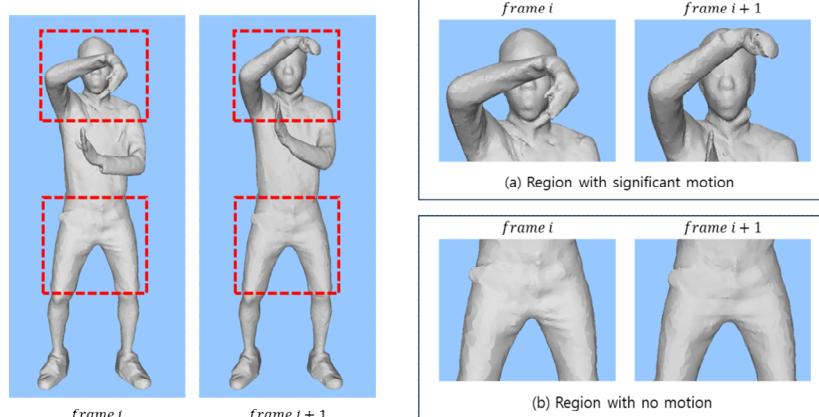


그림 3. 동적 메쉬의 움직임 특성
 Fig. 3. Motion Characteristics of Dynamic Mesh

의 움직임 벡터를 인접 정점의 벡터로부터 예측하고, 그 예측 오차(잔차)만을 부호화함으로써 전송해야 할 데이터의 양을 효과적으로 줄일 수 있다.

V-DMC에서는 영역별 움직임 특성을 반영하여, 움직임 벡터 부호화 모드를 다음과 같이 세 가지 모드로 분류한다. 첫째, 움직임 벡터 전체를 부호화하지 않고 생략하는 전체 스kip 모드(Total Skip Mode), 둘째, 움직임 벡터의 특정 성분만을 선택적으로 생략하는 컴포넌트 스kip 모드(Component Skip Mode), 셋째, 스kip되지 않은 성분에 대해 움직임 벡터 값을 부호화하는 움직임 벡터 값 부호화 모드(Motion Value Coding Mode)이다. 여기서 컴포넌트는 움직임 벡터를 구성하는 각 축(x, y, z 성분)을 의미한다. 이러한 움직임 벡터 부호화 모드를 정점 단위로 전송할 경우 정보량이 많아지므로, 부호화 순서에 따라 다수 개의 정점들을 하나의 움직임 그룹(Motion Group)으로 묶고, 그룹 단위로 부호화 모드를 결정하는 방법을 사용한다^[13]. 각 그룹에 포함되는 정점의 개수(n)는 SPS(Sequence Parameter Set) 내에 정의된 파라미터에 따라 결정되며, 그룹 내 모든 정점은 동일한 부호화 모드를 사용해 움직임 벡터가 부호화된다. 움직임 그룹은 부호화 순서를 기준으로, 중복되지 않는 정점들을 앞에서부터 n개씩 묶어 구성된다. 각 움직임 그룹의 부호화 모드는 다음 절차에 따라 결정된다. 먼저, 전체 스kip 모드 적용 여부를 우선 판단한다. 전체 스kip 모드로 설정된 경우, 별도의 시그널링 없이 그룹에 포함된 모든 정점의 움직임 벡터는 영벡터(0,0,0)로 복원된다. 전체 스kip

모드가 적용되지 않는 경우, 각 축(x, y, z)에 대해 컴포넌트 스kip 여부를 개별적으로 결정한다. 컴포넌트 스kip이 적용된 축의 경우, 해당 축에 대한 움직임 벡터는 그룹 내 모든 정점에서 0으로 복원된다. 컴포넌트 스kip 모드의 결정 순서는 x축, y축, z축 순으로 진행되며, 이때, x축과 y축이 모두 컴포넌트 스kip 모드로 설정된 경우, z축은 컴포넌트 스kip으로 설정하지 않는다. 이는 세 축 모두가 스kip될 경우 결과적으로 전체 스kip 모드와 동일해지기 때문에, 컴포넌트 스kip 모드에서의 중복 방지를 위한 조건이다. 컴포넌트 스kip이 적용되지 않은 축에 대해서는 움직임 벡터 값 부호화 모드가 적용된다. 이때, 해당 축에는 세 가지 예측 모드 중 하나가 선택된다. 예측 모드는 다음과 같이 구분된다. 첫째, 예측 없이 원본 값을 그대로 부호화하는 비예측 모드, 둘째, 인접 정점들의 움직임 벡터 평균을 사용하여 예측하는 평균 예측 모드, 셋째, 평균값 계산 시 반올림을 적용하여 예측값을 생성하는 반올림 평균 예측 모드가 있다^[14]. 이러한 예측 모드는 각 축 단위로 독립적으로 결정된다. 그림 4는 움직임 벡터 부호화 모드에 따른 블록도이다. 각 움직임 벡터 값이 스kip 모드(전체 스kip 모드, 축 스kip 모드)로 결정되면 움직임 벡터에 대해 추가적인 정보를 전송하지 않으며 움직임 벡터 값 부호화 모드로 결정되면 잔차 값을 양자화 과정 없이 전송한다.

움직임 벡터 예측 모드의 적용은 다음과 같이 이루어진다. 각 축에 대해 선택된 예측 모드가 비예측 모드인 경우, 해당 축의 움직임 벡터 값은 별도의 예측 과정 없이 그대로

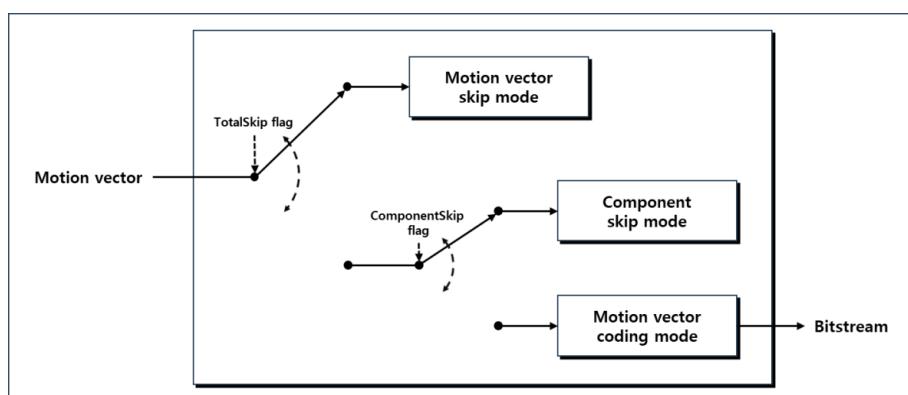


그림 4. 움직임 벡터 부호화 모드에 따른 블록도

Fig. 4. Motion vector encoder block diagram according to motion vector encode mode

