

에러 강인성을 위한 모순 검증 기반 움직임 벡터 부-복호화

원 광 현^{a)}, 양 정 엽^{a)}, 전 병 우^{a)‡}

Motion Vector Coding with Error Robustness Based on Contradiction Testing

Kwanghyun Won^{a)}, Jungyoup Yang^{a)}, and Byeungwoo Jeon^{a)‡}

요 약

본 논문은 에러 발생 가능한 예측 움직임 벡터를 제외한 후보 예측 움직임 벡터들 중 율-왜곡 관점에서 최적의 예측 움직임 벡터를 선택하고 그에 대한 인덱스 정보에 대해 모순 검증 기술을 적용한다. 이를 통하여 후보 예측 움직임 벡터 대상들 중 모순으로 판명된 일부를 대상 후보 집합에서 제외시켜 부호화 효율을 증가시키는 방법을 제안한다. 제안하는 모순 검증 기반 움직임 벡터 부호화 방법은 H.264/AVC 방법과 비교하여 최대 1.8%의 비트량을 절감시킨다.

Abstract

This paper proposes a motion vector coding method which selects the optimal predictive motion vector after excluding some candidate predictive motion vectors in the minimum rate-distortion sense, and prunes the candidate predictive motion vectors by a contradiction testing that operates under a predefined criterion at both encoder and decoder for reducing the signaling information. Experimental results show that the proposed method gains in BDBR by up to 1.8% compared to the H.264/AVC.

Keyword : H.264/AVC, video compression, motion vector coding, contradiction testing, motion vector predictor

1. 서 론

비디오 압축을 위해 부호화 되는 주요 정보들은 크게 모드(Mode), 영상(Texture), 움직임 벡터(Motion Vector) 세 종류로 구분될 수 있다. 특히 부호화된 비트스트림 내의 움직임 벡터 정보는 낮은 비트율로 갈수록 다른 영상 정보에

비해 상대적으로 많은 부분을 차지하게 된다. 이는 움직임 벡터 부호화 기술이 전체적인 압축율을 향상시킬 수 있는 중요한 요소임을 의미한다.

H.264/AVC 부호화 표준에서는 움직임 벡터를 부호화하는데 소요되는 비트량을 감소시키기 위해 주변 블록에 대한 움직임 벡터들의 중간값으로 얻어진 예측 움직임 벡터와 현재 블록의 움직임 벡터와의 차분값을 부호화하는 방법을 사용한다^[1-3]. 이것은 주변 블록의 움직임 벡터들의 중간값으로 얻어진 예측 움직임 벡터가 현재 블록의 움직임 벡터와 매우 유사할 것이라는 확률적 근거에 기반한 것이다. 그러나 중간값을 통해 얻어진 예측 움직임 벡터는 현재

a) 성균관대학교 정보통신공학부

School of Electronic and Electrical Engineering, Sungkyunkwan Univ.

‡ 교신저자 : 전병우 (bjeon@skku.edu)

※ 이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국연구재단의

지원을 받아 수행된 연구임(No. 2011-001-7578)

· 접수일(2011년12월30일), 수정일(2012년2월27일), 게재확정일(2012년3월26일)

블록의 움직임 벡터와의 차분값을 항상 최소로 만드는 최적의 예측값은 아니다.

전술한 문제를 해결하기 위한 여러 연구가 진행되어 왔는데, 그 중 가장 효율적인 압축 성능을 보이는 기술 중 하나가 MVComp(Motion Vector Competition) 방법이다^{[4][13]}. MVComp 방법은 시공간적으로 위치한 인접 블록들로부터 구성된 예측 움직임 벡터 후보들 중 윗-왜곡 측면에서 최적인 예측 움직임 벡터를 선택한 후, 그에 대한 차분값만을 전송하는 기술이다^[13]. 즉, H.264/AVC에서 이용한 공간적 위치뿐만 아니라 시간적 위치에 있는 다수의 인접 블록 움직임 벡터들 중에서 부호화 효율 측면에서 최적인 움직임 벡터를 예측값으로 선택하고 어떤 움직임 벡터가 예측값으로 사용되었는지에 대한 추가적인 인덱스를 함께 부호화한다. MVComp 기술은 추가적인 인덱스의 부호화 효율을 높이기 위해 예측 움직임 벡터 후보들 중 서로 동일한 것들에 대해서 그 중 하나만 후보에 남기고 나머지는 후보에서 제거하여 예측 움직임 벡터 후보의 개수를 감소시켜 부호화 효율을 향상시킨다. 그러나 MVComp 기술에 포함된 예측 움직임 벡터 후보의 개수를 감소시키는 방법은 통신 및 기타 환경요인에 의해 참조영상내 에러가 발생하였을 경우, 복호화기 판독 과정에 있어서 파싱 오류를 유발시킬 수 있다.

본 논문에서는 에러 발생 가능한 예측 움직임 벡터를 제외한 후보 예측 움직임 벡터들 중 모순 검증을 기반으로 현재 블록의 예측 움직임 벡터로 유효하지 않은 것들을 제외시켜 예측 움직임 벡터의 후보 개수를 감소시킴으로써 예측 움직임 벡터 인덱스를 부호화 하는데 필요한 비트량을 감소시키는 방법을 제안한다. 제안 방법은 현재 움직임 벡터에 대한 최적의 예측 움직임 벡터를 윗-왜곡 측면에서 선택한다는 사실을 근거로 모순 검증 과정을 거침으로써 모순 여부를 판단한다. 본 논문의 구성은 다음과 같다. 우선 II장에서는 H.264/AVC 부호화 표준에서 채택한 움직임 벡터 압축 방법과 MVComp 방법에서 제안한 움직임 벡터 압축 방법에 대해 전반적인 구조 및 동작을 설명하고 그 한계점에 대한 문제를 제기한다. III장에서는 그 한계를 극복하기 위한 제안 방법의 기본 개념 및 구체적 압축 방법에 대해서 설명한다. IV장에서는 실험 및 결과에 대해서 분석하고, 마지막으로 V장에서는 결론에 대해서 논의한다.

