

ECM 화면간 예측 기술 I: MV 부호화와 블록 병합

□ 박내리, 남정학, 임재현 / LG전자

요약

Joint Video Experts Team(JVET)이 2020년 7월 VVC/H.266 표준을 완성한 이후, 차세대 표준의 필요성에 대한 논의가 지속되었고, 이에 따라 ECM(Enhanced Compression Model) 참조 소프트웨어가 개발되어 다양한 기술에 대한 탐색 실험이 진행되었다. 2027년 1월 차세대 표준에 대한 CfP(Call for Proposals) 응답을 진행하기로 합의됨에 따라 ECM 개발은 마무리 단계에 접어들었으며, 대규모 기술 기고는 소강 상태에 이를 것으로 예상되는 바, 현재(ECM-19.0)까지 포함된 화면 간 예측 기술 중 AMVP와 MERGE 모드에서 채택된 기술을 소개하고자 한다.

I. 서론

Joint Video Experts Team(JVET)은 ISO/IEC SC29 WG5와 ITU-T SG21/WP3(구 SG16/WP3)이 공동으로 비디오 코딩 표준을 개발하기 위해 설립되었으며, 2020년 7월 VVC/H.266[1]을 제정한 이후 VVC/H.266보다 향상된 압축 성능을 목표로 기술 탐색을 진행하고 있다. 2021년 발표된 JVET-U0100[2] 문서는 VVC/H.266 표준화 과정에서 채택되지 못한 기술 및 템플릿 매칭 기반의 움직임 보정 기술 등을 소개하며, 차세대 비디오 표준을 위한 압축 성능 향상 가능성을 제시하였다. JVET은 이를 시작으로 차세대 표준을 준비하기 위한 논의를 시작하였으며

다양한 기술 탐색 실험을 통해 채택된 기술들을 기반으로 ECM(Enhanced Compression Model) 참조 소프트웨어를 개발 및 관리하고 있다.

ECM의 성능을 토대로 2025년 7월 JVET은 압축률, 구현 용이성, 다양한 콘텐츠에 대한 적용 가능성 등의 측면에서 차세대 표준의 필요성을 검증하기 위해 CfE(Call for Evidence)를 발간하고, 2025년 10월 Fraunhofer HHI, Qualcomm 등 국내외 연구기관 및 기업에서 제출한 결과를 기반으로 CfE 응답에 대한 평가를 진행하였다.

평가 결과[3], 추가적인 개발 및 표준화 과정을 통해 VVC/H.266 표준의 압축 성능을 크게 상회할 가능성이 있다고 판단되었으며, 이에 따라, 공식적으로 CfP(Call for

Proposal)를 준비하고 2027년 1월 회의에서 Cfp 응답 절차를 진행하기로 합의하였다.

공식적인 차세대 비디오 코딩 표준화 일정이 확립됨에 따라 ECM 개발은 점차 마무리 단계에 접어들 것으로 예상되며, 본 고는 2025년 10월까지 채택된 ECM 기술(ECM-19.0) 중 화면 간 예측 기술, 특히 AMVP와 MERGE 모드에서 채택된 기술을 중심으로 소개하고자 한다.

II. ECM 화면 간 예측 기술

본 장에서는 ECM-19.0에 포함된 화면 간 예측 기술 중, AMVP-MERGE 모드, MERGE 모드에서의 MVP 후보 기술, CIIP, GPM 기술의 개념과 동작에 대해 소개한다.

1. AMVP-MERGE 모드

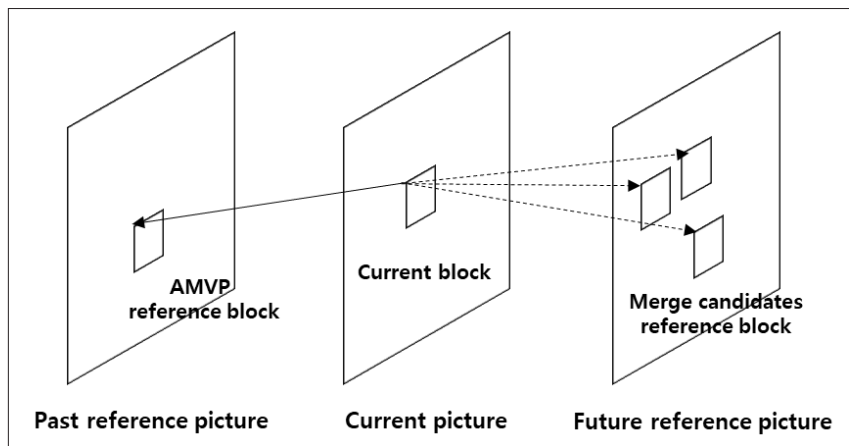
이전 비디오 코딩 표준인 HEVC와 VVC는 움직임 정보의 시그널링 방법을 크게 AMVP 모드와 MERGE 모드로서 구분하였다. AMVP 모드는 움직임 참조 방향을 비롯하여 각 방향의 MVP 인덱스 정보 및 MVD 정보 등을 시그

널링하는 반면, MERGE 모드는 MVP 인덱스 정보만을 시그널링하여 움직임 정보를 도출한다.