잔차 신호로 간주되어 부호화된다. 반면, 선택된 예측 모드가 평균 예측 모드 또는 반올림 평균 예측 모드인 경우에는 ‘오류! 참조 원본을 찾을 수 없습니다.’와 같은 절차에 따라 움직임 벡터를 부호화한다. 두 모드는 예측값 결정 과정에서 차이가 존재하며 다른 과정에서는 같은 과정이 수행된다. 먼저, 각 정점에 대해 인접 정점들 중에서 예측에 사용될 예측자를 선정한다. 이때 예측자는 예측값 결정을 위해 선택된 인접 정점으로 정의된다. 각 정점의 예측값은 선정된 예측자의 움직임 벡터를 통해 계산하고, 계산된 예측값과 실제 움직임 벡터 간의 차이를 산출하여 잔차 신호를 생성한다. 생성된 잔차 신호는 양자화 과정을 거치지 않고 엔트로피 부호화를 통해 비트스트림으로 전송된다. 즉, 움직임 벡터가 예측 모드를 통해 부호화되는 경우, 양자화 과정이 적용되지 않으므로 움직임 벡터에 오차가 발생하지 않는다. 반면, 스kip 모드에서는 움직임 벡터를 전송하지 않고 0으로 복원하게 되는데, 이 경우 스kip 모드로 복원된 움직임 벡터는 실제 움직임 벡터와 차이가 발생할 수 있어 오차가 포함될 수 있다.

구체적으로, 각 정점의 예측자 선정 과정은 그림 5에서 볼 수 있듯이, 예측 수행에 앞서 복원된 예측자를 확인하는 단계와 예측자를 최종적으로 결정하는 단계로 구성된다. 먼저, 예측자 확인 단계에서는 현재 정점의 인접 정점들 중

이미 움직임 벡터가 복원된 정점만을 선별한다. 다음으로, 예측자 결정 단계에서는 선별된 인접 정점들 중 최대 예측자 개수만큼 예측자를 선정한다. 만약 선별된 인접 정점의 수가 최대 예측자 개수 이하이면, 해당 정점들을 모두 예측자로 사용한다. 반면, 선별된 인접 정점의 수가 최대 예측자 개수를 초과할 경우, 정해진 순회 순서(참조 베이스 메쉬의 EdgeBreaker 알고리즘에 따라 정의)에 따라 예측자를 선정하는데, 최대 예측자 개수에 도달한 경우에는 예측자에서 가장 마지막에 추가된 예측자를 새로운 예측자로 교체하는 방식으로 예측자를 결정한다. 이 과정을 그림 6의 예시로 설명하면 다음과 같다. f 정점의 움직임 벡터를 예측하기 위해 인접 정점들을 탐색하는 과정에서, 예측자 확인 단계에서는 아직 부호화가 수행되지 않은 g 정점을 제외시킨다. 이후, 나머지 a, b, c, d, e 정점에 대해 순서대로 예측자 결정 과정을 진행한다. 처음 a, b, c 정점이 예측자로 선택될 때는 최대 예측자 개수(3개)를 초과하지 않아 모두 예측자에 포함된다. d 정점이 추가될 때는 기존의 c 정점이 제거되고, 이어서 e 정점이 추가되면 d 정점이 제거되어 최종적으로 a, b, e 정점이 f 정점의 예측자로 선정된다. 이러한 방식은 현재 정점과 인접 정점 간의 위치 관계나 예측자의 부호화 모드를 고려하지 않고, 단순히 예측자 개수 제한과 추가 순서만을 기준으로 예측자를 선정하기 때문에, 현재 정점

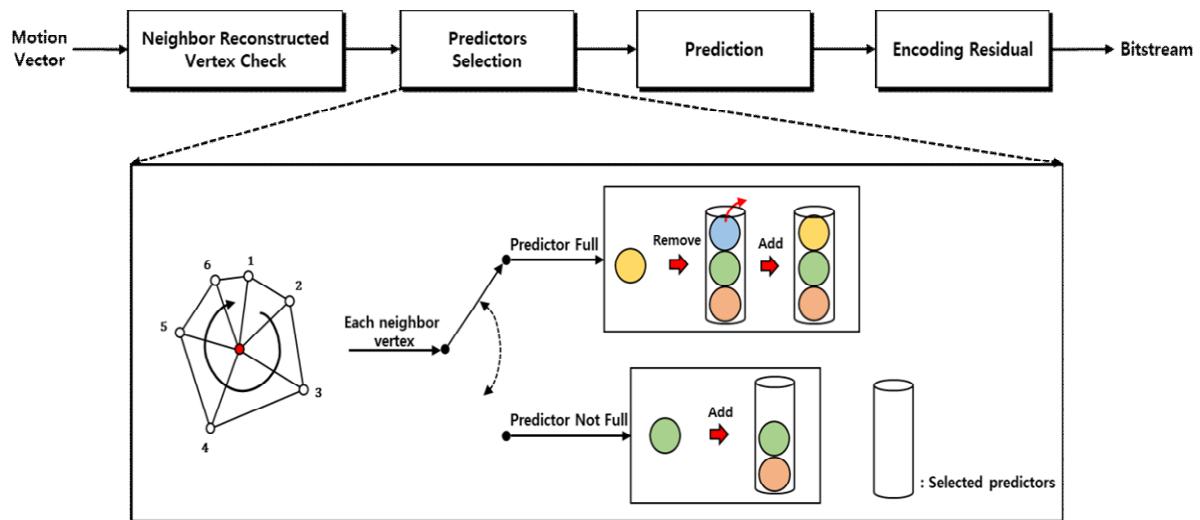


그림 5. V-DMC의 움직임 벡터 예측 모드에 대한 부호화기 블록도
 Fig. 5. V-DMC Encoder Block Diagram for Motion Vector Prediction Mode

