

특집논문-07-12-6-03

다시점 비디오 부호화를 위한 움직임벡터 예측값 선택 방법

최 원 준^{a)}, 박 광 훈^{a)†}, 서 덕 영^{a)}, 김 규 현^{a)}

Motion Vector Predictor selection method for multi-view video coding

Won-Jun Choi^{a)}, Gwang-Hoon Park^{a)†}, Doug-Young Suh^{a)}, and Kyu-Heon Kim^{a)}

요 약

본 논문에서는 JVT에서 표준화가 진행 중인 다시점 비디오 부호화의 부호화 효율을 높이기 위해 다시점 비디오의 예측구조의 특성을 고려하여 움직임벡터 예측값을 선택하는 방법을 제안한다. 다시점 비디오 부호화에서 시간 방향과 뷰 방향의 예측을 수행하면서 서로 다른 성향의 움직임벡터가 발생된다. 또한 주변 블록에서 시간 방향과 뷰 방향 참조의 움직임벡터가 혼재하는 상황에서 동일한 성향의 움직임벡터가 선택되지 않는 경우가 발생되어 움직임벡터의 예측 효율이 저하된다. 이에 다시점 비디오 부호화의 예측구조에 따라 움직임벡터 예측값을 선택할 수 있는 방법을 제안한다. 제안하는 움직임벡터 예측값 선택 방법을 통하여 다시점 비디오 부호화의 압축 효율을 높일 수 있었으며, JMVM 3.6에서 움직임벡터 예측값을 선택하는 방법을 사용하는 다시점 비디오 부호화와 비교하여 PSNR(Peak Signal-to-Noise Ratio)이 약 0.03 ~ 0.1 dB 정도 향상되었다.

Abstract

In this paper, we propose a method to select motion vector predictor by considering prediction structure of a multi view content for coding efficiency of multi view coding which is being standardized in JVT. Motion vector of a different tendency is happened while carrying out temporal and view reference prediction of multi-view video coding. Also, due to the phenomena of motion vectors being searched in both temporal and view order, the motion vectors do not agree with each other resulting a decline in coding efficiency. This paper is about how the motion vector predictor are selected with information of prediction structure. By using the proposed method, a compression ratio of the proposed method in multi-view video coding is increased, and finally 0.03 ~ 0.1 dB PSNR(Peak Signal-to-Noise Ratio) improvement was obtained compared with the case of JMVM 3.6 method.

Keyword: 다시점 비디오 부호화(Multi-view Video Coding: MVC), Motion Vector Predictor, H.264, MPEG-4 AVC

1. 서 론

현재 ISO/IEC/JTC1/WG11/MPEG과 ITU-T VCEG의 공

a) 경희대학교·전자정보대학 미디어랩
Media Lab., College of Electronics and Information, Kyung Hee University

† 교신저자 : 박광훈(ghpark@khu.ac.kr)

* 이 논문은 2007년도 정부(과학기술부)의 재원으로 한국과학재단의 국가 지정연구실사업으로 수행된 연구임(No.R0A-000-10061-0(2007))

동표준화 그룹인 JVT(Joint Video Team)에서 MPEG-4 AVC/H.264 Amd. 4 MVC(Multi-view Video Coding)으로 표준화가 진행 중이다. MVC는 기존의 동영상 국제표준인 MPEG-4 part 10 Advanced Video Coding (AVC; H.264)^[1] 방법을 기반으로 다시점 비디오를 부호화하며 시간 방향으로는 JSVC (Joint Scalable Video Coding)에서 시간적 스케일러빌리티(temporal scalability)를 지원하기 위해 수행되는 방법인 계층적인 B-픽처(hierarchical B-pictures) 부호화

를 수행하고 또한 뷰 간(inter-view)의 예측을 수행하여 부호화 성능의 향상을 이루고 있다.

지금까지 MVC 표준화 회의를 통해 인접한 뷰 사이에서 예측 수행 시에 효과적인 조명변화 보상(Illumination Compensation: IC) 방법^[2], IC 정보를 유도하는 방법^[3] 그리고 IC 디블록킹 필터링 방법^[4]과 다른 뷰의 매크로블록 모드 정보와 움직임벡터, 참조영상 인덱스 정보를 그대로 이용하여 부호화하는 방법^[5], 그리고 다른 뷰의 픽처로부터 현재 뷰의 픽처를 생성하는 뷰 합성 예측(View Synthesis Prediction) 방법^{[6][7]} 등이 발표되었으며 일부 알고리즘은 참조 소프트웨어인 JMVM(Joint Multi-view Video Model)에 채택되거나 연구가 계속 진행되고 있다.