본 장에서 설명하는 AMVP-MERGE 모드는 이러한 AMVP 모드와 MERGE 모드를 혼합한 형태의 시그널링 방법을 포함하는 기술로, <그림 1>과 같이 양방향 예측에 있어 한쪽 방향은 AMVP 예측자, 반대 방향은 MERGE 예측자를 사용한다[4].

AMVP-MERGE 모드는 특정 조건을 만족할 때 사용여부가 시그널링된다. 본 모드는 참조 픽처 리스트(RPL, Reference Picture List) 내에 현재 픽처를 기준으로 시간적으로 과거 픽처와 미래 픽처가 모두 존재하는 경우(True-Bi)에 적용된다. 또한, 과거 참조 픽처만 존재하는 저지연 환경(Low-delay configuration)에 한해서는 True-bi 조건을 만족하지 않는 경우에도 AMVP-MERGE 모드를 허용하는 기술이 추후 포함되었다[5].

AMVP-MERGE 모드가 적용될 때, AMVP 예측자의 방향 정보(LX, $X=0, 1$)를 시그널링하며, 이에 따라 MERGE 예측자는 반대 방향($1 - LX$)으로 결정된다. AMVP 예측자를 생성하기 위한 움직임 정보(참조 픽처 인덱스, MVP 인덱스, MVD)가 추가로 시그널링되며, 이때 참조 픽처 재정렬 기술[6]이 적용된 경우 참조 픽처 인덱스의 시그널링



<그림 1> AMVP-MERGE 모드의 개념도

은 생략된다. MERGE 예측자를 위한 움직임 정보(MVP 인덱스)는 별도로 시그널링되지 않으며, 앞서 시그널링된 AMVP 예측자를 위한 MVP 인덱스로부터 추론된다[4][7]. 또한, AMVP 예측자를 위한 MVD 정보의 시그널링을 생략하는 방법이 채택되었으며, 구체적으로 MVP 인덱스가 2의 값을 가지는 경우에만 MVD 정보를 시그널링하고 그 외의 경우는 생략하도록 변경되었다[7].

AMVP-MERGE 모드의 디코딩 과정에서 AMVP 예측자를 위한 후보 리스트를 구성하고, MERGE 예측자는 MERGE 후보 중 AMVP 후보와 True-Bi 조건을 만족하는 후보로 구성된 뒤, 각 후보의 조합은 양방향 정합(Bilateral matching) 방법을 기반으로 재정렬된다. 이후 AMVP-MERGE 후보를 대상으로 움직임 보정 과정이 수행된다. 이때 현재 픽처와 각 참조 픽처 간 POC 간 거리가 동일할 때 양방향 매칭(Bilateral matching) 방법을 수행하고, 그렇지 않은 경우 템플릿 매칭(Template matching) 방법을 이용하여 움직임 보정을 수행한다.

이러한 AMVP-MERGE 모드의 구조는 IBC 모드의 양방향 예측 모드로 확장되어 적용되기도 하였다[8].

2. MERGE 모드의 MVP 후보 기술

HEVC 표준부터 도입된 MERGE 모드는 AMVP 모드에 비해 시그널링 정보가 적어 압축 성능 향상에 크게 기여해 왔다. 특히, MVP 후보의 종류, 개수 및 후보 리스트에 포함되는 순서 등이 성능 변화에 민감하게 작용하여 ECM 개발 과정에서도 다양한 후보 생성 기술이 제안되었다. VVC에서 MERGE 모드를 위한 MVP 후보는 공간적 후보(spatial candidate), 시간적 후보(temporal candidate), 히스토리 기반의 후보(history-based candidate), 쌍-평균 후보(pairwise-average candidate), 그리고 제로 벡터 후보(zero-vector candidate)로 구성된다. ECM에서는 템플릿 매칭 기반의 MVP 후보를 대상으로 한 재정렬 방식[35]이 도입되면서 시그널링에 대한 부담은 줄이면서도 보다 많은 수의 후보를 사용하는 것이 가능해짐에 따라 기

존의 후보들을 개선함과 동시에 새로운 형태의 후보를 추가하는 형태로 발전하였다. MERGE 후보 중 VVC 대비 변경사항이 미미한 히스토리 기반의 후보와 제로 벡터 후보에 대한 설명은 생략하였다.

1) 공간적 후보

공간적 인접 후보는 VVC에서의 탐색 위치를 유지하는 동시에, 현재 코딩 블록의 중심을 기준으로 좌측 및 상측에 인접한 블록을 추가로 탐색하도록 확장되었다[9]. 특히, 저지연 여부 및 픽처 크기에 따라 탐색 블록의 위치를 다르게 설정함으로써 다양한 위치의 후보가 고려될 수 있도록 하였다. 또한, 공간적 움직임 유사도를 이용하기 위한 후보 생성 방법은 인접 블록이 아닌 비인접(non-adjacent) 블록으로 확장되었다[2]. 비인접 후보의 탐색 위치는 현재 블록의 가로·세로 크기와 사전에 정의된 거리 인덱스(Distance index, $D = 1..4$)에 따라 결정되도록 설계되었고, 이후 거리 인덱스가 최대 7까지 확장되고 후보의 개수가 증가하면서 탐색 순서를 두 단계로 구분하여 우선순위에 따라 후보를 탐색할 수 있도록 변경되었다[36].