II. 움직임 벡터 압축 기술

본 장에서는 H.264/AVC 압축 표준에서 채택한 움직임 벡터 부호화 기술 및 MVComp 방법에서 제안한 움직임 벡터 부호화 기술을 설명하고 이러한 방법의 한계점에 대한 문제를 제기한다.

1. H.264.AVC 압축 표준의 움직임 벡터 부호화 기술

H.264/AVC 압축 표준에서는 보다 향상된 압출 효율을 달성하기 위해 이미 복원된 주변 블록의 움직임 벡터를 통해 예측 움직임 벡터를 생성하고 이를 이용해 해당 움직임 벡터와의 차분 움직임 벡터(Differential Motion Vector) $\overrightarrow{dmv}^{Curr}$ 를 식 (1)과 같이 계산하고, 이를 엔트로피 부호화(Entropy Coding)하여 복호화기에 전송하는 기술을 채택하고 있다^[1]. 이러한 이유는 주변 블록이 가지는 움직임 벡터가 현재 블록의 움직임 벡터와 많은 경우 매우 유사하여 차분값을 (0,0)에 가깝게 만들어 엔트로피 부호화 시에 압출 효율을 향상시킬 수 있기 때문이다.

$$\overrightarrow{dmv}^{Curr} = \overrightarrow{mv}^{Curr} - \overrightarrow{pmv}^{Curr} = \begin{Bmatrix} mv_x^{Curr} - pmv_x^{Curr} \\ mv_y^{Curr} - pmv_y^{Curr} \end{Bmatrix} \quad (1)$$

그림 1은 H.264/AVC 부호화 표준에서 해당 움직임 벡터를 위한 예측값을 생성하는데 이용되는 공간적으로 인접한 블록의 움직임 벡터를 보여준다. 그림 1에서 $\overrightarrow{mv}^{Curr}$ 는 부호화할 현재 블록이 가지고 있는 움직임 벡터를 의미하며, 이

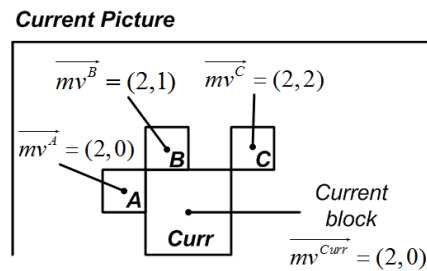


그림 1. 현재 블록과 공간적으로 이웃한 블록들(A, B, 그리고 C)
Fig. 1. Current block and its spatially neighboring blocks (A, B, and C)

는 식 (1)에서와 같이 x축 성분 mv_x^{Curr} 와 y축 성분 mv_y^{Curr} 로 구성된다. $\overrightarrow{pmv}^{Curr}$ 는 주변 블록의 움직임 벡터를 통해 얻어지는 예측 움직임 벡터(Predictive Motion Vector)를 의미하며, 동일하게 x축 성분과 y축 성분으로 구성된다. 예측 움직임 벡터는 식 (2)와 같이 인접한 움직임 벡터들의 중간값으로 계산된다.

$$\overrightarrow{pmv}^{Curr} = \begin{cases} pmv_x^{Curr} \\ pmv_y^{Curr} \end{cases} = \begin{cases} median(mv_x^A, mv_x^B, mv_x^C) \\ median(mv_y^A, mv_y^B, mv_y^C) \end{cases} \quad (2)$$

그림 1에서와 같이 $\overrightarrow{mv}^A, \overrightarrow{mv}^B, \overrightarrow{mv}^C$ 는 각각 주변 블록의 움직임 벡터를 의미하고, $median$ 함수는 입력된 움직임 벡터들의 x축 성분과 y축 성분 각각의 중간값을 정의하는 함수이다. 이러한 방법은 중간값을 통해 얻어진 예측 움직임 벡터가 현재 움직임 벡터와 많은 경우 상당히 유사하기 때문에 엔트로피 부호화하는데 소요되는 비트량을 확률적으로 최소화할 수 있고, 부호화기와 복호화기가 동일한 $median$ 함수를 사용할 수 있으므로 예측 움직임 벡터를 지시하거나 생성하기 위한 추가적인 정보가 불필요하다는 장점이 있다. 중간값을 통해 얻어진 예측 움직임 벡터는 확률적으로 해당 움직임 벡터와 가장 유사한 예측값이지만, 항상 움직임 벡터의 차분값을 최소로 만드는 최적의 예측값은 아니다. 예를 들어, 그림 1에서와 같이, 현재 움직임 벡터 $\overrightarrow{mv}^{Curr}$ 가 (2,0)이고, 중간값을 통해 얻어진 예측 움직임 벡터 $\overrightarrow{pmv}^{Curr}$ 가 (2,1)이라고 가정하면, 차분 움직임 벡터 $\overrightarrow{dmv}^{Curr}$ 는 (0,1)이 된다. 만약 중간값을 사용하지 않고 블록 A의 움직임 벡터를 예측 움직임 벡터로 선택 할 경우, 차분 움직임 벡터는 (0,0)이 된다. 즉, 중간값을 이용한 예측 움직임 벡터가 항상 최적의 예측 움직임 벡터는 아니라는 문제점을 야기하기 때문에 최적의 예측 움직임 벡터를 선택할 수 있는 개선된 기술이 필요하다.