다시점 비디오 부호화에서는 예측 방향에 따라서 두 가지 종류의 움직임 벡터(MV)를 구분할 수 있다. 각 뷰에서 시간 방향으로 예측을 수행하여 구하는 일반적인 움직임 벡터(MV)와 동일한 시간에서의 뷰 방향으로 예측을 수행하여 구하는 움직임 벡터(MV)가 있다. 뷰 방향의 움직임 벡터(MV)를 디스패리티벡터(Disparity Vector; DV)라고도 한다. 이 두 가지 종류의 움직임 벡터(MV)는 하나의 영상에서 동시에 존재할 수도 있다.

본 논문에서는 다시점 비디오 부호화에서 움직임 예측(Motion Estimation) 및 움직임 보상(Motion Compensation)에서 움직임벡터의 예측값(Motion Vector Predictor: MVP)을 선택할 때, 현재 부호화하는 블록의 주변 움직임 벡터와 현재 부호화하는 영상의 동일한 뷰의 다른 시간대 영상의 움직임벡터 그리고 동일한 시간대의 다른 뷰 영상의 움직임벡터 정보를 움직임벡터 예측값으로 선택하여 사용하는 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 움직임 벡터 예측값 선택 방법을 알아보고, 3장에서는 다시점 비디오 부호화 방법에 적합한 움직임벡터 예측값 선택 방법을 설명한다. 4장에서는 제안한 알고리즘의 실험 결과를 분석하고 5장에서는 결론을 기술한다.

II. 기존의 움직임벡터 예측값 선택 방법

현재 다시점 비디오 부호화에서 움직임벡터의 예측 방법

으로 기존의 동영상 부호화 방법 국제표준인 AVC/H.264에서와 같은 방법을 사용한다. 이것은 기존의 다른 동영상 국제표준들에서도 사용되었던 방법으로써, 움직임 예측을 통해 구해진 현재 블록의 움직임벡터는 주변 블록의 움직임벡터와 유사한 움직임을 갖는다는 가정에 따라 주변 블록의 움직임벡터들로부터 움직임벡터 예측값을 구하여 현재 블록의 움직임벡터에서 움직임벡터 예측값을 차분하여 움직임벡터 차분값(Motion Vector Difference : MVD)을 부호화하여 디코더(Decoder)에 전송함으로써 부호화 성능을 높이는 방법을 사용하고 있다. 디코더에서도 움직임벡터 예측값을 구하는 방법에 대한 정보는 엔코더(Encoder)에서 보내지지 않고 주변 블록의 움직임벡터를 이용하여 엔코더와 동일하게 움직임벡터 예측값이 구해지며, 엔코더에서 디코더로 전송되는 움직임벡터 차분값을 계산하기 위해 수식 (1)을 사용한다.

$$MV_D = MV - MV_P \quad (1)$$

움직임벡터 차분값은 엔코더에서 디코더로 정보가 전송되며, 이를 디코더 단에서 수신하고 나서 움직임벡터를 구해야 한다. 이 때 디코더에서도 마찬가지로 현재 부호화하는 블록의 주변 블록의 움직임벡터들로부터 움직임벡터 예측값을 구하여 현재 블록에서 전송받은 움직임벡터 차분값을 더하여 움직임벡터를 구한다. 디코더 단에서 움직임벡터를 계산하기 위해 수식 (2)을 사용한다.

$$MV = MV_P + MV_D \quad (2)$$

그림 1은 현재 부호화하는 블록과 주변 블록을 나타낸다. 여기서 움직임벡터 예측값을 선택하기 위해 사용하는 블록은 'A', 'B', 'C'가 되며, 이와 같은 경우에 현재 부호화하는 블록의 움직임벡터 예측값은 'A', 'B', 'C' 블록의 움직임벡터들의 중앙값(Median)을 사용하여 선택한다. 이 때 'C' 블록의 움직임벡터를 사용할 수 없는 경우 'D' 블록의 움직임벡터를 사용하게 된다.

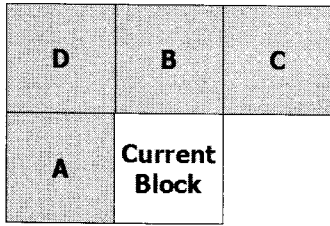


그림 1. 움직임벡터 예측값을 구하기 위한 주변 블록의 위치
Fig 1. Neighbor-block location for Motion vector prediction value

AVC/H.264에서는 하나의 매크로블록이 다양한 블록 크기(16X16, 16X8, 8X16, 8X8, 8X4, 4X8, 4X4 등)로 분할되어 부호화될 수 있기 때문에 움직임벡터 예측값을 구하는 방법이 복잡하고 다양하다. 그림 2는 주변 블록이 분할되어 있는 경우에 현재 블록의 움직임벡터 예측값을 선택하기 위해 사용되는 주변 블록 'A', 'B', 'C'를 나타낸다.

그림 3은 현재 블록이 8x16, 16x8 크기로 부호화되는 경우에 세 개의 주변 블록에서 특정 블록의 움직임벡터를 사용하여 움직임벡터 예측값으로 사용하는 방법을 나타낸다.