2) 시간적 후보

시간적 후보는 VVC 대비 큰 변화를 보였다. 우선, 대응 블록(co-located block)으로 고려될 수 있는 대상 후보가 크게 증가하였다. VVC에서는 현재 블록 위치를 기준으로 우하단 및 중앙 위치를 대응 블록으로 정의하였으나, ECM에서는 현재 블록 위치를 기준으로 블록의 가로·세로 크기와 거리 인덱스에 따라 유도된 우측, 하측, 우하단의 특정 위치들을 추가[36]로 탐색할 수 있도록 하여 비인접 후보를 포함하였으며, 현재 블록 위치의 가장자리를 따라 블록 내·외부의 특정 인접 위치들도 포함하였다[9]. 이러한 대응 블록 후보들은 쌍(pair)으로 구성되며[36], 한 쌍의 후보 중 첫 번째 후보 위치의 LX($X=0, 1$) 방향 움직임 벡터가 유효하지 않을 경우 두 번째 후보의 동일 방향의 움직임 벡터를 우선 활용할 수 있도록 하거나, 첫 번째 후보 위치의 LX 방향의 움직임 벡터가 유효하지 않을 때 첫

번째 후보의 (1-LX) 방향의 움직임 벡터를 우선 활용할 수 있도록 하여[37] 탐색 순서의 다양성을 높였다.

대응 블록 결정 후 후보의 참조 픽처 인덱스가 유도된다. VVC에서는 참조 픽처 인덱스가 '0'으로 고정되어 있었으나, ECM에서는 대응 블록의 참조 픽처와 동일한 픽처를 가리키는 참조 픽처 인덱스로 유도하도록 제안되었다[10]. 해당 참조 픽처가 존재하지 않는 경우에는 대응 픽처(co-located picture)와 대응 블록의 참조 픽처(co-located reference picture) 사이의 픽처 중 현재 픽처와 가장 먼 참조 픽처를 가리키는 참조 픽처 인덱스를 사용하거나[10], 대응 픽처와 그 참조 픽처 간의 거리 및 현재 픽처와 현재 참조 픽처 간 거리 비율이 1에 가까운 참조 픽처를 가리키는 인덱스를 사용할 수 있다[36]. 이후, 대응 블록의 움직임 벡터는 현재 픽처와 결정된 현재 참조 픽처 간의 POC 차이, 그리고 대응 픽처와 그 참조 픽처 간의 POC 차이를 고려하여 스케일링된다.

앞서 기술된 대응 블록 후보 외에도 TMVP 후보의 수가 충분하지 않은 경우, TMVP 리스트 내에 포함된 후보들 중 참조 픽처 인덱스가 '0'보다 큰 값을 갖는 후보를 대상으로 참조 픽처 인덱스를 '0'으로 고정하고 움직임 벡터를 스케일링한 변형 후보를 추가하는 방법이 포함되었다[11]. 또한, TMVP 리스트 내에 포함된 후보를 기반으로 CMVP(Chained Motion Vector Predictor) 후보를 유도하여 추가하는 방법도 포함되었다[11].

TMVP 후보에 대한 템플릿 매칭 기반의 움직임 보정 기술이 포함되었다. 이때 대응 블록의 움직임 벡터는 현재 참조 픽처에 맞게 스케일링되며, 이를 시작점으로 하여 현재 참조 픽처 내 템플릿 블록과 대응 픽처 내 대응 블록 템플릿 간의 SATD(Sum of Absolute Transformed Differences) 비용이 최소가 되는 위치로 보정된다[12].

또한, Bi-TMVP 후보가 제안되었다[10]. 이는 현재 참조 픽처 내 한 블록과 그 블록의 참조 블록을 가리키는 움직임 벡터를 현재 픽처 기준의 움직임 벡터 쌍으로 구성함으로써 양방향의 움직임 벡터로 확장한다. Bi-TMVP 후보는 앞서 기술된 TMVP 후보들과 달리 HMVP 후보 이후에

MERGE 후보 리스트에 포함된다.

이와 같이 시간적 후보의 종류 및 디코딩 과정이 대폭 변경되었으며, 그 결과 RA(Random-Access)뿐만 아니라 LD(Low-Delay) 실험 환경에서의 성능 향상이 이루어졌다.

3) 쌍-평균 후보

쌍-평균 후보는 MERGE 후보 리스트에 포함된 첫 번째 후보와 두 번째 후보의 각 방향별 움직임 벡터를 평균하여 생성되는 후보로, ECM에서는 VVC의 동작에 더해 최대 6개의 후보를 추가할 수 있도록 하였다[9]. 이때 추가된 쌍-평균 후보에 대해서는 독립적인 템플릿 매칭 기반 재정렬 방법이 적용될 수 있다. 또한, 다수의 평균(Multiple-average) 후보[9]가 포함되었다. 쌍-평균 후보가 두 개의 후보를 평균하여 생성되는 것과 달리, 다수의 평균 후보는 네 개의 후보를 대상으로 각 예측 방향별 참조 픽처 인덱스가 동일한 경우 각 후보의 움직임 벡터를 평균하여 생성된다.