2. MVComp 방법의 움직임 벡터 부호화 기술

MVComp 방법은 그림 1과 같이 주변 블록에 속한 움직임 벡터뿐만 아니라 그림 2와 같이 참조영상의 동일 위치 블록에 포함되어 있는 움직임 벡터까지 예측 움직임 벡터

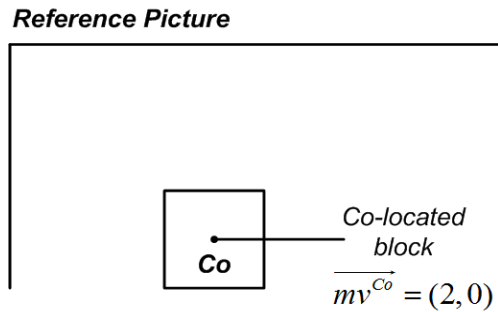


그림. 2. 시간적으로 이웃한 블록(Co-located block)
Fig. 2. Temporally neighboring block (Co-located block)

후보로 사용한다^[13]. 즉, 그림 1과 그림 2와 같이 시공간적으로 인접한 블록의 움직임 벡터 각각에 대해 식 (3)의 윌-왜곡 비용 함수를 계산하고, 그 중에서 가장 부호화 효율이 우수한 예측 움직임 벡터를 선택하여 현재 움직임 벡터와의 차분값을 계산하여 복호화기에 전송함으로써 부호화 효율 측면에서 최적인 예측 움직임 벡터를 선택하도록 한다.

$$J = D + \lambda(R + R_m + R_{mv}) \quad (3)$$

J 는 윌-왜곡 비용, D 는 원본영상과 복원영상간의 왜곡, λ 는 lagrange multiplier, R 은 전체 비트량에서 차분 움직임 벡터에 대한 비트량과 최적의 예측 움직임 벡터 인덱스에 대한 비트량이 제외된 비트량, R_{mv} 는 차분 움직임 벡터에 대한 비트량, R_m 은 최적의 예측 움직임 벡터를 가르키는 인덱스에 대한 비트량을 의미한다. MVComp 방법은 R_m 의 비트량을 감소시키기 위해 예측 움직임 벡터 후보들 중 서로 동일한 것들에 대해서 단 하나의 예측 움직임 벡터 후보를 제외한 나머지 후보들을 제거하여 예측 움직임 벡터 후보의 개수를 감소시켜 최적의 예측 움직임 벡터를 표현하기 위한 코드 길이를 감소시킴으로써 부호화 효율을 높일 수 있다. 그러나 MVComp 기술에 포함된 예측 움직임 벡터 후보의 개수를 감소시키는 방법은 통신장애에 따른 패킷손실로 인해 정상적으로 복호화하지 못한 영상이 참조영상으로 되거나 또는 외부 요인(예를 들어, 메모리 오버플로어)에 의해 메모리에 잡음이 발생하여 참조영상이 훼손되어 부-복호화기가 공히 가지고 있는 시간적 예측 움직임 벡

터의 정보가 서로 달라질 경우, 복호화가 판독 과정에 있어서 파싱 오류가 발생할 수 있다. 즉, 부호화기에서 예측 움직임 벡터 후보 개수를 감소시켜 그에 해당하는 코드워드를 생성하였지만, 복호화기에서는 참조영상에 발생한 에러에 의해 변화된 시간적 예측 움직임 벡터로 인해 부호화기와 다르게 예측 움직임 벡터 후보 개수를 감소시켜 부호화기에서 전송한 코드워드 파싱과정에서 오류가 발생하여 복호화기가 동작 불가능한 상태가 될 수 있다. 그렇기 때문에 파싱 오류 문제를 해결하면서 동시에 부호화 되는 정보량을 감소시켜야할 기술이 필요하다.

III. 제안하는 모순 검증을 이용한 움직임 벡터 압축 기술

본 장에서는 제안하는 모순 검증을 이용한 움직임 벡터 압축 기술에 대한 기본 개념 및 문제점을 해결하기 위한 구체적인 압축 기술에 대해서 설명한다.

1. 기본 개념

모순 검증을 이용한 움직임 벡터 부호화 방법의 기본 개념은 임의의 가정을 기반으로 도출된 사실이 그 가정과 동일하지 여부를 검증하여 동일하지 않을 경우, 모순이라고

판단하고 가정된 사실을 부정하여 그 가정이 의미하는 결정을 하지 않도록 하는 것이다^[4]. 상기 개념을 움직임 벡터 부호화 방법에 적용하였을 경우, 임의의 가정은 특정 예측 움직임 벡터가 유효-왜곡 비용 관점에서 최적의 움직임 벡터로 선택이 되었다는 것이고, 도출된 사실은 그 가정을 기반으로 새롭게 선택한 최적의 예측 움직임 벡터이다^[15]. 이 두 가지 최적의 예측 움직임 벡터들을 비교하여 서로 다를 경우 모순이라고 판단하여 후보 예측 움직임 벡터들에서 가정된 예측 움직임 벡터를 후보 집합에서 제외시킨다. 따라서 후보 예측 움직임 벡터의 개수가 감소하게 되어 최적의 예측 움직임 벡터 인덱스에 할당되는 비트량을 감소시켜 전체적으로 부호화 효율을 높일 수 있다. 기존 연구^[15]에서는 에러 발생 가능한 예측 움직임 벡터를 고려치 않고 모든 예측 움직임 벡터 후보들에 대해서 모순 검증을 수행하는 움직임 벡터 부-복호화 방법을 제안하였기 때문에, MVComp과 동일하게 파싱 오류 문제가 발생할 수 있다. 예를 들어, 부호화기에서 정상적으로 모순 검증 방법에 의해 예측 움직임 벡터 후보의 개수를 감소시키더라도, 복호화기에서 에러가 발생한 예측 움직임 벡터에 의해 예측 움직임 벡터 후보 개수를 부호화기와 다르게 감소시킬 수 있다. 그래서 부호화기에서 생성한 인덱스 코드워드의 길이와 복호화기에서 파싱해야하는 인덱스 코드워드의 길이가 달라져 파싱 오류가 발생할 수 있다. 또한 예측 움직임 벡터 인덱스가 에러가 발생한 예측 움직임 벡터를 가리킬 경우,