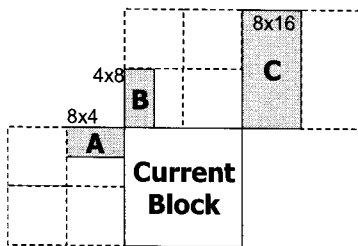


그림 2. 움직임벡터 예측값을 구하기 위한 주변 블록의 위치
Fig 2. Neighbor-block location for Motion vector prediction value

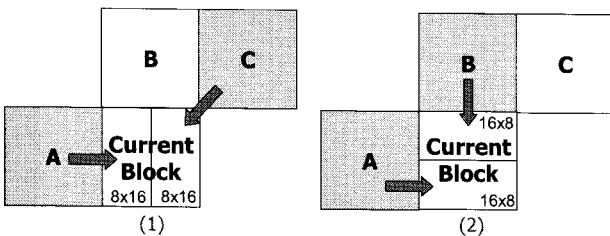


그림 3. 매크로블록 타입이 (1)8x16, (2)16x8인 경우의 움직임벡터 예측값 선택
Fig 3. Motion vector predictor for macroblock type for (1) 8x16, (2) 16x8

AVC/H.264는 예측 수행 시에 복수개의 참조 영상을 사용할 수 있기 때문에 참조 영상(Reference Frame)을 두 개

의 참조영상 리스트(Reference List)로 관리한다. 움직임벡터의 예측값을 선택하기 위해 주변블록 'A', 'B', 'C'의 움직임벡터, 참조영상 리스트 정보와 해당 리스트의 참조 인덱스(Reference Index) 정보를 사용하여 다음과 같은 과정에 따라 움직임벡터 예측값을 선택한다.

- (1) 움직임벡터 예측값으로 블록 'A'의 움직임벡터 (MV_A)를 선택 후 종료 :
 - 예측 방향이 'A' 블록이고 MV_A의 참조인덱스 (RefIdx_A)가 현재 예측하는 참조인덱스(curRefIdx)와 일치할 때
 - RefIdx_A와 curRefIdx와 동일하고 MV_B와 MV_C의 참조인덱스가 curRefIdx와 일치하지 않을 때
 - MV_B와 MV_C가 블록이 존재하지 않을 때
- 움직임벡터 예측값으로 블록 'B'의 움직임벡터 (MV_B)를 선택 후 종료 :
 - 예측 방향이 'B' 블록이고 MV_B의 참조인덱스 (RefIdx_B)가 현재 예측하는 참조인덱스(curRefIdx)와 일치할 때
 - RefIdx_B와 curRefIdx와 동일하고 MV_A와 MV_C의 참조인덱스가 curRefIdx와 일치하지 않을 때
- (2) 움직임벡터 예측값으로 블록 'C'의 움직임벡터 (MV_C)를 선택 후 종료 :
 - 예측 방향이 'C' 블록이고 MV_C의 참조인덱스 (RefIdx_C)가 현재 예측하는 참조인덱스(curRefIdx)와 동일할 때
 - RefIdx_C와 curRefIdx와 동일하고 MV_A와 MV_B의 참조인덱스가 curRefIdx와 일치하지 않을 때
- (3) 움직임벡터 예측값으로 중앙값을 선택 후 종료:
 - (2), (3), (4)의 과정에 해당하지 않을 때

III. 다시점 비디오 부호화에 적합한 움직임벡터 예측값 선택 방법

다시점 비디오 부호화에서 예측이 수행되는 참조 영상의 방향에 따라 서로 다른 움직임벡터가 발생한다. 이 두 가지

종류의 움직임벡터는 다시점 비디오 부호화에서 예측 구조에 따라서 혼재하게 된다.

이 때, 주변 블록이 모두 시간 방향 참조의 움직임벡터를 가지고, 현재 부호화하는 블록이 뷰 방향의 움직임벡터를 가지게 되는 경우에 주변 블록의 움직임벡터와 현재 블록의 움직임벡터는 서로 다른 성향을 나타내어 적절한 움직임벡터 예측값이 선택되지 않는다. 또한 주변 블록에서 시간 방향과 뷰 방향 참조의 움직임벡터가 혼재하는 상황에서도 동일한 성향의 움직임벡터가 선택되지 않는 경우가 발생하게 된다.

1. 다시점 비디오 부호화에서 예측구조 판별

다시점 비디오 부호화의 예측구조는 참조 소프트웨어인 JMVM의 엔코더에서 설정한 정보에 따라 결정된다. 예측구조를 결정하기 위한 정보는 참조 소프트웨어인 JMVM에서 멀티뷰 - 시퀀스파라미터셋(Multiview Video Coding SequenceParameterSet)의 anchor_ref_listX, non_anchor_ref_listX 정보를 이용하여 판별하게 된다. 구분할 수 있는 예측 구조는 그림 4에서와 같이 구분할 수 있다.