4) 추가 후보

기존의 MERGE 후보와는 구분되는 새로운 후보가 제안되었다. 아핀 모드(Affine mode)에서의 히스토리 기반 MVP 후보로 채택된 HAMI 후보[13]는 아핀 모드뿐만 아니라 기본 MERGE 모드(Regular merge mode)에도 변형 적용되었다. 히스토리 버퍼에 저장된 아핀 파라미터와 현재 블록에 인접한 블록의 움직임 정보 및 위치 정보를 기반으로, 현재 코딩 블록 내 중앙 위치의 움직임 벡터를 유도하여 코딩 블록 단위의 움직임 벡터를 갖도록 한다. 해당 후보는 쌍-평균 후보 이후, 하나의 제로 벡터 후보 이후에 포함될 수 있다.

또한, CMVP(Chained Motion Vector Predictor)가 새로운 후보로서 포함되었다[14]. 사전에 정의된 현재 블록 내 특정 위치(basePos)를 기준으로 앞서 MERGE 후보 리스트에 포함된 후보의 움직임 벡터를 이용하여 참조 블록을 찾고, 해당 참조 블록의 움직임 벡터를 최종 움직임 벡터로 정의한다. 이때 MERGE 후보 리스트에 포함된 후보

중 TMVP 후보는 CMVP를 위한 후보에서 제외되었다. 이후 현재 블록 내 특정 위치뿐만 아니라 인접 블록의 위치를 basePos에 포함하는 기술이 제안되었으며[15], CMVP 후보를 유도하기 위해 참조되었던 MVP 후보 리스트 내의 후보 외, 이후에 추가된 후보를 대상으로 추가적인 CMVP 후보를 구성하는 방법이 채택되기도 하였다[16].

또 다른 후보로, 생성된 MERGE 후보(Generated merge candidate)가 포함되었다[17]. 앞서 설명한 추가적인 CMVP 후보를 포함하여 MVP 후보 리스트에 포함된 후보를 대상으로 각 예측 방향별로 움직임 정보를 분할하고 템플릿 매칭 기반 재정렬을 수행한 뒤 각 예측 방향의 움직임 정보를 재조합함으로써 새로운 후보를 생성할 수 있도록 하였다.

3. CIIP(Combined Inter/Intra Prediction)

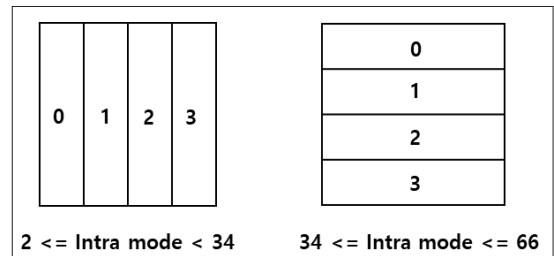
CIIP 모드는 MERGE 모드의 하위 기술로서 화면 간 예측 블록과 화면 내 예측 블록을 결합한 기술이다. 전통적으로 예측 모드는 화면 간 예측과 같이 참조 픽처로부터 획득된 예측 블록, 혹은 화면 내 예측과 같이 현재 픽처 내 현재 블록에 인접한 샘플을 이용하여 유도된 예측 블록으로 구분되나, CIIP는 두 모드를 이용하여 유도된 블록을 조합함으로써 새로운 형태의 예측 블록을 생성할 수 있도록 하였다. VVC에서는 화면 간 예측 블록(*predInter*)을 유도하기 위해 MERGE 후보 중 하나의 움직임 정보를 이용하며, 화면 내 예측 블록(*predIntra*)을 유도하기 위해 화면 내 예측 모드 중 하나인 PLANAR 모드를 사용하였다. 또한 화면 간 예측 블록과 화면 내 예측 블록을 가중합하여 최종 예측 블록(*predComb*)을 생성하였는데, 이때 가중치는 현재 블록에 인접한 좌측 블록(A) 및 상측 블록(B)의 화면 내 모드 여부에 따라 결정된다. A, B가 모두 화면 내 블록이면 $w = 3$, A 또는 B 중 하나만 화면 내 블록이면 $w = 2$, 모두

화면 간 블록이면 $w = 1$ 로 설정되며, 최종 블록은 블록 내 (x, y) 좌표에 대해 수식 (1)과 같이 유도된다.

ECM에서 CIIP 모드의 화면 간 예측 부분은 다른 MERGE 모드들과 결합하여 CIIP-TM[18], CIIP-AFFINE[19] 모드로 분기되었다. 이에 따라 기존의 화면 간 예측 블록을 생성하는 모드, 템플릿 매칭 기반의 움직임 보정을 수행하는 모드, 아핀 모델을 이용하여 서브블록 단위의 움직임 벡터를 유도하는 모드로 구분된다.

CIIP 모드의 화면 내 예측 부분은 PDPC(Position-Dependent Prediction Combination)의 적용 여부를 나타내는 플래그가 추가로 시그널링되며[2], 화면 내 예측 모드 또한 조건에 따라 PLANAR 혹은 TIMD(Template-based Intra Mode Derivation) 모드를 이용하여 유도할 수 있도록 변경되었다[18].