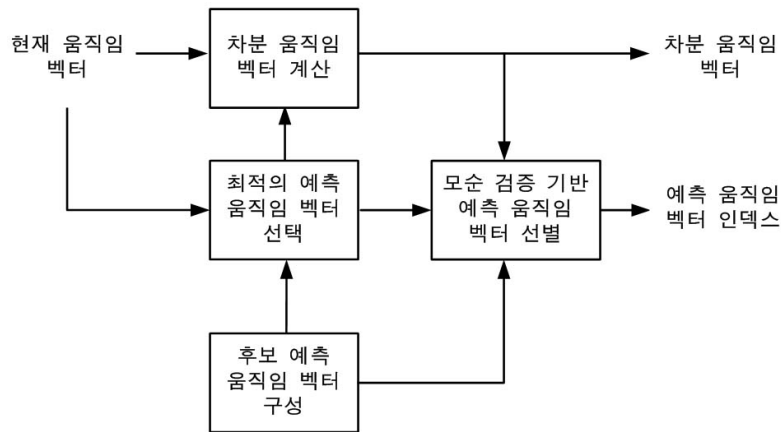


그림. 3. 제안한 움직임 벡터 부-복호화 방법의 부호화 과정
 Fig. 3. Encoding in the proposed MV coding method

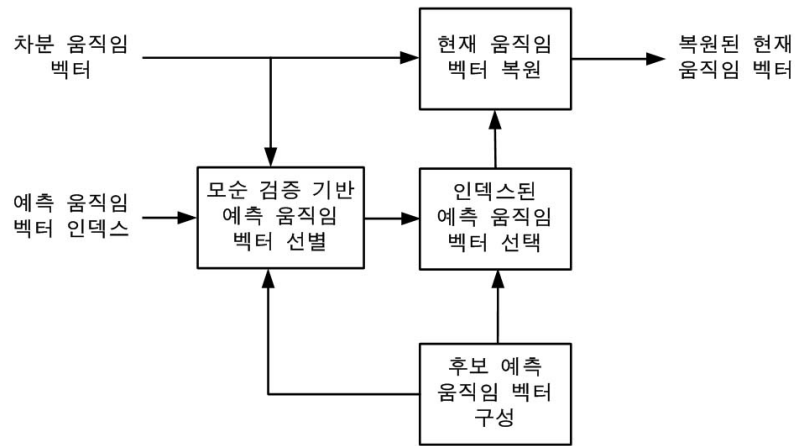


그림 4. 제안한 움직임 벡터 부복호화 방법의 복호화 과정
 Fig. 4. Decoding in the proposed MV coding method

현재 블록의 움직임 벡터가 잘못 복원되어 다음 블록 부-복호화시 상기 잘못된 움직임 벡터가 예측 움직임 벡터 후보에 포함되어 예측 움직임 벡터 후보에 대한 모순 검증 과정이 잘못 수행될 수도 있다. 본 논문에서는 기존 연구와는 다르게 에러 발생 가능한 예측 움직임 벡터, 즉 시간적 예측 움직임 벡터를 제외한 예측 움직임 벡터 후보들에 대해 전술한 기존 연구^[15]와 동일하게 모순 검증 기반 움직임 벡터 부-복호화 하는 개선된 방법을 제안한다.

그림 3과 그림 4는 본 논문에서 제안하는 모순 검증 기반 움직임 벡터의 부-복호화 방법의 전체 블록 다이어그램이다. 그림 3의 수행 과정은 움직임 추정 과정에서 결정한 현재 움직임 벡터를 기준으로 후보 예측 움직임 벡터들 중 최적의 예측 움직임 벡터를 선택하여 차분 움직임 벡터를 계산하여 부호화를 수행한다. 그리고 상기 정보를 바탕으로 모순 검증 기술을 사용하여 불필요한 예측 움직임 벡터를 후보 예측 움직임 벡터들에서 제외시켜 예측 움직임 벡터를 선별한다. 마지막으로 최적의 예측 움직임 벡터 인덱스에 할당되는 비트량을 감소시켜 부호화를 수행한다. 그림 4의 수행 과정은 판독한 차분 움직임 벡터와 후보 예측 움직임 벡터들을 기반으로 모순 검증 기술을 수행하여 유효한 예측 움직임 벡터를 선별하여 판독할 예측 움직임 벡터 인덱스에 할당되는 비트량을 선별된 예측 움직임 벡터의 개수에 따라 결정하여 복호화를 수행한다. 그리고 인덱

스에 해당하는 예측 움직임 벡터와 차분 움직임 벡터를 사용하여 현재 움직임 벡터를 복원시킨다.