- (1) 뷰 방향 예측만 수행되는 구조
 - anchor_ref_listX 값이 활성화
 - 동일 시간의 Anchor축에 있는 영상에서 다른 뷰의 영상을 참조하여 부호화 수행
 - View0을 제외한 T0축 영상이 해당(Anchor영상)
- (2) 시간 방향, 뷰 방향 예측이 수행되는 구조
 - non_anchor_ref_listX 값이 활성화
 - 동일 뷰의 시간 방향 영상과 동일 시간의 다른 뷰 영상을 참조하여 부호화 수행
 - View1, 3의 non-Anchor영상이 해당
- (3) 시간 방향 예측이 수행되는 구조
 - anchor_ref_listX의 값이 활성화 되어 있고 non_anchor_ref_listX 값이 활성화 되어 있지 않은 경우
 - 동일 뷰의 시간 방향 영상을 참조하여 부호화 수행
 - View2의 non-Anchor영상이 해당

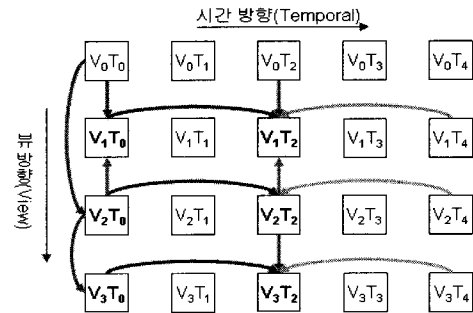


그림 4. 다시점 비디오 부호화에서 예측구조
Fig 4. Prediction structure of multi-view video coding

2. 예측구조에 따른 움직임벡터 예측값 선택 블록 집합

예측구조 판별 후, 움직임벡터 예측값을 선택하기 위한 블록 집합을 설정하게 되며 사용 가능한 블록 집합을 그림 5에 나타내었다. 사용 가능한 블록 집합은 다음과 같다.

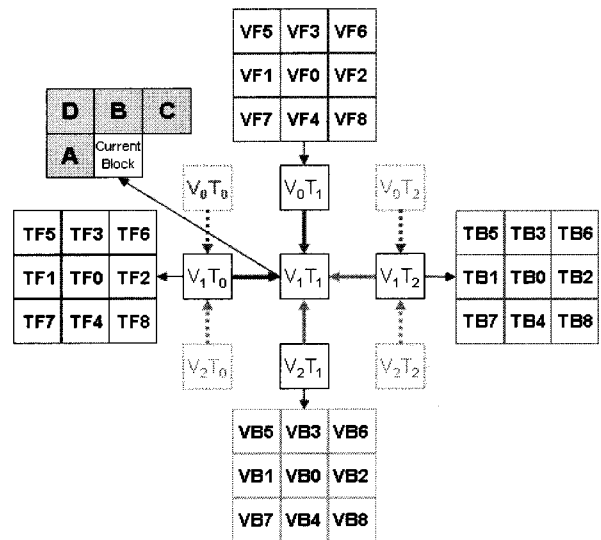


그림 5. 움직임벡터 예측값을 선택하기 위한 블록 집합
Fig 5. Block sets for motion vector predictor

- 현재 부호화하는 블록(Current Block)의 주변 : 'A', 'B', 'C',
- 전방 시간방향(Temporal-Forward: TF) 참조 영상의 주변 : 'TF0', ... , 'TF8'
- 후방 시간방향(Temporal Backward: TB) 참조 영상의 주변 : 'TB0', ... , 'TB8'

- 전방 뷰방향(View Forward: VF) 참조 영상의 주변 : 'VF0', ... , 'VF8'
- 후방 뷰방향(View Backward : VB) 참조 영상의 주변 : 'VB0', ... , 'VB8'

여기서 블록 'TF0', 'TB0'의 위치는 Current Block의 위치와 동일한 위치를 설정하고 전방, 후방 뷰에 해당하는 블록 'VF0', 'VB0'의 경우 전역 변이 벡터(Global Disparity Vector: GDV) 정보^[5]를 사용하여 위치를 설정한다.

예측구조에 따라 움직임벡터 예측값을 선택하기 위한 블록 집합은 달라진다. 예측 구조에 따른 움직임벡터 예측값 블록 집합의 설정은 다음과 같이 이루어진다.