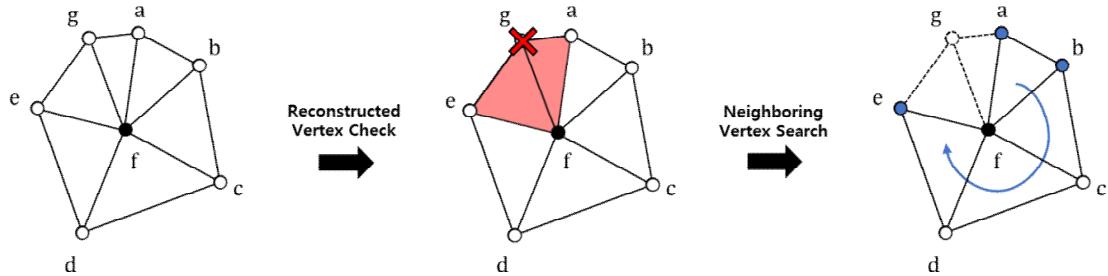


그림 6. V-DMC의 움직임 벡터 예측을 위한 예측자 결정 방법 예시

Fig. 6. Example of Predictor Selection Process in V-DMC Motion Vector Prediction

을 예측하기 위한 최적의 예측자 선택 방식으로 보기 어렵다. 특히, 예측자 선정 시 인접 정점의 움직임 벡터 부호화 모드를 고려하지 않기 때문에, 스kip 모드로 복원된 정점의 움직임 벡터가 예측자에 포함될 수 있다. 이러한 경우, 스kip 된 움직임 벡터 값이 예측값 결정 과정에 포함되기에 예측 정확도를 저하시킨다.

예측값 결정 과정에서는 앞서 선정된 예측자들의 움직임 벡터 값을 이용하여 예측값을 계산하고, 이를 현재 정점의 움직임 벡터와 차분하여 잔차 신호를 생성한다. 예측값은 현재 정점의 예측 모드에 따라 산출 방식이 달라질 수 있다. 현재 정점의 예측 모드가 평균 예측 모드인 경우, 예측값은 식 (1)과 같이 계산되며, 반올림 평균 예측 모드인 경우에는 예측값이 식 (2)와 같이 계산된다. 식 (1)과 식 (2)에서 $V_i[k]$ 는 i번째 정점의 k번째 성분의 예측값, p 는 i번째 정점의 예측자 수, $mv_j[k]$ 는 j번째 예측자의 k번째 성분의 움직임 값을 나타낸다.

$$V_i[k] = \frac{\sum_{j=0}^{p-1} mv_j[k]}{p} \quad (1)$$

$$V_i[k] = \frac{(\sum_{j=0}^{p-1} mv_j[k]) + (p \gg 1)}{p} \quad (2)$$

III. 제안하는 방법

V-DMC 기술에서는 움직임 벡터 예측을 위한 예측자를 결정하는데 있어서 현재 정점과 인접 정점 간의 위치 관계

를 반영하지 않고 단순히 개수와 순서에 따라 예측자를 선정한다. 또한 예측자 결정 과정에서 스kip 모드로 복원된 움직임 벡터가 포함될 수 있는 한계가 있다. 이러한 문제로 인해 예측 정확도가 저하되고, 예측 오차가 발생할 수 있다. 본 논문에서는 이러한 기존 방식의 문제점을 개선하기 위한 새로운 예측자 결정 방법을 제안한다. 이 방법은 움직임 벡터 예측의 정확도 향상을 위해, 현재 정점과 인접 정점의 연결 관계 및 거리, 예측자의 부호화 모드와 예측자의 중복 여부를 기반으로 적응적으로 예측자를 결정하는 방법을 포함한다. 제안하는 방법에서는 그림 7과 같은 절차에 따라 움직임 벡터 예측을 수행하며, 예측자를 결정하는 과정에서 현재 정점과 가장 가까운 인접 정점을 우선적으로 예측자 결정에서 고려하며, 선택되는 예측자들의 공간적 다양성을 위해 무게 중심에 따라 예측자 후보 리스트를 구성하고, 리스트 내에서 예측자 후보의 스kip 모드 여부와 중복 여부를 제외한 후 예측자를 결정한다.

구체적으로 본 논문에서 제안하는 예측자 후보 리스트의 구성 과정에서는 현재 정점과 인접 정점 간의 위치 관계를 기반으로 정렬하는 방식을 사용한다. 이는 동적 메쉬의 정점들이 국소적으로 유사한 움직임을 보이는 경향이 있기 때문이다. 특히, 두 정점 간의 거리가 가까울수록 움직임 벡터의 유사성이 높게 나타나는 경향이 있다. 이에 따라, 예측자 후보 리스트에서는 현재 정점과의 거리가 가까운 인접 정점일수록 리스트의 앞부분에 배치하여 예측자 선정 과정에서 우선적으로 고려되도록 한다. 이러한 위치 관계는 움직임 벡터가 적용되지 않은 참조 베이스 메쉬 상에서 측정된다. 참조 베이스 메쉬 상에서 현재 정점에 인접 정점들 간의 거리는 식 (3)과 같이 정의된다.

$$D_j = (p_i[x] - q_j[x])^2 + (p_i[y] - q_j[y])^2 + (p_i[z] - q_j[z])^2 \quad (3)$$

식 (3)에서 D_j 는 참조 베이스 메쉬 상의 현재 정점에 대응되는 정점 p_i 와 그 주변의 인접 정점 q_j 간의 거리(squared L2 norm)를 나타낸다. $p_i[x]$, $p_i[y]$, $p_i[z]$ 는 p_i 의 위치 좌표를 의미하며, $q_j[x]$, $q_j[y]$, $q_j[z]$ 는 p_i 의 인접 정점 q_j 의 위치 좌표를 나타낸다.