2. 동일 율-왜곡 비용 처리 규칙

제안 방법의 율-왜곡 비용 관점 최적의 예측 움직임 벡터를 선택하는 과정에 있어서 동일한 비용을 소비하는 서로 다른 예측 움직임 벡터가 발생할 수 있다. 그 발생 원인은 엔트로피 부호화된 서로 다른 차분 움직임 벡터가 그림 5와 같이 동일한 비트량을 소비할 수 있기 때문이다. 그렇기 때문에 제안 방법의 성능을 더욱 높이기 위해서는 동일한 비용을 소비하는 서로 다른 예측 움직임 벡터의 선택방법을

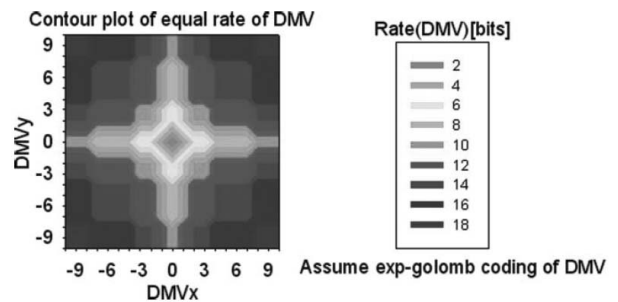


그림 5. H.264/AVC에서 엔트로피 코딩 된 DMV의 동일 비트량 등고선^[15]
 Fig. 5. Contour plot of equal rate of entropy-coded DMV in H.264/AVC^[15]

부-복호화기가 기 정의하여야 한다. 본 논문에서는 동일한 율-왜곡 비용을 소비하는 최적의 예측 움직임 벡터가 한 개 이상일 경우, 후보 예측 움직임 벡터에 순서를 정하여 순서상 가장 앞선 예측 움직임 벡터를 선택하는 간단한 규칙을 적용하여 부-복호화기가 공히 알고 있게 함으로써 제안 방법의 성능을 더욱 상승시켰다.

3. 파싱 문제 극복에 따른 에러 강인성

제안 방법은 기존 연구^[15]와 다르게 에러 발생 가능한 시간적 예측 움직임 벡터를 전체 예측 움직임 벡터 후보에서 제외하고 대신에 공간적 예측 움직임 벡터를 추가한 전체 예측 움직임 벡터 후보에 대해서 율-왜곡 관점에서 최적의 예측 움직임 벡터를 선택하고 그에 대한 인덱스 정보에 대해 모순 검증 기술을 적용하였다. 즉, 시간적 예측 움직임 벡터가 어떠한 값이던 고려하지 않고 항상 예측 움직임 벡터 후보에서 제외시킨 후 대신에 공간적 예측 움직임 벡터를 추가하고 이를 참조하여 부호화기는 코드워드를 생성하고 복호화기는 파싱을 한다. 공간적 예측 움직임 벡터를 추

가하는 순서는 그림 1에서와 같이 $\overline{mv^A}$, $\overline{mv^B}$, 그리고 $\overline{mv^C}$ 순서대로 추가한다. 제안 방법이 파싱 문제를 해결할 수 있는 이유는 예측 움직임 벡터 후보의 개수에 따라 코드워드가 변하는 사실을 바탕으로 시간적 예측 움직임 벡터가 후보 개수에 영향을 줄 수 없도록 하였기 때문이다. 결국 전술한 참조영상에 문제가 발생하여 부-복호화기가 공히 가지고 있는 시간적 예측 움직임 벡터의 정보가 서로 달라져도 예측 움직임 벡터 후보의 개수에는 변화가 없기 때문에 부호화기가 생성한 코드워드를 복호화기는 정상적으로 파싱할 수 있다.

4. 제안방법의 예

표 1은 제안하는 움직임 벡터 복호화 방법의 예로, MV는 현재 움직임 벡터, PMV1, PMV2, PMV4는 공간적 예측 움직임 벡터, PMV3은 에러 발생 가능한 시간적 예측 움직임 벡터, CS(Candidate Set)는 후보 예측 움직임 벡터들을 포함하는 집합, OPMV는 부호화기에서 선택한 최적의 예측 움직임 벡터, DMV는 차분 움직임 벡터라고 가정한다.

표 1. 제안방법의 예
Table 1. An example of the proposed method

MV = (66,-1), CS={PMV1,PMV2,PMV3} where PMV1=(65,0), PMV2=(66,1), PMV3=(66,-1) 그리고 PMV3은 에러 발생 가능한 시간적 예측 움직임 벡터라고 가정한다. 또한 공간적 예측 움직임 벡터 PMV4=(65,1)가 존재한다고 가정한다.						
에러 강인성을 위한 CS 구성 단계	PMV3=(66,-1)을 CS에서 제외하고, PMV4=(65,1)을 CS에 포함시켜 예측 움직임 벡터 후보를 구성함 (CS={PMV1,PMV2,PMV4})					
모순 검증 기반 예측 움직임 벡터 선별 단계 (본 단계는 [15]논문의 내용과 동일)	복호화기가 받은 DMV=(1,-1)		모순 검증			
	복호화기 가정 OPMV is	가정 기반 복원 MV	DMV for a PMV in CS	Rate(DMV) (bits)	복호화기가 선택한 OPMV is	검증 결과
	PMV1=(65,0)	(66,-1)	(1,-1) for PMV1	6	PMV1	정상
			(0,-2) for PMV2	6		
			(1,-2) for PMV4	8		
	PMV2=(66,1)	(67,0)	(2,0) for PMV1	6	PMV1	모순 (PMV2를 CS에서 제외)
			(1,-1) for PMV2	6		
			(2,-1) for PMV4	8		
	PMV4=(65,1)	(66,0)	(1,0) for PMV1	4	PMV1	모순 (PMV4를 CS에서 제외)
			(0,-1) for PMV2	4		
(1,-1) for PMV4			6			
CS={PMV1} 이기 때문에 시그널링을 하지 않음						

표 1을 참조하여 에러 강인성을 위한 CS를 구성시 PMV3은 에러 발생 가능한 시간적 예측 움직임 벡터이기 때문에 CS에서 제외하고 대신에 공간적 예측 움직임 벡터인 PMV4를 CS에 포함시킨다.