- 뷰 방향 예측만 수행되는 구조 : 'A', 'B', 'C'
- 시간 방향, 뷰 방향 예측이 수행되는 구조 : 'A', 'B', 'C', 'TF0',..., 'TF8', 'TB0',..., 'TB8', 'VF0',..., 'VF8', 'VB0',..., 'VB8'
- 시간 방향 예측만 수행되는 구조 : 'A', 'B', 'C', 'VF0',..., 'VF8', 'VB0',..., 'VB8'

3. 움직임벡터 예측값 선택 알고리즘

움직임벡터 예측값 선택을 위한 블록 집합을 설정한 후, 현재 부호화하는 블록의 움직임벡터를 찾기 위해 움직임추정을 수행하는 참조 영상과 동일한 영상 및 동일한 참조 방향에서 생성된 움직임벡터를 예측값으로 선택한다. 움직임벡터 예측값을 선택하는 방법은 현재 부호화하는 블록의 주변 블록에서 동일한 참조 영상을 가지는 움직임벡터가 존재하는 경우 예측값으로 선택하고 존재하지 않는다면 예측 구조에 따라 동일한 시간 방향, 동일한 뷰 방향의 참조를 가지는 움직임벡터를 검색하여 예측값으로 선택하게 된다.

현재 부호화하는 블록의 주변 블록에서 예측값을 선택하는 방법은 다음과 같다.

- (1) 움직임벡터 예측값으로 블록 'A'의 움직임벡터 (MV_A)를 선택 후 종료 :
 - 예측 방향이 'A' 블록이고 MV_A의 참조인덱스 (RefIdx_A)가 현재 예측하는 참조인덱스(curRefIdx)

와 일치할 때

- RefIdx_A와 curRefIdx와 동일하고 MV_B와 MV_C의 참조인덱스가 curRefIdx와 일치하지 않을 때
 - RefIdx_A와 curRefIdx와 동일하고 MV_B와 MV_C가 블록이 존재하지 않을 때
- (2) 움직임벡터 예측값으로 블록 'B'의 움직임벡터 (MV_B)를 선택 후 종료 :
 - 예측 방향이 'B' 블록이고 MV_B의 참조인덱스 (RefIdx_B)가 현재 예측하는 참조인덱스(curRefIdx)와 일치할 때
 - RefIdx_B와 curRefIdx와 동일하고 MV_A와 MV_C의 참조인덱스가 curRefIdx와 일치하지 않을 때
 - (3) 움직임벡터 예측값으로 블록 'C'의 움직임벡터(MV_C)를 선택 후 종료 :
 - 예측 방향이 'C' 블록이고 MV_C의 참조인덱스 (RefIdx_C)가 현재 예측하는 참조인덱스(curRefIdx)와 동일할 때
 - RefIdx_C와 curRefIdx와 동일하고 MV_A와 MV_B의 참조인덱스가 curRefIdx와 일치하지 않을 때
 - 움직임벡터 예측값으로 중앙값을 선택 후 종료:
 - MV_A, MV_B, MV_C의 각각의 참조인덱스가 두 개 이상 curRefIdx와 일치할 때

위의 과정에서 동일한 예측 방향을 가지는 움직임벡터가 존재하지 않는 경우 현재 부호화하는 블록의 참조 방향에 따라 시간 방향, 뷰 방향의 참조영상에서 동일한 예측 방향의 움직임벡터를 검색하여 예측값을 선택한다. 시간 방향, 뷰 방향의 참조영상에서 예측값을 선택하는 방법은 다음과 같다.

- (1) 현재 예측 방향이 시간 방향인 경우:
 - FV와 BV 블록 집합에서 동일한 시간 방향에서 참조를 가지는 움직임벡터를 검색하여 저장한 후, 중앙값 연산을 수행하여 움직임벡터 예측값으로 사용한다.
- (2) 현재 예측 방향이 뷰 방향인 경우:
 - FT와 BT 블록 집합에서 동일한 뷰 방향에서 참조를 가지는 움직임벡터를 검색하여 저장한 후, 중앙값 연산을 수행하여 움직임벡터 예측값으로 사용한다.

IV. 실험 결과

1. 실험 환경

본 논문에서 제안한 방법의 실험을 위하여 다시점 비디오 부호화의 참조 소프트웨어인 JMVM 3.6을 기반으로 제안하는 움직임벡터 예측값 선택 방법을 구현하였다.

표 1은 실험에 사용되는 테스트 영상의 특성을 나타낸 것이다. 이 영상은 다시점 비디오 부호화 표준화에 사용되고 있는 영상이다.

표 1. 테스트 영상

Table 1. Test Sequences

영상	영상 속성	카메라 배열
Race1	640x480, 30fps 300 frame/view	8 camera 1D parallel
Breakdancers	1024x768, 15fps 100 frame/view	8 camera 1D arc
Akko&Kayo	640x480, 30fps 300 frame/view	15 camera 2D array
Ballroom	640x480, 25fps 250 frame/view	8 camera 1D parallel
Exit	640x480, 25fps 250 frame/view	8 camera 1D parallel

실험 조건은 표 2와 같이 시간 축으로 임의 접근(Random Access) 시간을 0.5초 또는 1초로 제약한다. 기본 QP 값을 22, 27, 32, 27로 고정하고 계층적인 B 부호화를 위해 Delta Qp 값을 표 3과 같이 사용하여 Bjontegaard 측정 법^[8]으로 평균 PSNR/bitrate를 실험하였다.