하지만, 거리 기반 정렬 방식은 편향된 방향에 있는 예측자들만 선택될 수 있는 한계가 있다. 이는 가까운 정점들이 특정 방향에 편중되어 있을 경우, 현재 정점을 기준으로 다양한 방향에 존재하는 움직임 벡터들을 통해 예측을 수행하는 것이 아니라 한 방향에 존재하는 움직임 벡터만을 사용하여 예측을 수행하게 되므로, 예측 정확도가 저하될 수 있다. 따라서 거리만을 기준으로 한 예측자 선정 방식은 예측 정확도 향상에 있어 제한적인 효과를 가지며, 방향성, 분포 균형 등 다양한 공간적 요소를 함께 고려하는 보다 정교한 접근이 필요하다. 이에 본 논문에서는 삼각형의 무게 중심을 활용한 예측자 후보 리스트 정렬 방법을 제안한

다. 세 개의 정점으로 구성된 삼각형의 무게 중심이 현재 정점에 가까울수록, 해당 삼각형을 이루는 정점들이 현재 정점을 기준으로 서로 다른 방향에 분포해 있을 가능성이 높다. 이러한 특성을 활용하면, 현재 정점을 중심으로 다양한 방향에 위치한 인접 정점들을 효과적으로 선택할 수 있으며, 결과적으로 예측자 후보 리스트의 공간적 다양성을 확보하여 예측 정확도 향상에 기여할 수 있다. 그러나 인접 정점들로부터 가능한 모든 세 정점 조합을 고려할 경우, 계산 복잡도가 급격히 증가할 뿐만 아니라, 모든 조합이 실제로 의미 있는 정렬 기준을 제공하지는 않는다는 한계가 존재한다. 특히 인접 정점의 수가 많아질수록 조합의 수가 기하급수적으로 증가하고, 단순 조합만으로는 후보 리스트 내 정렬의 명확성과 일관성을 보장하기 어렵다. 이에 효율성과 정렬 기준의 명확성을 동시에 확보하기 위해, 현재 정점과 거리가 가장 가까운 두 정점을 우선적으로 선택하고, 이 두 정점을 기준점으로 삼아 나머지 인접 정점들과 조합하여 삼각형을 구성하고, 해당 삼각형들의 무게 중심이 현재 정점에 가까운 순서대로 예측자 후보 리스트를 정렬한다. 이를 통해 연산량을 효율적으로 제한하면서도, 다양한

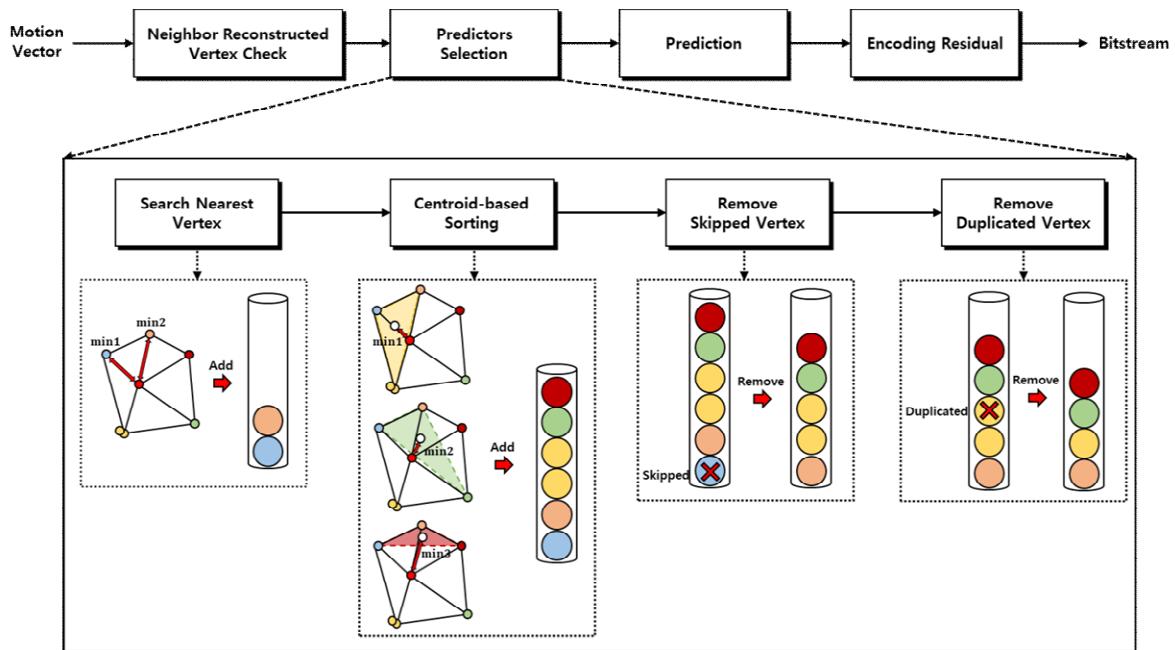


그림 7. 제안하는 방법의 움직임 벡터 예측 모드에 대한 부호화기 블록도

Fig. 7. Encoder Block Diagram for the Proposed Motion Vector Prediction Mode

방향성을 가진 예측자 구성이 가능하여 예측 성능의 향상을 기대할 수 있다. 세 정점(q_1, q_2, q_j)으로부터 유도되는 삼각형의 무게 중심의 좌표는 식 (4)와 같이 정의된다.

$$TC_j = \left(\frac{q_1[x] + q_2[x] + q_j[x]}{3}, \frac{q_1[y] + q_2[y] + q_j[y]}{3}, \frac{q_1[z] + q_2[z] + q_j[z]}{3} \right) \quad (4)$$

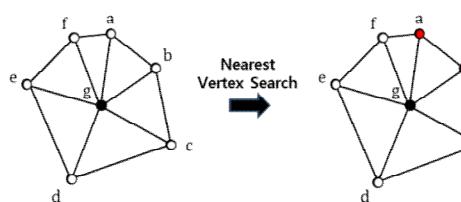
식 (4)에서 TC_j 는 두 기준점(q_1, q_2)과 추가로 선택된 인접 정점 q_j 로부터 형성된 삼각형의 무게 중심 좌표를 나타낸다. 각 무게 중심에 대한 현재 정점과의 거리(D_{TC_j})는 식 (5)와 같이 계산된다.