화면 간 예측 블록과 화면 내 예측 블록 간 가중합을 적용하는 단계에서는 블록 내 샘플의 위치에 따라 다른 가중치가 적용되도록 설계되었으며[18], 조건에 따른 블록의 분할 방법을 <그림 2>에 나타내었다. 그림과 같이 화



<그림 2> 화면 내 예측 모드에 따른 블록 분할 방법

<표 1> 영역별 가중치

Index	(wIntra, wInter)
0	(6, 2)
1	(5, 3)
2	(3, 5)
3	(2, 6)

$$PredComb[x][y] = (w * predIntra[x][y] + (4 - w) * predInter[x][y] + 2) >> 2 \quad (1)$$

$$PredComb[x][y] = (wIntra * predIntra[x][y] + wInter * predInter[x][y] + 4) >> 3 \quad (2)$$

면 내 예측 모드가 2 이상 34 미만일 때 블록은 세로로 분할되며, 34 이상 66 이하일 때 가로로 분할된다. 각 분할된 영역을 인덱스(Index) 0-3으로 표시하였으며 각 영역별 가중치는 <표 1>에 나타내었다. 화면 내 예측 블록을 위한 가중치를 $wIntra$ 로, 화면 간 예측 블록을 위한 가중치를 $wInter$ 로 정의할 때 최종 블록은 수식 (2)와 같이 유도된다.

다만, 화면 내 예측 모드가 PLANAR 혹은 DC 모드인 경우이거나 블록의 크기가 작은 경우 VVC와 동일한 방법으로 인접 블록의 화면 내 모드 여부에 따라 가중치가 결정된다.

4. GPM(Geometric Partitioning Mode)

GPM 모드는 MERGE 모드의 또 다른 하위 기술로서 현재 코딩 블록을 두 개의 기하학 영역으로 분할하여 각 영역별 예측을 수행한다. 사전에 정의된 분할 각도와 블록 중심으로부터의 거리를 하나의 인덱스로 정의 및 시그널링하며, 분할 경계 영역에서의 블렌딩 프로세스를 통해 두 개의 예측 블록 간 불연속성을 제거한 기술로 VVC에 채택되었다.

ECM은 GPM 모드와 기존 VVC에 존재하는 다른 모드 및 ECM에 새롭게 채택된 모드와의 조합을 통해 각 분할 영역에서의 자유도를 높였다. 또한, VVC 표준화 과정에서 복잡도 이슈로 포함되지 못한 툴들을 포함하기도 하였다. 템플릿 매칭 기반의 기술이 GPM에도 적용되어 GPM 분할 정보의 재정렬 및 각 분할 영역에서의 움직임 정보를 포함한 통합-재정렬 모드 등이 포함되었으며, 블렌딩 과정에서의 변화가 포함되었다.

1) GPM 모드와 기존 모드와의 조합

ECM에서는 GPM 모드의 각 예측 블록을 위한 움직임 예측 방법의 다양성을 보장하기 위해 기존 예측 모드와의 조합을 시도하였다. 우선, GPM 모드의 각 분할 영역은 MMVD(Merge with Motion Vector Difference) 모드를 가질 수 있으며, 이를 통해 움직임 벡터의 정확도를 높일 수 있도록 제안되었다[20]. GPM 모드의 각 예측 블록은 MMVD 모드(GPM-MMVD) 혹은 기존 모드(GPM-REGULAR)로 판단될 수 있으며, GPM-MMVD 모드의 경우 VVC의 MMVD 모드와 유사하게 움직임 벡터 보정치를 방향과 거리로 정의하여 시그널링한다. 또한, 각 예측 블록은 TM 모드(GPM-TM)로 판단될 수 있으며, 해당 모드에서는 템플릿 매칭 기반의 움직임 벡터 보정 기법이 적용된다[20]. GPM-MMVD 모드는 각 예측 블록 단위로 시그널링되는 반면, GPM-TM 모드는 코딩 블록 단위로 결정되며, 두 개의 예측 블록이 모두 GPM-MMVD가 아닐 때 GPM-TM 플래그가 시그널링될 수 있다.

기존 모드와의 조합은 MERGE 모드에 한정하지 않고 화면 내 예측 모드로 확대되었다[21]. GPM with Inter and Inter(GPM-INTRA) 기술이 채택되어, GPM 분할 영역별로 화면 간 예측 또는 화면 내 예측을 통해 생성될 수 있도록 하였다. 단, 두 개의 분할 영역이 모두 화면 내 예측으로 선택되는 경우는 제외된다. 분할 영역이 화면 내 예측 모드로 선택될 경우 해당 영역의 화면 내 예측 모드는 분할 각도와 평행한 모드(parallel mode), DIMD, TIMD, 인접 블록의 모드 순으로 구성된 MPM 리스트에서 결정된다. 또한, 블렌딩되는 분할 경계 영역에서의 움직임 벡터 및 화면 내 예측 모드의 저장 방법이 변경 적용되었다.

아울러, GPM 모드와 기존 모드의 조합은 서브블록 단위 예측 모드로 확장되어 각 GPM 분할 영역이 아핀 모드(GPM-AFFINE)로 결정될 수 있도록 제안되었다[19]. 각 분할 영역별 아핀 모드의 적용 여부를 판단하는 플래그가 시그널링되며, 아핀 모드의 MVP 후보 구성 방법을 따

르되, 단방향 예측 후보로 구성하고 SbTMVP(Subblock-based Temporal Motion Vector Prediction)는 제외한 후보 리스트를 구성하도록 하였다. 이후 GPM-AFFINE 및 GPM-REGULAR를 대상으로 템플릿 매칭 기반 움직임 보정을 수행하는 기술(GPM-AFFINE-TM, GPM-REGULAR-TM)이 제안되었으며[22], IBC 모드와 GPM-REGULAR, GPM-MMVD, GPM-AFFINE과의 조합도 채택[38]됨으로써 각 분할 영역에서의 모드 자유도가 향상되었다.