표 2. 부호화 조건

Table 2. Coding Conditions

영상	임의 접근 시간 간격
Race1	0.5 sec
Breakdancers	1.0 sec
Akko&Kayo	0.5 sec
Ballroom	0.5 sec
Exit	0.5 sec

표 3. Delta QP 값

Table 3. Delta QP Value

Delta QP Value	
DeltaLayer0Quant	0
DeltaLayer1Quant	3
DeltaLayer2Quant	4
DeltaLayer3Quant	5
DeltaLayer4Quant	6
DeltaLayer5Quant	7

2. 실험 결과

다시점 비디오 부호화에 적합하게 제안한 움직임벡터 예측값 선택 방법을 적용하여 실험한 결과 표 4와 같이 Race1에서 0.1 dB, 2.4%, Breakdancers에서 0.05 dB, 2.2%, Akko&Kayo에서 0.1 dB, 2.4%, Ballroom에서 0.04dB, 1.1%, Exit에서 0.03 dB, 1.2%의 성능향상이 이루어 졌다.

표 4. 실험 결과

Table 4. Experiment result

	JMVM3.6		Proposed		gain		BD-Psnr	
	bit-rate	psnr	bit-rate	psnr	bit-saving	psnr	%	dB
Ballroom								
QP=37	193.07	31.72	190.56	31.75	-1.3%	0.02	-1.1%	0.04
QP=32	351.75	34.47	348.70	34.49	-0.9%	0.02		
QP=27	687.02	37.08	682.98	37.09	-0.6%	0.01		
QP=22	1411.80	39.26	1407.64	39.26	-0.3%	0.00		
Exit								
QP=37	100.40	34.35	99.69	34.39	-0.7%	0.04	-1.2%	0.03
QP=32	173.42	36.66	172.80	36.68	-0.4%	0.03		
QP=27	339.61	38.52	338.77	38.54	-0.2%	0.02		
QP=22	828.20	40.01	826.89	40.02	-0.2%	0.01		
Race1								
QP=37	157.34	33.14	152.91	33.20	-2.8%	0.05	-2.4%	0.10
QP=32	274.96	35.90	269.26	35.94	-2.1%	0.04		
QP=27	546.17	38.61	539.39	38.63	-1.2%	0.02		
QP=22	1132.91	41.24	1124.10	41.25	-0.8%	0.01		
Flamenco2								
QP=37	218.15	33.30	217.30	33.33	-0.4%	0.03	-0.7%	0.03
QP=32	396.65	36.24	395.56	36.26	-0.3%	0.02		
QP=27	747.64	39.13	745.68	39.15	-0.3%	0.01		
QP=22	1420.01	41.82	1416.85	41.83	-0.2%	0.01		
Uli								
QP=37	691.90	32.07	690.92	32.07	-0.1%	0.00	-0.2%	0.01
QP=32	1244.05	34.79	1242.20	34.80	-0.1%	0.00		
QP=27	2252.25	37.25	2249.17	37.25	-0.1%	0.00		
QP=22	4292.68	39.16	4288.36	39.16	-0.1%	0.00		
Breakdancers								
QP=37	129.76	35.05	127.14	35.08	-2.0%	0.03	-2.2%	0.05
QP=32	227.32	37.10	223.81	37.12	-1.5%	0.03		
QP=27	449.31	38.62	444.66	38.64	-1.0%	0.01		
QP=22	1128.07	39.84	1122.39	39.84	-0.5%	0.01		
Akko&Kayo								
QP=37	183.04	33.10	180.86	33.15	-1.2%	0.05	-1.5%	0.07
QP=32	302.80	36.12	300.44	36.17	-0.8%	0.05		
QP=27	543.19	39.11	539.79	39.14	-0.6%	0.03		
QP=22	1050.39	41.77	1046.26	41.79	-0.4%	0.02		
Rena								
QP=37	92.63	35.74	92.29	35.75	-0.4%	0.02	-0.6%	0.03
QP=32	161.97	38.80	161.08	38.82	-0.5%	0.01		
QP=27	319.40	41.97	318.53	41.97	-0.3%	0.01		
QP=22	667.91	44.78	667.15	44.78	-0.1%	0.00		

Race1과 Breakdancers 영상에서 8개의 뷰 중에 non_Anchor 영상에서 인접한 뷰를 예측 참조하여 부호화를 수행하는 뷰는 1, 3, 5, 7번 뷰이고 Anchor 영상에서만 인접한 뷰를 참조하여 부호화를 수행하는 뷰는 2, 4, 6번 뷰이다. Akko&Kayo 영상은 non_Anchor 영상에서 인접한 뷰를 참조하여 부호화를 수행하는 뷰는 1, 3, 6, 8, 11, 13번 뷰이고 Anchor 영상에서만 인접한 뷰를 참조하여 부호화를 수행하는 뷰는 2, 4, 7, 9, 10, 12, 14번 뷰이다. 표5, 6, 7은 Race1, Breakdancers, Akko&Kayo에서 Qp37의 실험 결과를 뷰별로 나타낸 것이다.