$$D_{TC_j} = (p_i - TC_j)^2 \quad (5)$$

이러한 방식은 복원된 인접 정점 수가 최대 예측자 개수를 초과할 경우에 적용된다. 예측자 후보 리스트의 첫 번째와 두 번째 후보는 기준점으로 인접 정점 중, 현재 정점과의 거리가 가까운 순서대로 선택되며, 이후 세 번째 후보부터는, 기준점과 나머지 후보 정점으로 구성된 삼각형의 무게 중심을 기준으로, 참조 베이스 메쉬 상에서 해당 무게 중심과 현재 정점의 위치 간의 거리가 가까운 순서대로 리스트에 추가된다. 그림 8의 예시를 통해 이 과정을 설명하면 다음과 같다. 복원된 인접 정점들 중에서 현재 정점과의 거리가 가까운 두 정점(a, b)을 기준점으로 선택하고, 이들을 예

측자 후보 리스트의 앞부분에 추가한다. 이후 기준점을 제외한 나머지 인접 정점들(c, d, e, f)은 기준점들과 조합하여 삼각형을 구성하고, 그 무게 중심과 현재 정점 위치 간의 거리를 계산한다. 이 거리 값을 기준으로 가까운 순서대로 정렬하며($D_{TC_d} < D_{TC_e} < D_{TC_c} < D_{TC_f}$), 이에 따라 예측자 후보 리스트에는 (d, e, c, f) 순서로 정점이 추가된다. 최종적으로 예측자 후보 리스트에는 (a, b, d, e, c, f) 순으로 예측자 후보들이 정렬된다.

예측자 후보 리스트로부터 현재 정점의 움직임 벡터를 예측하기 위한 예측자 결정 과정은, 후보 리스트의 앞에서부터 순차적으로 수행되며, 각 예측자 후보는 중복 여부 및 움직임 벡터의 부호화 모드에 따라 선택 여부가 결정된다. 예측자의 중복 여부 확인은, 이미 선택된 예측자와 동일한 위치에 동일한 움직임 벡터를 가지는지를 판단하는 과정이다. 만약 이러한 중복된 움직임 벡터가 예측자 후보에 포함될 경우, 예측값이 특정 정점으로 편향되어 예측 정확도가 저하될 수 있다. 따라서 예측자 결정 과정에서 이러한 중복 예측자는 후보 리스트에서 제외하여, 예측의 다양한 가능성을 확보하고 정확성을 향상시키도록 한다. 예측자의 부호화 모드 확인은, 해당 예측자가 스kip 모드를 통해 복원되었는지 여부를 판단하기 위한 과정이다. 스kip 모드로 복원된 움직임 벡터의 경우, 실제 움직임 벡터 값은 사용하지 않고 모든 값이 0으로 복원되므로, 이를 예측에 활용할 경



Nearest Vertex Search
Triangle Centroid Measurement

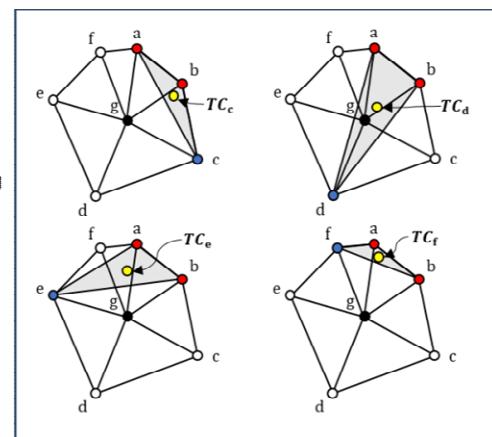


그림 8. 제안하는 방법의 예측자 후보 리스트 구성 예시

Fig. 8. Example of Predictor Candidate List Construction in the Proposed Method

우 예측 정확성이 저하될 수 있다. 따라서 예측자 후보 리스트를 구성할 때, 스킵 모드에 의해 복원된 움직임 벡터를 보유한 예측자 역시 후보에서 제외한다. 예측자 결정 과정은 움직임 벡터의 각 축(x, y, z)별로 결정되며, 결정된 예측자의 수가 최대 예측자 수에 도달하거나 후보 리스트를 모두 확인할 때까지 수행된다. 앞선 과정은 부호화된 모드를 기반으로 수행되므로 각 움직임 벡터의 부호화 모드를 저장하는 것이 필수적이다. 예측자 후보의 부호화 모드를 확인하기 위해 정점 단위로 부호화 모드를 개별 저장할 수도 있으나, 부호화 모드는 정점 단위가 아닌 움직임 그룹 단위로 정의되므로, 움직임 그룹 단위로 부호화 모드를 저장하고 이를 참조하는 방식을 적용한다. 즉, 예측자 후보가 속한 움직임 그룹을 식별한 뒤 해당 그룹에 할당된 부호화 모드를 참조함으로써 예측자 후보의 부호화 모드를 확인할 수 있다. 한편, 예측자의 중복 여부는 각 정점별로 전송되는 중복 플래그를 통해 판별할 수 있다.

IV. 실험 결과

본 논문에서 제안하는 방법의 성능을 평가하기 위하여, V-DMC 참조 소프트웨어 TMM v12.0^[15]에 제안하는 방법을 구현하여 기존 TMM v12.0과의 성능 비교를 수행한다. 제안하는 적응적 예측자 결정 방법을 V-DMC 참조 소프트웨어에 적용하기 위해, 기존 베이스 메쉬의 정점별 예측자 저장 방식과 예측자 결정 방식을 다음과 같이 개선하였다. 기존에는 각 정점별로 SPS에서 정의된 최대 예측자 개수만큼만 예측자들을 저장하였으나, 제안하는 방법에서는 움직임 벡터가 복원된 인접 정점들을 정점 간의 거리 및 삼각형 무게 중심 정보를 기준으로 정렬하여 모든 인접 정점을 후보로 저장하도록 수정하였다. 예측자 결정 방식에서는 기존 방식은 예측자를 별도의 검증 없이 바로 선택하는 반면, 제안하는 방법에서는 먼저 이미 선택된 예측자와의 중복 여부를 확인하고, 추가로 스킵 모드로 복원된 움직임 벡터인지 여부까지 검토하여 최종 예측자를 결정하도록 개선하였다. 추가적으로 움직임 그룹에 대한 부·복호화가 종료되면 예측 모드를 움직임 그룹 단위로 저장하도록 추가하였다.