우선 PMV1=(65,0)을 OPMV라고 가정하고 OPMV=(65,0)와 판독한 DMV=(1,-1)를 더하여 임시로 MV=(66,-1)을 복원한다. 복원된 MV=(66,-1)를 기준으로 윗-왜곡 관점에서 PMV1=(65,0), PMV4=(65,1) 중 최적의 예측 움직임 벡터 PMV1을 선택한다. 전술한 OPMV가 앞서 윗-왜곡 관점에서 선택한 최적의 예측 움직임 벡터 PMV1과 동일하기 때문에 OPMV=PMV1 가정은 정상이다. 결국 PMV1은 후보 예측 움직임 벡터 집합 내에 유지된다.

다음, PMV2=(66,1)을 OPMV라고 가정하고 OPMV=(66,1)와 판독한 DMV=(1,-1)를 더하여 임시로 MV=(67,0)을 복원한다. 복원된 MV=(67,0)를 기준으로 윗-왜곡 관점에서 PMV1=(65,0), PMV4=(65,1) 중 최적의 예측 움직임 벡터 PMV1을 선택한다. 전술한 OPMV가 앞서 윗-왜곡 관점에서 선택한 최적의 예측 움직임 벡터 PMV1과

다르기 때문에 OPMV=PMV2 가정은 모순이다. 결국 PMV2은 후보 예측 움직임 벡터 집합에서 제외시킨다. 이와 같은 과정으로 PMV4를 OPMV라고 가정하여 전술한 방법과 동일하게 수행하면 OPMV=PMV4 가정 또한 모순이 된다. 결국, 후보 예측 움직임 벡터의 후보는 PMV1만 있기 때문에 예측 움직임 벡터 인덱스를 부호화할 필요가 없다.

IV. 실험 조건 및 결과

본 논문의 제안하는 모순 검증 기반 움직임 벡터 부호화 방법에 대한 성능을 평가하기 위하여 KTA2.6r1 SW를 수정하여 사용하였고 보다 자세한 부호화 조건은 표 2와 같다^[16]. 제안 방법의 부호화 성능과 복호화 복잡도를 평가하기 위해 BDBR(Bjontegaard Delta Bit-Rate), BDPSNR(Bjontegaard Delta PSNR) 그리고 식 (4)와 같이 복호화 시간을 측정하였다^[17].

표 2. 실험 조건
Table 2. Simulation condition

Spatial Resolution	416x240	832x480	1920x1080
Sequence (Frame Rate in Hz)	RaceHorses (30) BlowingBubbles (50) BasketballPass (50) BQSquare (60)	RaceHorses (30) PartyScene (50) BasketballDrill (50) BQMall (60)	ParkScene (24) Kimono1 (24) Cactus (50) BasketballDrive(50) BQTerrace (60)
Number of Encoded Pictures	300		
Motion Estimation	EPZS([18]) Motion Estimation with search range ± 64		
GOP Structure	IPPP		
Quantization Parameter	QPI : 22, 27, 32, 37, QPP: QPI + 1		
Number of Reference Pictures	4		
Profile	High Profile		
Other Coding Options	8x8 transform on, RD on, CABAC on, 1/4-pel ME accuracy		
Performance Comparison	1. Anchor (MV coding in H.264/AVC scheme) 2. MVComp ^[13] 3. Contradiction Testing-based MV Coding ^[15] (Conventional Method) 4. Error Robust Contradiction Testing-based MV Coding (Proposed Method)		
Number of PMV Candidate Set	=2 (Temporal PMV, and Median PMV)		
	=3 (Temporal PMV, Median PMV, and MV in left block)		
	=4 (Temporal PMV, Median PMV, MV in left block, and MV in upper block)		
Implementation	KTA2.6r1 software based on JM11.0 software		

$$DecodingComplexity = 100 \times \frac{DecodingTime_{ComparedMethod}}{DecodingTime_{Anchor}} \quad (4)$$

식 (4)에서 $DecodingTime_{Anchor}$ 는 anchor의 복호화 시간을 의미하고 $DecodingTime_{ComparedMethod}$ 는 비교기술 (MVComp 또는 제안 방법)의 복호화 시간을 의미한다. 여기서 BDBR의 (-) 부호는 비교기술의 성능 개선을 나타내고, (+) 부호는 비교기술의 성능 저하를 나타낸다. 그리고, BDPSNR의 (+) 부호는 비교기술의 성능 개선을 나타내고, (-) 부호는 비교기술의 성능 저하를 나타낸다. 또한 $DecodingComplexity$ 가 100을 기준으로 증가하면 비교기술의 복잡도 증가를 나타내고, 감소하면 비교기술의 복잡도 감소를 나타낸다. 그리고 100이면 anchor와 복잡도가 동일하다는 것을 의미한다.

표 3은 후보 예측 움직임 벡터의 후보 개수와 영상 해상도에 대한 평균 BDBR, BDPSNR, 그리고 복호화기 복잡도를 나타낸다. BDBR관점에서 표 3의 결과를 살펴보면 세 기술 모두 평균적으로 H.264/AVC보다 높은 압축 성능을 보이고 있다. 후보 예측 움직임 벡터의 개수가 2개와 3개일 때는 제안 방법이 MVComp 방법과 기존 연구에 비해 성능이 저하된다. 그 이유는 영상의 특성상 경우에 따라 공간적 예측 움직임 벡터보다 시간적 예측 움직임 벡터가 현재 움직임 벡터와 유사할 수도 있기 때문이다. 그러나, 후보 예측 움직임 벡터의 개수가 4개 이상일 경우 전체적으로