표 5. Race1 Qp37의 실험 결과

Table 5. Experiment result of Race1 Qp37

QP37	Race1					
	JMVM3.6		Proposed		bit-saving	gain
View	Rate	PSNR	Rate	PSNR		
0	236.87	32.90	236.87	32.90	0.00%	0.00
1	121.82	33.22	117.41	33.30	-3.62%	0.08
2	166.20	32.80	164.22	32.82	-1.19%	0.02
3	114.92	33.05	108.95	33.13	-5.19%	0.09
4	173.53	34.30	170.66	34.33	-1.65%	0.03
5	128.80	34.04	121.97	34.16	-5.30%	0.12
6	191.57	32.12	185.46	32.14	-3.19%	0.03
7	125.00	32.71	117.77	32.79	-5.79%	0.08
Avg	157.34	33.14	152.91	33.20	-2.81%	0.05

표 6. Breakdancers Qp37의 실험 결과

Table 6. Experiment result of Breakdancers Qp37

QP37	Breakdancers					
	JMVM3.6		Proposed		bit-saving	gain
View	Rate	PSNR	Rate	PSNR		
0	165.01	35.15	165.01	35.15	0.00%	0.00
1	112.86	34.93	107.96	35.00	-4.35%	0.07
2	140.69	35.09	139.88	35.09	-0.58%	0.00
3	116.97	34.83	111.21	34.87	-4.92%	0.04
4	132.09	35.37	131.08	35.37	-0.76%	0.00
5	109.24	34.94	103.63	35.03	-5.14%	0.09
6	143.32	35.08	142.95	35.09	-0.26%	0.01
7	117.90	35.02	115.40	35.06	-2.12%	0.04
Avg	129.76	35.05	127.14	35.08	-2.02%	0.03

표 7. Akko&Kayo Qp37의 실험 결과

Table 7. Experiment result of Akko&Kayo Qp37

QP37	Akko&Kayo					
	JMVM3.6		Proposed		bit-saving	gain
View	Rate	PSNR	Rate	PSNR		
0	220.70	33.81	220.70	33.81	0.00%	0.00
1	121.54	33.89	117.94	34.02	-2.97%	0.14
2	153.62	34.02	152.00	34.03	-1.05%	0.01
3	132.98	33.60	128.17	33.73	-3.62%	0.12
4	179.62	33.25	178.17	33.25	-0.80%	0.00
5	233.65	33.08	233.44	33.07	-0.09%	0.00
6	161.66	33.08	157.25	33.21	-2.73%	0.13
7	196.58	32.36	195.25	32.36	-0.68%	-0.01
8	150.86	33.29	146.86	33.44	-2.65%	0.15
9	202.29	32.68	200.60	32.68	-0.83%	0.00
10	251.03	32.27	250.19	32.27	-0.34%	0.00
11	156.94	32.81	152.83	32.94	-2.62%	0.13
12	210.47	32.79	209.66	32.79	-0.38%	0.00
13	153.17	32.53	150.05	32.67	-2.04%	0.14
14	220.46	32.98	219.73	32.98	-0.33%	0.00
Avg	183.04	33.10	180.86	33.15	-1.19%	0.05

제안한 움직임벡터 예측값 선택 방법의 효율성이 나타나 는 부분을 알아보기 위해 실험 결과를 AVC/H.264와의 호 환성을 유지해야 하는 기본 뷰(0번 뷰)를 제외하고 non_Anchor 영상의 부호화 시에 시간 방향 참조와 뷰 방향 참조 를 사용하는 경우와 시간 방향 참조만 사용하는 두 가지 경우로 나누어 성능 향상을 분석한다.

그 결과 표 8에서와 같이 non-Anchor에서 뷰를 참조하는 경우에 Anchor에서만 뷰를 참조하는 경우 보다 비트감소 가 2~5배, PSNR 3~14배 정도 대부분의 성능향상이 non_Anchor 영상에서 뷰를 참조하는 경우에 발생하는 것을 알 수 있다.