제안하는 방법의 성능을 평가하기 위한 실험은 V-DMC CTC(Common Test Condition)^[16]의 LD(Low Delay) 조건에서 움직임 벡터 부호화가 수행되는 데이터셋을 대상으로 수행하였다. 추가적으로, 인코더 옵션으로 베이스 메쉬의 인터 프레임 매칭 과정을 적용한 결과도 분석하였다^[17]. 이 방법은 일대일 대응 관계가 성립하지 않았던 현재 베이스 메쉬와 참조 베이스 메쉬 사이에 최근접 이웃 매칭 기법을 활용해 새로운 일대일 대응 관계를 구축하여 인터 프레임(인터 모드) 부호화를 가능하게 한다. 이를 통해, 기존에는 인터 모드 부호화가 불가능했던 프레임에서도 움직임 벡터 부호화를 적용할 수 있는 프레임의 수가 늘어나며, 그 결과 전체 프레임 중 움직임 벡터 부호화가 차지하는 비율이 증가하는 효과가 나타난다. 움직임 벡터 부호화가 수행되는 데이터셋으로는 Category-A로 분류된 Soldier 시퀀스와 Category-C로 분류된 Levi, Mitch, Thomas 시퀀스들의 동적 메쉬 데이터가 사용되었으며, 각 동적 메쉬 데이터는 300프레임으로 구성된다. 본 논문에서는 각 동적 메쉬 데이터의 첫 32프레임을 실험에 사용하였다. 각 동적 메쉬 데이터에 대해 목표 비트율은 R1~R5 환경을 적용하여 실험을 진행하였다. 이때, 목표 비트율 R1은 3~4Mbps, R2는 4~8Mbps, R3는 9~12Mbps, R4는 14~17Mbps, R5는 21~25Mbps의 목표 비트율 환경을 의미한다. 실험 조건은 Intel(R) i7-10세대 10700 2.9GHz 프로세서, 64G RAM, Ubuntu 20.04 운영체제 환경에서 진행하였다. 성능 평가는 움직임 벡터 부호화로 인한 정점 위치 및 색상 변화가 원본 메쉬와 얼마나 차이가 나는지를 명확하게 평가하기 위해, V-DMC의 성능 평가는 포인트 클라우드 기반 지표와 2D 이미지 형태로 변환하여 측정하는 지표들을 함께 사용하였다. 포인트 클라우드 기반 주요 성능 평가 지표로는 D1 PSNR과 D2 PSNR이 있다. D1 PSNR은 두 정점 간의 오차를 측정하는 정점 대 정점(point-to-point) 방식이며, D2 PSNR은 대응하는 두 정점 간의 차이를 평면에 투영하여 오차를 측정하는 정점 대 평면(point-to-plane) 방식이다. Luma PSNR은 메쉬의 색상 정보 왜곡 정도를 평가한다. 이미지 기반 주요 성능 평가 지표로는 Geom이 있으며, 이는 메쉬 데이터를 2D 이미지 형태로 변환한 후 복원 결과와 원본 이미지 간의 기하학적 오차를 측정하는 방식이다. 또한, Image-based Luma PSNR은 투영법을 이용해 렌더링된

이미지를 간의 색상 왜곡 정도를 평가한다.

표 1은 예측자 후보 리스트 추가에 따른 BD-Rate 성능 차이를 비교하여 제시한다. 기준 방법에서는 예측자를 결정할 때, 순회 순서에 따라 예측자를 결정하며, 최대 예측자 개수에 도달하면 가장 마지막에 추가된 예측자를 새로운 예측자로 교체하는 방식을 사용하였다. 반면, 제안하는 방법에서는 참조 베이스 메쉬 상에서 현재 정점과 가장 가까운 두 점을 리스트에 추가하고, 이 두 점을 기준점으로 선정한 뒤, 나머지 인접 정점들에 대해서는 무게 중심이 현재 정점과 가까운 순서대로 리스트에 추가하는 방식을 적용하였다. 모든 실험에서는 정점별 최대 예측자 개수를 3개로 설정하였다. 표 1의 성능 비교 결과 Mitch 시퀀스를 제외한 모든 시퀀스에서는 D1, D2 PSNR BD-Rate이 모두 향상되었으며 D1, D2에서 평균 -0.07%, -0.08%의 성능 향상을 확인할 수 있다. 이는 단순히 마지막에 추가된 예측자를 교체하는 기존 방식과 달리, 참조 베이스 메쉬 상에서 현재 정점과 인접 정점 간의 거리와 다양한 방향성을 함께 고려하여 리스트를 구성하는 것이 각 정점에 대해 더욱 효과적

인 예측자 선정을 가능하게 하여 압축 성능 향상에 기여함을 확인할 수 있다.

표 2는 예측자를 결정하는데 있어서 조건 없이 예측자를 결정하던 기준 방법 대비 예측자 후보의 부호화 모드와 중복 여부를 고려하여 결정하는 방법의 성능을 나타낸다. 성능 비교 결과 모든 시퀀스에 대해 D1, D2 PSNR BD-Rate에서는 손실이 발생하지 않았으며 전체 평균으로 봤을 때, D1과 D2 PSNR BD-Rate에서 각각 -0.09%, -0.10%의 성능 향상을 보였으며, Luma PSNR BD-Rate에서 평균적으로 -0.07% 성능 향상이 확인되었다. 표 2의 실험 결과를 바탕으로, 예측자를 결정하는 과정에서 예측 오차가 발생할 수 있는 스kip 모드로 복원된 예측자 후보와 동일한 예측자 후보가 예측자로 중복 포함되는 경우를 제외함으로써, 움직임 벡터 부호화의 예측 성능이 향상되었음을 확인할 수 있다. 이러한 방식은 예측자를 결정하는데 있어서 오차가 존재하는 후보와 편향을 일으킬 수 있는 후보를 효과적으로 제거하여, 보다 신뢰성 높은 예측자를 선정하고 결과적으로 압축 효율을 개선하는데 기

표 1. TMM v12.0 대비 예측자 후보 리스트 구성 방법의 BD-Rate 성능

Table 1. BD-Rate performance of the predictor candidate list construction method compared to TMM v12.0