MVComp 방법보다 제안 방법의 부호화 효율이 증가한다. 왜냐하면 후보 예측 움직임 벡터의 개수가 증가할수록 차분 움직임 벡터의 크기를 감소시키는 것보다 예측 움직임 벡터 후보에 대한 인덱스의 크기를 감소시키는 것이 더욱 효과적이기 때문이다. 주변 블록에 포함되어 있는 움직임 벡터를 획득하여 예측 움직임 벡터 후보의 개수를 증가시킨다고 해서 차분 움직임 벡터의 크기가 그에 따라 감소한다고 볼 수 없다. 왜냐하면 주변 블록에 포함되어 있는 움직임 벡터들은 대부분 유사할 가능성이 높기 때문이다. 오히려 예측 움직임 벡터 후보의 개수가 증가할수록 그에 대한 인덱스의 크기는 증가할 수 있다. 결국, MVComp 방법은 추가적으로 시그널링 해야 할 인덱스 크기를 효율적으로 감소시키지 못하지만, 제안방법은 모순 검증 과정을 수행하여 인덱스 크기를 효율적으로 감소시킬 수 있기 때문에 후보 예측 움직임 벡터의 개수가 4개 이상일 경우 전체적으로 MVComp 방법보다 제안 방법의 부호화 효율이 증가하게 된다. 비록 기존 연구^[15]에 비해 압축 효율이 감소하지만 MVComp과 기존 연구^[15]의 큰 문제점인 파싱 오류 문제를 제안 방법은 시간적 예측 움직임 벡터에 대해서 예측 움직임 벡터 후보에서 제외시킨다는 간단한 방법으로 해결하였다.

복호화 복잡도는 세 기술 모두 anchor에 비해 낮은 복잡도를 가진다. anchor의 복호화 복잡도를 100%이라고 가정하였을 경우, MVComp의 복호화 복잡도는 약 89%이고, 기존 연구 및 제안 방법의 복호화 복잡도는 약 90%이다.

표 3. MVComp 방법[13], 기존 방법[15] 그리고 제안 방법에 대한 성능 비교(anchor: H.264/AVC High Profile)
Table 3. Performance comparison of MVComp[13], Conventional[15] and Proposed method (anchor: H.264/AVC high profile)

Coding Efficiency	비교 방법	MVComp ^[13]			기존 연구 ^[15]			제안 방법		
	후보 예측 움직임 벡터 개수	2	3	4	2	3	4	2	3	4
BDBR [%] (BDPSNR [dB])	1920x1080	-3.4 (0.09)	-1.8 (0.05)	-0.6 (0.01)	-3.6 (0.10)	-2.7 (0.07)	-1.8 (0.05)	-1.8 (0.06)	-0.9 (0.03)	-0.8 (0.03)
	832x480	-2.0 (0.09)	-0.7 (0.03)	0.2 (-0.01)	-2.2 (0.09)	-1.5 (0.07)	-1.1 (0.05)	-0.4 (0.02)	-0.1 (0.01)	-0.1 (0.00)
	416x240	-2.2 (0.10)	-0.5 (0.02)	0.2 (-0.01)	-2.5 (0.11)	-1.6 (0.07)	-1.2 (0.06)	-0.2 (0.01)	0.1 (0.00)	0.1 (-0.01)
	Overall	-2.6 (0.09)	-1.1 (0.04)	-0.1 (0.00)	-2.8 (0.10)	-2.0 (0.07)	-1.4 (0.05)	-0.8 (0.03)	-0.3 (0.01)	-0.2 (0.01)
Decoding Complexity [%] (anchor의 decoding complexity를 100이라고 가정)		89			90			90		

MVComp, 기존 연구 그리고 제안 방법이 낮은 복잡도를 달성할 수 있는 이유는 SKIP 모드 개수가 증가하여 역양자화 및 역변환을 수행하지 않기 때문이다. 구체적으로, H.264/AVC에서 SKIP 모드가 결정되는 조건은 첫째 잔차 신호가 0이고, 둘째 참조영상이 현재영상과 가장 가까워야 하고, 셋째 차분 움직임 벡터가 0이어야 한다. 이때, 첫 번째, 두 번째 조건은 만족하지만 세 번째 조건을 만족하지 못해서 SKIP 모드가 되지 않는 경우가 발생할 수 있다. 이 경우 MVComp, 기존 연구 그리고 제안 방법을 사용하면 차분 움직임 벡터를 더욱 0에 가깝게 하는 최적의 예측 움직임 벡터를 선택할 수 있기 때문에, 세 번째 조건을 만족시킬 수 있어 SKIP 모드의 개수가 증가하게 되고, 그에 따른 역양자화, 역변환, 복원과정이 생략이 되기 때문에 복호화 복잡도가 감소할 수 있다.

V. 결론

본 논문은 MVComp 방법에서 발생할 수 있는 파싱 오류 문제를 해결할 수 있는 모순 검증 기반 움직임 벡터 부-복호화 방법을 제안하였다. 제안 방법은 오류 발생 가능성이 있는 시간적 예측 움직임 벡터를 제외한 후보 예측 움직임 벡터들 중 특정 예측 움직임 벡터가 유효-왜곡 관점에서 최적의 예측 움직임 벡터로 선택되었다고 가정하고, 이미 판독한 차분 움직임 벡터와 가정된 최적의 예측 움직임 벡터를 더하여 임시로 현재 움직임 벡터를 생성한다. 그리고 부호화기가 현재 움직임 벡터를 사용하여 유효-왜곡 관점에서 최적의 예측 움직임 벡터를 선택하기 위해 동작하는 과정과 동일하게 부호화기가 임시로 생성한 현재 움직임 벡터를 사용하여 최적의 예측 움직임 벡터를 선택한다. 앞서 최적의 예측 움직임 벡터라고 가정된 특정 예측 움직임 벡터와 부호화기에서 선택한 예측 움직임 벡터가 다를 경우 모순이라고 결정하여 후보 예측 움직임 벡터에서 제외를 한다. 결국, 후보 예측 움직임 벡터의 개수가 감소하여 선택된 최적의 예측 움직임 벡터의 인덱스에 소모되는 비트량을 감소시켜 부호화 효율을 높일 수 있을 뿐만 아니라 후보 예측 움직임 벡터내 오류 발생 가능성 있는 예측 움직임 벡터를

예측 움직임 벡터 후보에서 제외시켜 파싱 오류 문제를 해결하였다. 제안하는 모순 검증 기반 움직임 벡터 부호화 방법은 H.264/AVC 방법과 비교하여 최대 1.8%의 비트량을 절감시킨다.