2) GPM의 기반 기능 확장

VVC에 포함된 GPM 모드는 각 분할 영역에서 단방향 예측만 수행하도록 제한되었다. 이는 하나의 코딩 블록이 최대 네 번의 움직임 보상 과정을 거치게 됨에 따라 발생할 수 있는 메모리 대역폭 문제를 고려하여 결정되었다. 한편, ECM은 복잡도나 메모리 증가 등 구현상의 이슈보다는 압축 성능 향상에 중점을 두어, 향후 표준 기술의 필요성을 평가하기 위한 단계로서 과거에 제안되었던 기술들이 다시 채택되기도 하였다.

앞서 기술한 바와 같이, 단방향 예측만 수행하던 GPM 모드에 대해 양방향 예측이 허용되었다[23]. 다만, 블록 크기가 작은 경우에는 단방향 예측을 유지하고, 그 외의 경우에는 양방향 예측을 허용하도록 하였으며, 이러한 양방향 예측 확장은 GPM-MMVD 및 GPM-TM 모드에도 적용되었다.

또한, VVC 표준화 과정에서 GPM 모드는 20개의 분할 각도와 블록 중심으로부터의 거리를 고려하여 총 64개의 분할 모드 형태로 정의되었으나, 이는 GPM 분할 모드별 성능과 복잡도 간의 trade-off, 그리고 분할 모드 수 증가에 따른 구현 및 검증 비용 등을 종합적으로 고려한 결과이다. ECM에서는 이러한 제약을 완화하여 GPM 분할 모드를 일부 확장하였다. 기존 20개의 분할 각도는 32개로 확장되었으며, 분할각의 가로·세로 비율을 1:8까지 허용하여 각도를 보다 세분화하였다. 또한, 코딩 블록의 가로세로 비율에 따라 기존 분할 각도를 조정하기도 하였다[24].

3) GPM 모드를 위한 템플릿 매칭 기반 재정렬 방법

ECM에 템플릿 매칭을 이용한 기술이 적용되면서 GPM 모드에도 이를 활용한 다양한 기법이 채택되었다. 우선, 64개의 GPM 분할 형태에 대해 템플릿 매칭 기반 재정렬을 수행한 후 32개의 후보를 선택하고, 이를 인덱스 시그널링하는 방법이 제안되었다[25]. 또한, 기존 Fixed-length coding 방식은 Golomb-Rice coding 방식으로 대체되었다[25].

GPM 모드는 블록 분할 모드와 각 분할 영역별 움직임 정보(MVP 인덱스)를 각각 시그널링하지만, 이를 하나의 인덱스로 통합하여 시그널링하는 통합-재정렬(Joint-reordering) 방법이 제안되었다[26]. 이 방법의 적용 여부를 판단하는 플래그가 시그널링되며, 해당 모드가 선택되었을 때 분할 모드와 각 분할 영역에서의 움직임 정보를 포함하는 후보 리스트가 구성된다. 이 리스트는 템플릿 매칭 기반 재정렬된 후 인덱스를 시그널링한다. 이러한 통합-재정렬 방법은 GPM-REGULAR와 GPM-TM을 포함하며, GPM-MMVD는 성능-복잡도 trade-off를 고려하여 제외되었다.

이후, 통합-재정렬 방법은 GPM-AFFINE 모드와 GPM-INTRA 모드를 포함하도록 확장되었다[27]. GPM-AFFINE 모드가 선택된 경우, 하나의 블록 분할 모드와 각 분할 영역의 움직임 정보 및 모드 지시자가 하나의 후보 리스트로 구성되며, 해당 리스트는 템플릿 매칭 기반 재정렬 후 인덱스 시그널링한다[28][27]. 여기서 모드 지시자는 각 분할 영역이 아핀 모드를 사용하는지를 나타내며, 아핀 모드일 때 아핀 모드의 MVP 후보 리스트 구성 방법을 따르고 서브블록 단위의 움직임 보상을 수행한다. 이 기술은 후보 리스트의 크기를 줄이고 블록의 크기를 제한하거나 인접/비인접 블록에서의 아핀 모드의 적용 여부를 고려함으로써 복잡도를 줄이기도 하였다.

통합-재정렬 기술이 적용될 때, GPM-AFFINE 모드가 아닌 경우 GPM-INTRA 모드 여부가 판단될 수 있다. GPM-INTRA 모드가 선택될 때, 하나의 블록 분할 모드와 각 분할 영역에서의 움직임 정보 및 INTRA 예측 모드

$$W0(x,y) = a.x + b.y + c \quad \text{and} \quad W1(x,y) = 1 - W0(x,y) \quad (3)$$

가 하나의 후보 리스트로 구성된다[29][27]. 이 리스트 역시 템플릿 매칭 기반 재정렬 후 인덱스 시그널링한다. 이때 화면 내 예측 모드는 MPM 후보 리스트와 동일한 방식으로 적용된다.