표 8. 다시점 예측구조에 따른 성능 비교

Table 8. Comparison of the performance multi-view prediction structure

Qp37	예측 구조		non-Anchor 뷰 참조(O)		non-Anchor 뷰 참조(X)	
	Sequence	bit-saving	psnr	bit-saving	psnr	psnr
Race1		-4.98%	0.09	-2.21%	0.03	
Breakdancers		-4.13%	0.06	-0.91%	0.01	
Akko&Kayo		-2.77%	0.14	-0.56%	0.00	

V. 결론

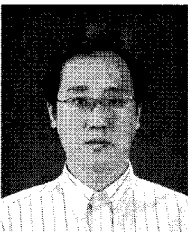
본 논문에서는 다시점 비디오 부호화에서 움직임벡터 예측값을 선택하는데 있어 현재 부호화하는 블록의 주변 블록에서만 결정하는 기존의 방법과는 다르게 부호화하는 영상의 동일한 뷰 내의 시간 방향 영상과 동일한 시간대의 인접한 뷰 영상의 움직임벡터 정보를 예측값으로 사용하는 방법을 제안하였다. 그 결과 Race1 영상에서 기존의 JMVM 3.6과 비교하여 0.1 dB, 2.4%의 성능향상이 이루어졌다. 그리고 5개의 영상(Race1, Breakdancers, Akko& Kayo, Ballroom, Exit)에서 평균적으로 0.06 dB, 1.7%의 성능향상이 이루어졌다. 다시점 비디오 부호화에서 non-Anchor 영상의 뷰 간 예측이 늘어난다면 더 나은 성능을 기대할 수 있을 것이다.

향후 연구로 다시점 비디오 부호화의 성능을 높일 수 있는 다른 방법들과 조합하여 더 나은 성능 향상이 가능하도록 하는 연구가 필요하다.

참고 문헌

- [1] ITU-T and ISO/IEC JTC1, "Advanced Video Coding for Generic Audiovisual Services," ITU-T Recommendation H.264 ISO/IEC 14496-10 AVC, 2003.
- [2] Yung-Lyul Lee, and et al., "CE11: Illumination Compensation," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-052, Hangzhou, October 2006.
- [3] Jeong-Hyu Yang, "CE09: Derivation of MVC IC Information," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-V063, January, 2007
- [4] Woo Sung Shim, "CE5 Results-Joint Proposal for MVC Deblocking", ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-W024, April, 2007
- [5] Han-Suh Koo, "MVC Motion Skip Mode," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-W081, April, 2007
- [6] S. Yea, A. Vetro, "CE10: View synthesis prediction," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-W059, San Jose, US, April, 2007.
- [7] S. Shimizu, H. Kimata, "MVC view synthesis residual prediction," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-X084, Geneva, CH, July 2007.
- [8] G. Bjontegaard, "Calculation of average PSNR differences between RD-Curves", document VCEG-M33, 13th VCEG meeting, Austin TX, Mar'01

저자 소개



최원준

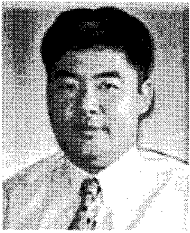
- 2002년 : 경희대학교 컴퓨터공학과 학사
- 2004년 : 경희대학교 컴퓨터공학과 석사
- 2004년 ~ 현재 : 경희대학교 컴퓨터공학과 박사과정
- 주관심분야 : 멀티미디어, 비디오 신호처리, 영상처리



박광훈

- 1985년 : 연세대학교 전자공학과 학사
- 1987년 : 연세대학교 전자공학과 석사
- 1991년 : Case Western Reserve Univ., Dept. of EEAP (M.S.)
- 1995년 : Case Western Reserve Univ., Dept. of EEAP (Ph.D.)
- 1995년 3월 ~ 1997년 2월 : 현대전자 멀티미디어 연구소 책임연구원
- 1997년 3월 ~ 2001년 2월 : 연세대학교 문리대학 전산학과 부교수
- 2001년 3월 ~ 현재 : 경희대학교 전자정보대학 컴퓨터공학 전공 교수
- 주관심분야 : 멀티미디어, 비디오 신호처리, 패턴인식, 영상처리, 계산지능 등

저 자 소 개



서 덕 영

- 1980년 2월 : 서울대학교 핵공학과 학사
- 1986년 6월 : 미국 조지아텍 핵공학과 석사
- 1990년 6월 : 미국 조지아텍 전기 및 컴퓨터공학과 박사
- 1992년 3월 ~ 현재 : 경희대학교 전자정보대학 전자전파공학 전공 교수
- 주관심분야 : Networked Multimedia(MPEG standardization, Mobile Internet, VOD, Video Phone)



김 규 현

- 1989년 : 한양대학교 전자공학과 학사
- 1992년 : University of Newcastle 전기전자공학과 석사
- 1996년 : University of Newcastle 전기전자공학과 박사
- 1997년 9월 ~ 2006년 02월 : 한국전자통신연구원 대화형미디어연구 팀장
- 2001년 7월 ~ 2005년 07월 : MPEG 한국대표단장
- 2005년 9월 ~ 2006년 02월 : 과학기술연합대학원대학교(UST) 부교수
- 2006년 2월 ~ 현재 : 경희대학교 전자정보대학 전자전파공학 전공 부교수
- 주관심분야 : 영상처리, 멀티미디어통신, 디지털대화형방송