Condition	Sequence	Pointcloud-based BD Rate [%]					Image-based BD Rate [%]	
		D1	D2	Luma	Chroma Cb	Chroma Cr	Geom	Luma
LD	Soldier	-0.07	-0.11	0.12	-0.74	-1.58	0.01	0.15
	Levi	-0.14	-0.12	-0.06	-0.59	-0.16	0.27	0.16
	Mitch	0.06	0.03	0.17	-0.75	-1.98	-0.10	0.15
	Thomas	-0.12	-0.10	-0.28	1.27	0.23	0.17	0.06
	Overall average	-0.07	-0.08	-0.01	-0.20	-0.87	0.09	0.13
	Avg. Enc Time [%]				100			
	Avg. Dec Time [%]				100			

표 2. TMM v12.0 대비 스kip 모드 및 중복 정점 기반의 예측자 결정 방법의 BD-Rate 성능

Table 2. BD-Rate performance of the predictor selection method based on skip mode and duplicate vertices compared to TMM v12.0

Condition	Sequence	Pointcloud-based BD Rate [%]					Image-based BD Rate [%]	
		D1	D2	Luma	Chroma Cb	Chroma Cr	Geom	Luma
LD	Soldier	-0.04	-0.15	0.15	-1.21	-1.54	-0.07	0.21
	Levi	-0.10	-0.09	-0.01	-0.39	-0.05	0.39	0.00
	Mitch	-0.01	-0.03	0.09	-1.63	-0.41	0.46	0.12
	Thomas	-0.19	-0.15	-0.50	-0.95	0.02	-0.88	-0.16
	Overall average	-0.09	-0.10	-0.07	-1.04	-0.50	-0.03	0.04
	Avg. Enc Time [%]				100			
	Avg. Dec Time [%]				100			

여함을 확인할 수 있다.

표 3은 예측자 후보 리스트 구성 방식과 예측자 후보 리스트에서 예측자를 결정하는 방식 두 가지를 동시에 적용했을 때의 BD-Rate 성능 차이를 나타낸다. 해당 실험에서 제안하는 방법은 각 정점의 예측자 후보 리스트를 구성할 때, 현재 정점과 인접 정점 간의 위치 관계를 고려하여 후보 리스트를 생성하며, 리스트의 앞에서부터 예측자를 선택하는 과정에서 예측 오차가 존재하거나 편향된 결과를 유발할 수 있는 후보들은 제외하고, 최적의 예측자를 결정하여 움직임 벡터 예측을 수행하였다. 이러한 방식으로 실험을 진행한 결과, 모든 시퀀스에서 D1, D2 PSNR BD-Rate의 개선이 확인되었으며, D1, D2 PSNR BD-Rate에서는 평균 -0.09%, -0.12%, 포인트 클라우드 기반의 Luma PSNR BD-Rate에서는 평균 -0.15%의 감소를 통해 성능이 향상되었음을 확인할 수 있다.

표 4는 TMM v12.0의 베이스 메쉬의 인터 프레임 매칭 과정을 수행하는 조건에서, 기존 방법과 제안하는 방법 간의 성능 차이를 나타낸다. 성능 분석 결과, D1과 D2 PSNR

BD-Rate에서 각각 -0.28%, -0.23%의 성능 향상이 있었으며, 포인트 클라우드 기반 Luma PSNR BD-Rate에서는 평균 -0.26%, 이미지 기반 Luma PSNR BD-Rate에서는 평균 -0.08%의 성능 향상이 확인되었다.

실험 결과, 부호화 및 복호화의 시간 복잡도는 증가하지 않았다. 이는 움직임 벡터가 적용되는 베이스 메쉬의 정점 수가 복원 메쉬 대비 적기 때문이다. 예를 들어 베이스 메쉬에 세분화 과정을 3회 수행하는 경우, 베이스 메쉬의 정점 수는 전체 정점 수의 약 1/64 수준에 불과하여 추가 연산량이 크지 않다. 따라서 제안하는 방법에서 수행되는 거리 계산과 같은 연산이 전체 연산량에 미치는 영향은 미미하며, 결과적으로 시간 증가가 발생하지 않은 것으로 판단된다.

V. 결 론

본 논문에서는 동적 메쉬 압축을 위한 V-DMC 표준 기술

표 3. TMM v12.0 대비 무게 중심 정렬과 적응적 예측자 후보 결정 방법의 BD-Rate 성능

Table 3. BD-Rate performance of the centroid-based sorting and adaptive predictor candidate selection method compared to TMM v12.0

Condition	Sequence	Pointcloud-based BD Rate [%]					Image-based BD Rate [%]	
		D1	D2	Luma	Chroma Cb	Chroma Cr	Geom	Luma
LD	Soldier	-0.02	-0.14	0.05	1.18%	-0.26%	-0.18	0.16
	Levi	-0.11	-0.09	-0.44	-0.34	0.09	-0.29	-0.07
	Mitch	-0.11	-0.14	0.15	-0.59	0.05	0.41	0.14
	Thomas	-0.11	-0.10	-0.35	0.41	-1.37	0.02	0.02
	Overall average	-0.09	-0.12	-0.15	0.17	-0.37	-0.01	0.06
	Avg. Enc Time [%]				100			
	Avg. Dec Time [%]				100			

표 4. 베이스 메쉬의 인터 프레임 매칭 과정 수행 조건 하에 TMM v12.0 대비 제안하는 방법의 BD-Rate 성능

Table 4. BD-Rate performance of the proposed method under the base mesh inter-frame prediction optimization option compared to TMM v12.0

Condition	Sequence	Pointcloud-based BD Rate [%]					Image-based BD Rate [%]	
		D1	D2	Luma	Chroma Cb	Chroma Cr	Geom	Luma
LD	Soldier	-0.05	-0.09	-0.03	0.24	-0.11	-0.02	-0.01
	Levi	-0.11	-0.09	-0.44	-0.34	0.09	-0.29	-0.07
	Mitch	-0.47	-0.36	-0.09	1.30	0.85	-1.23	0.10
	Thomas	-0.50	-0.39	-0.47	-4.22	-1.42	-0.54	-0.34
	Overall average	-0.28	-0.23	-0.26	-0.75	-0.15	-0.52	-0.08
	Avg. Enc Time [%]				100			
	Avg. Dec Time [%]				100			

에서 베이스 메쉬의 움직임 벡터 예측 과정에 적응적 예측자 결정 방법을 제안하였다. 제안하는 방법은 현재 정점과 인접 정점 간의 위치 관계를 활용하여 예측자 후보 리스트를 구성하고, 스kip되지 않고 중복되지 않는 예측자를 선택하여 움직임 벡터 예측을 수행한다. 실험 결과, 제안하는 방법은 별도의 부·복호화 시간 증가 없이 Point cloud-based BD-Rate 측면에서 기존 방법(TMM v12.0) 대비 LD 환경에서 D1, D2 PSNR BD-Rate에서 -0.09%, -0.12%, Luma PSNR BD-Rate에서 -0.15%의 성능 향상을 보였으며, 베이스 메쉬의 인터 프레임 매칭 과정을 수행한 조건에서 D1, D2, Luma PSNR BD-Rate에서 -0.28%, -0.23%, -0.26%의 추가적인 성능 향상이 확인되었다. 이를 통해 제안하는 적응적 예측자 결정 방법이 동적 메쉬 압축 성능 향상에 효과적임을 검증하였다.