4) 적응적 블렌딩 방법

GPM 모드의 분할 영역 간 경계에서 수행되는 블렌딩(blending) 과정은 GPM 모드의 압축 성능 향상에 중요한 역할을 한다. 이러한 특성을 반영하여 GPM 모드를 위한 적응적 블렌딩 기술이 제안되었다[30]. 기존 VVC에서는 코딩 블록의 크기에 따라 사전에 정해진 영역에 대해 블렌딩을 수행하였으나, 제안된 기술에서는 코딩 블록 단위로 적응적 블렌딩의 적용 여부와 블렌딩 영역 크기를 나타내는 인덱스를 시그널링한다. 기존의 블렌딩 영역 크기를 τ 라고 할 때 ($\tau/4$, $\tau/2$, 2τ , 4τ)의 네 가지 블렌딩 영역이 추가되었다. 이후 블록의 크기에 따라 허용하는 블렌딩 영역의 범위를 달리하는 기술이 채택되었다[31]. 구체적으로 블록의 가로 혹은 세로 크기가 32보다 작은 경우 블렌딩 영역의 후보는 ($\tau/4$, $\tau/2$, τ , 2τ , 4τ)로 정의되며, 조건을 만족하지 않는 경우 ($\tau/2$, τ , 2τ , 4τ , 8τ)의 후보를 갖도록 변경되었다.

5) 회기 기반 블렌딩 모드

GPM 모드를 위한 암시적(implicit) 모드가 포함되었다[32]. 이는 템플릿 기반으로 블렌딩 가중치를 추정하는 방법을 사용한다. 템플릿 영역의 참조 샘플을 이용하여 선형 회기 기반 근사 방식으로 가중치를 추정하고 이를 통해 아핀 모델의 파라미터 (a , b , c)를 도출한다. 이후, 현재 코딩 블록 내 샘플의 위치 (x , y)에 따른 가중치, $W0$, $W1$

를 수식 (3)과 같이 계산하여 최종 예측 블록을 생성한다.

또한, GPM 암시적 모드는 코딩 블록 단위로 플래그를 시그널링한다. 해당 모드가 선택되었을 때 각 분할 영역에서의 움직임 정보(MVP 인덱스)를 포함하는 하나의 통합된 후보 리스트가 구성되며, 템플릿 매칭 기반 재정렬된 후 인덱스를 시그널링한다.

초기 GPM 암시적 모드는 GPM-REGULAR 모드만을 허용하였으나, 이후 GPM-TM 모드[33]와 GPM-INTRA 모드[34]도 포함하도록 확장되었다.

III. 결론

본 고에서는 최신 ECM(ECM-19.0)에 포함된 화면 간 예측 기술 중 AMVP-MERGE 모드, MERGE 모드에서의 MVP 후보 기술, CIIP 기술 및 GPM 기술의 개념과 동작을 소개하였다. 차세대 표준을 위한 사전 탐색 단계에서, 본 고에서 다룬 기술을 포함한 화면 간 예측 기술뿐만 아니라, 화면 내 예측, 변환 및 양자화, 인-루프 필터에서의 다양한 기술 기고와 시도를 통해 VVC 대비 압축 성능이 크게 향상되었다. 이를 바탕으로 차세대 비디오 압축 표준의 필요성에 대해 합의가 도출되었으며, 구체적인 표준 시작 시점이 결정되었다. 향후 본 표준화 과정에서 국내외 여러 기관이 참여하여 ECM에 포함된 기술을 기반으로 활발한 표준화 활동이 진행될 것으로 예상된다. 다만, 차세대 비디오 코딩 표준의 성공적인 배포를 위해서는 복잡도가 높거나 하드웨어 비용 증가 및 구현상의 이슈를 포함하는 기술들에 대해서는 신중한 접근이 필요할 것으로 판단된다.