참고 문헌

- [1] JVT of ISO/IEC MPEG and ITU-T VCEG, Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification, ITU-T Rec. H.264, ISO/IEC 14496-10 AVC, 2003.
- [2] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," IEEE Trans. on Circuits and Syst. for Video Tech., vol.13, no.7, pp.560-576, Jul. 2003.
- [3] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: tools, performance, and complexity," IEEE Circuits Syst. Mag., vol. 4, no. 1, pp. 7 - 28, 1Q, 2004.
- [4] T. Wedi, "1/8-pel Motion Vector Resolution for H.26L," ITU-T VCEG, Proposal Q15-K-21, Aug. 2000.
- [5] L. A. Da Silva Cruz, and J. W. Woods, "Adaptive motion vector quantization for video coding," in Proc. IEEE ICIP, Oct. 2000, vol. 2, pp. 867 - 870.
- [6] A. Deever and S. S. Hemami, "Dense motion field reduction for motion estimation," Signals, Syst. Comput., vol. 2, pp. 944 - 948, Nov. 1998.
- [7] M. C. Chen and A. N. Willson, "A spatial and temporal motion vector coding algorithm for low-bit-rate video coding," Proc. IEEE Inter. Conf. Image Processing (ICIP), Oct. 1997, vol. 2, pp. 791 - 794.
- [8] 원광현, 양정엽, 전병우, "복호화기 측의 예측을 이용한 움직임 벡터 부호화", 2008 한국방송공학회 학술대회, pp.131~134, 2008년 11월 8일.
- [9] 원광현, 정봉수, 전병우, "B 영상에서의 직접 예측 부호화 방법의 개선", IPIU 제20회 영상처리 및 이해에 관한 워크샵, pp.4, 2008. 2월
- [10] 정봉수, 원광현, 전병우, "B 슬라이스의 압축 효율 향상을 위한 개선된 양방향 대칭 예측 부호화 방법", 방송공학회논문지, 제14권, 제1호, pp.59~69, 2009년 1월.
- [11] S. D. Kim and J. B. Ra, "An efficient motion vector coding scheme based on minimum bit rate prediction," IEEE Trans. Image Process., vol. 8, no. 8, pp. 1117 - 1120, 1999.
- [12] J. Yang, K. Won, and B. Jeon, "Motion vector coding with selection of an optimal predictive motion vector" SPIE Opt. Eng., vol. 48, no.1, pp.010501, Jan. 2009.
- [13] G. Laroche, J. Jung, and B. Pesquet-Popescu, "RD Optimized Coding for Motion Vector Predictor Selection," IEEE Trans. on Circuits and Syst. for Video Tech., vol.18, no.12, pp.1681-1691, Dec. 2008.
- [14] W. Byers, How Mathematicians Think: Using Ambiguity, Contradiction, and Paradox to Create Mathematics, Princeton University Press, NJ (2007).
- [15] K. Won, J. Yang and B. Jeon, "Motion vector coding with contradiction testing", SPIE Optical Engineering, vol. 50, no. 4, pp.

040502-1~040502-3, Apr. 2011.

[16] Key Technology Area Reference Software version 2.6r1 [online], Available: <http://iphome.hhi.de/suehring/tml/download/KTA>, Nov. 2009

[17] G. Bjøntegaard, "Calculation of average PSNR differences between

RD-curves," Doc. of ITU-T SG16/Q6 VCEG, VCEG-M13, Apr. 2001.

[18] A.M.Tourapis, "Enhanced Predictive zonal Search for Single and Multiple Frame Motion Estimation," in proceedings of Visual Communications and Image Processing 2002 (VCIP-2002), pp.1069-79, San Jose, CA, Jan. 2002.

저 자 소 개



원 광 현

- 2007년 : 광운대학교 컴퓨터공학부 학사 졸업
- 2009년 : 성균관대학교 휴대폰학과 석사 졸업
- 2009년 ~ 현재 : 성균관대학교 휴대폰학과 박사 과정
- 주관심분야 : 멀티미디어 영상압축



양 정 업

- 2004년 : 성균관대학교 정보통신공학부 학사 졸업
- 2006년 : 성균관대학교 전자전기공학과 석사 졸업
- 2012년 : 성균관대학교 전자전기공학과 박사 졸업
- 2012년 ~ 현재 : 삼성전자 반도체총괄 System LSI 사업부
- 주관심분야 : 멀티미디어 영상압축



전 병 우

- 1985년 : 서울대학교 전자공학과 학사 졸업
- 1987년 : 서울대학교 전자공학과 석사 졸업
- 1992년 : Purdue Univ, School of Elec. 공학박사 졸업
- 1993년 ~ 1997년 : 삼성전자 신호처리연구소 선임/수석연구원
- 1997년 ~ 현재 : 성균관대학교 정보통신공학부 교수
- 주관심분야 : 멀티미디어 영상압축, 영상인식, 신호처리