참 고 문 현 (References)

- [1] Y. Choi, J. -B. Jeong, S. Lee and E. -S. Ryu, "Overview of the Video-based Dynamic Mesh Coding (V-DMC) Standard Work," 2022 13th International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, Republic of, 2022, pp. 578-581.
doi: <https://doi.org/10.1109/ICTC55196.2022.9952734>
- [2] CfP for Dynamic Mesh Coding, ISO/IEC JTC1/SC29/WG7, N145, Oct. 2021.
- [3] Text of ISO/IEC FDIS 23090-29 Video-based dynamic mesh coding, ISO/IEC JTC1/SC29/WG7, N01271, Daejeon, Jul. 2025.
- [4] J. Rossignac, "Edgebreaker: connectivity compression for triangle meshes," in IEEE Transactions on Visualization and Computer Graphics, vol. 5, no. 1, pp. 47-61, Jan.-March 1999.
doi: <https://doi.org/10.1109/2945.764870>
- [5] Charles Teorell Loop, "Smooth Subdivision Surfaces Based on Triangles," M.S. Mathematics thesis, University of Utah, 1987.
- [6] M. Kim, J. Byeon, D. Sim, D. Kim, J. Suh, H. Park, S. Koh, N. Kwon, "[V-DMC][EE4.15-related][New] Normal-based subdivision method," ISO/IEC JTC1/SC29/WG7, m67581, Apr. 2024.
- [7] G. Sullivan, J. Ohm, W. Han, T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," IEEE Transactions on circuits and systems for video technology, Vol.22, No.12, pp.1649-1668, Dec. 2012.
doi: <https://doi.org/10.1109/TCSVT.2012.2221191>
- [8] M. Lee, H. Song, J. Park, B. Jeon, J. Kang, J. Kim, Y. Lee, J. Kang, D. Sim, "Overview of Versatile Video Coding (H. 266/VVC) and Its Coding Performance Analysis," IEIE Transactions on Smart Processing & Computing, Vol. 12, No.2, pp.122-154, Apr. 2023.
- [9] M. Kraus, "The pull-push algorithm revisited," Proceedings GRAPP 2: 3, 2009.
- [10] H. Huang, L. Hou, Q. Yang and Y. Xu, "Inter-Frame Coding for Dynamic Meshes via Coarse-to-Fine Anchor Mesh Generation," 2024 IEEE International Conference on Visual Communications and Image Processing (VCIP), Tokyo, Japan, 2024.
doi: <https://doi.org/10.48550/arXiv.2411.03921>
- [11] J. Xu, K. Kawamura, "[V-DMC][EE4.1 related][new] On improving mesh motion coding with skip mode of motion prediction," ISO/IEC JTC1/SC29/WG7, m60214, Jul. 2022.
- [12] Jianfeng Xu, Haruhisa Kato, Kei Kawamura, "Enhancement of Motion Field Coding by Integrating Duplicated Vertices in Dynamic Mesh Compression", ITE Trans. on Media Technology and Applications, Vol. 12, No. 4, 2024.
doi: <https://doi.org/10.3169/mta.12.248>
- [13] J. Tian, X. Xu, C. Huang, X. Zhang, and S. Liu, "On Motion Field Coding," ISO/IEC JTC1/SC29/WG7, m61211, Oct. 2022.
- [14] Toshiyasu Sugio, "[V-DMC][EE4.12] Report of EE4.12 motion field coding improvements," ISO/IEC JTC1/SC29/WG7, m65137, Oct. 2023.
- [15] V-DMC TMM12.0, ISO/IEC JTC1/SC29/WG7, N01176, May, 2025.
- [16] Common Test Condition for V3C and V-DMC, ISO/IEC JTC1/SC29/WG7, N00555, Mar, 2023.
- [17] W. Zou, J. Chen, W. Zhang, F. Yang, Z. Lv , "[V-DMC][new] Base mesh generation for inter-frame coding," ISO/IEC JTC1/SC29/WG7, m61046, Oct. 2022.

저 자 소 개

김 영 융



- 2024년 2월 : 광운대학교 소프트웨어학부 학사
- 2024년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 석사과정
- ORCID : <https://orcid.org/0009-0008-4201-3607>
- 주관심분야 : 3D 데이터 압축, 영상압축, 컴퓨터비전

변 주 형



- 2019년 2월 : 광운대학교 컴퓨터공학과 학사
- 2021년 2월 : 광운대학교 컴퓨터공학과 석사
- 2021년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 박사과정
- ORCID : <https://orcid.org/0000-0002-6165-9189>
- 주관심분야 : 3D 데이터 압축, 영상압축, 컴퓨터비전

심 동 규



- 1993년 2월 : 서강대학교 전자공학과 공학사
- 1995년 2월 : 서강대학교 전자공학과 공학석사
- 1999년 2월 : 서강대학교 전자공학과 공학박사
- 1999년 3월 ~ 2000년 8월 : 현대전자 선임연구원
- 2000년 9월 ~ 2002년 3월 : 바로비전 선임연구원
- 2002년 4월 ~ 2005년 2월 : University of Washington Senior research engineer
- 2005년 3월 ~ 현재 : 광운대학교 컴퓨터공학과 교수
- ORCID : <https://orcid.org/0000-0002-2794-9932>
- 주관심분야 : 영상신호처리, 영상압축, 컴퓨터비전