참 고 문 헌

- [1] Versatile Video Coding (VVC), ITU-T Recommendation H.266 and ISO/IEC 23090-3, July 2020.
- [2] Y. J. Chang et al., "Compression efficiency methods beyond VVC", Jan. 2021, JVET-U0100.
- [3] F. Bossen et al., "Results of the Joint Call for Evidence on video compression with capability beyond VVC", Oct. 2025, JVET-AN2029.
- [4] Z. Zhang et al., "EE2: Bilateral and template matching AMVP-merge mode (test3.3)", Oct. 2021, JVET-X0083.
- [5] H. Jang et al., "EE2-2.1: AmvpMerge for low delay", Oct. 2022, JVET-AB0078.
- [6] H. Huang et al., "EE2-2.1: On the extended number of active reference pictures and block level reference picture reordering", Apr. 2022, JVET-Z0054.
- [7] Z. Zhang et al., "Non-EE2: MVD and merge index signaling of AMVP-merge mode", Jan. 2022, JVET-Y0129.
- [8] Y. Kidani et al., "EE2-2.2/EE2-2.3/EE2-2.4: Bi-predictive IBC GPM, bi-predictive IBC, and IBC MBVD list derivation for camera captured and screen contents", July 2023, JVET-AE0169.
- [9] N. Zhang et al., "EE2-3.3: Additional inter merge candidates", Apr. 2025, JVET-AL0157.
- [10] Z. Zhang et al., "EE2-3.4: MVP extension", July 2024, JVET-AI0183.
- [11] P.-H. Lin et al., "EE2-3.7: TMVP for chained motion vector prediction", July 2024, JVET-AI0187.
- [12] Z. Zhang et al., "EE2-3.6: MV refinement for TMVP", Apr. 2025, JVET-AL0214.
- [13] W. Chen et al., "EE2-2.7, 2.8, 2.9: History-parameter-based affine model inheritance and non-adjacent spatial neighbors for affine merge mode", Apr. 2022, JVET-Z0139.
- [14] Y. Kidani et al., "EE2-3.1: Chained motion vector prediction", Apr. 2024, JVET-AH0069.
- [15] D. Bugdayci Sansli et al., "EE2-2.1a/b: Chained candidates in AMVP and merge lists", June 2025, JVET-AM0106.
- [16] N. Zhang et al., "EE2-3.5: Additional chained motion vector prediction candidates", July 2024, JVET-AI0103.
- [17] D. Bugdayci Sansli et al., "EE2-3.1: Generated Merge Candidates", Oct. 2025, JVET-AN0236.
- [18] Z. Deng et al., "EE2-3.1-related: CIIP with template matching", Oct. 2021, JVET-X0141.
- [19] L. Zhao et al., "EE2-2.11d/e/f: Combination of Test 2.9, Test 2.10 and Test 2.11a", Jan. 2024, JVET-AG0142.
- [20] X. Xiu et al., "EE2-related: Combination of EE2-3.3, EE2-3.4 and EE2-3.5", July 2021, JVET-W0097.
- [21] Y. Kidani et al., "EE2-3.1: GPM with inter and intra prediction (JVET-X0166)", Jan. 2022, JVET-Y0065.
- [22] Y. Wang et al., "EE2 Test 3.12: Combined test of Test 3.6 and Test 3.11", Nov. 2024, JVET-AJ0274.
- [23] R. Yu et al., "EE2-3.2: Bi-predictive GPM", July 2023, JVET-AE0046.
- [24] Y. Ahn et al., "EE2-3.4: GPM mode extension", Nov. 2024, JVET-AJ0107.
- [25] C.-C. Chen et al., "EE2-2.4: Template matching based reordering for GPM split modes", July 2022, JVET-Z0056.
- [26] C. Ma et al., "EE2-3.4: Joint reordering of GPM split modes and partition indices", Apr. 2025, JVET-AL0134.
- [27] L. Zhang et al., "EE2-3.4: Combination of EE2-3.2 and EE2-3.3", Oct. 2025, JVET-AN0093.
- [28] L. Zhang et al., "EE2-3.2: Joint reordering of GPM with affine prediction", Oct. 2025, JVET-AN0091.
- [29] Z. Sun et al., "EE2-3.3: Joint reordering of GPM with intra prediction", Oct. 2025, JVET-AN0092.
- [30] Y. Kidani et al., "EE2-2.7: GPM adaptive blending (JVET-Z0059, JVET-Z0137)", July 2022, JVET-AA0058.
- [31] L. Zhao et al., "EE2-3.4: Adaptive GPM blending", Apr. 2024, JVET-AH0073.
- [32] P. Bordes et al., "EE2-2.11: Regression-based GPM blending (tests a,b,c)", Jan. 2024, JVET-AG0112.
- [33] K. Jia et al., "Non-EE2: Regression-based GPM extension", July 2024, JVET-AI0115.
- [34] K. Jia et al., "EE2-2.3: Regression-based GPM with intra and inter prediction", Jan. 2025, JVET-AK0101.
- [35] N. Zhang et al., "EE2-3.1/EE2-3.2: Adaptive Reordering of Merge Candidates with Template/Bilateral Matching", July 2021, JVET-W0090.
- [36] Y.-J. Chang et al., "EE2-3.4, EE2-3.5, EE2-3.6: Experimental results of the MV candidates reordering in candidate types based on template matching costs", Jan. 2022, JVET-Y0134.
- [37] S. Hong et al., "EE2-2.5: TMVP Candidate Selection", Jan. 2025, JVET-AK0185.
- [38] Y. Wang et al., "EE2-3.3: GPM with inter prediction and IBC", July 2024, JVET-AI0082.

저 자 소 개



박 내 리

- 2005년 : 세종대학교 전자-정보통신공학과 학사
- 2007년 : 세종대학교 정보통신공학과 석사
- 2007년 ~ 2011년 : 온타임텍 선임연구원
- 2011년 ~ 현재 : LG전자 C&M표준연구소 책임연구원
- 주관심분야 : 영상처리, 비디오 압축



남 정 학

- 2006년 : 광운대학교 컴퓨터공학과 학사
- 2008년 : 광운대학교 컴퓨터공학과 석사
- 2013년 : 광운대학교 컴퓨터공학과 박사
- 2013년 ~ 현재 : LG전자 C&M표준연구소 책임연구원
- 주관심분야 : 영상처리, 비디오 압축



임 재 현

- 2003년 : 연세대학교 전자공학과 학사
- 2005년 : 연세대학교 전자공학과 석사
- 2005년 ~ 현재 : LG전자 C&M표준연구소 책임연구원
- 주관심분야 : 영상처리, 비디오 압축