

# 효율적 경쟁 기반 스킵모드 부호화를 위한 적응적 문맥 기반 움직임 예측 후보 선택 기법

정회원 김 성 제\*, 김 용 구\*\*°, 종신회원 최 윤 식\*

## An Efficient Competition-based Skip Motion Vector Coding Scheme Based on the Context-based Adaptive Choice of Motion Vector Predictors

Sungjei Kim\*, Yong-Goo Kim\*\*° *Regular Members*, Yoonsik Choe\* *Lifelong Member*

### 요 약

지속적인 네트워크 전송 능력 및 저장 매체 성능 향상에도 불구하고, 다양한 기술적·경제적 이유에서, 화질 열화를 최소화하면서 압축 성능을 증대시키기 위한 비디오 부호화 기술이 꾸준히 개발되고 있다. 본 논문에서는 종래의 비디오 성능을 더욱 향상시키기 위해, 경쟁 기반 스킵모드 부호화를 위한 적응적 문맥기반 움직임 예측 후보 선택 기법을 제안한다. 제안하는 기법은 카메라의 움직임 여부와 주변 블록의 움직임 벡터 문맥을 기반으로 움직임 예측 후보를 적응적으로 구성하여, 고정된 움직임 예측 후보를 가지는 종래의 방식에 비해 효율적인 움직임 예측 후보를 제공하였다. 이와 같이 선택된 적응적 움직임 예측 후보를, 비디오 부호기의 인터 모드 중 가장 효율적인 부호화 모드로 알려진 스킵 모드의 움직임 예측 후보 선택에 적용함으로써 종래 방식 중 가장 우수한 성능을 나타내는 경쟁 기반 움직임 벡터 부호화 방식의 BD-RATE 성능을 최대 2배까지 개선할 수 있었다.

**Key Words** : Motion Vector Coding, Skip Competition, Motion Predictors, Context-adaptive, Predictor Selection

### ABSTRACT

The demand for high quality of multimedia applications, which far surpasses the rapid evolution of transmission and storage technologies, makes better compression coding capabilities ever increasingly more important. In order to provide enhanced video coding performance, this paper proposes an efficient competition-based motion vector coding scheme. The proposed algorithm adaptively forms the motion vector predictors based on the contexts of scene characteristics such as camera motion and nearby motion vectors, providing more efficient candidate predictors than the previous competition-based motion vector coding schemes which resort to the fixed candidates optimized by extensive simulations. Up to 200% of compression gain was observed in the experimental results for the proposed scheme applied to the motion vector selection for skip mode processing.

### I. 서 론

지속적인 네트워크 전송 능력 및 저장 매체 성능

향상에도 불구하고, 다양한 기술적·경제적 이유에서, 화질 열화를 최소화하면서 압축 성능을 증대시키기 위한 여러 비디오 부호화 기술들이 꾸준히 개발되고

※ 본 연구는 한국 기술 표준원의 지원 사업 B0011443의 지원을 통해 수행되었습니다.

\* 연세대학교 전기전자공학과 영상정보연구실(coldeyes@yonsei.ac.kr),

\*\* 한독미디어대학원대학교 뉴미디어학부 미디어통신연구실(ygkim@kgit.ac.kr), (° : 교신저자)

논문번호 : KICS2010-03-102, 접수일자 : 2010년 4월 26일, 최종논문접수일자 : 2010년 4월 28일

있다. 이들 중 움직임 벡터 및 움직임 복원을 위한 부호화 파라미터의 비트량 감소를 위한 연구는 특히 중요한 의미를 가지는데, 이는 낮은 비트율 환경에서 움직임 벡터가 차지하는 데이터의 비율이, 현재 대표적인 비디오 코덱인 H.264/AVC<sup>[1]</sup>의 경우, 전체 비트량의 50%에까지 이르고 있을 뿐 아니라, 압축 부호화 성능을 더욱 개선할 목적의 차세대 비디오 압축 부호화 표준의 연구<sup>[2]</sup>에서는 보다 정확한 움직임 모델의 도입을 통해 이 비율이 더욱 증가될 것으로 예상되기 때문이다.

이와 같은 이유로 다양한 움직임 벡터 부호화 기술들이 제안되었는데<sup>[1,3,4,5]</sup>, 이들 중 H.263<sup>[3]</sup> 또는 H.264/AVC<sup>[1]</sup>와 같은 기존의 블록 기반 비디오 압축 국제 표준에서는, 한 블록의 움직임 벡터가 그 주변 블록의 움직임 벡터와 높은 상관도를 가지고 있는 점을 이용하여, 각 블록의 움직임 벡터를 부호화할 때, 그 주변 블록의 움직임 벡터 중앙값(median)을 현재 부호화 할 움직임 벡터의 예측 치로 생성함으로써, 부호화 할 움직임 벡터에서 예측치를 뺀 잔여 움직임 벡터만을 엔트로피 부호화하여 높은 성능 개선을 이루었다. 하지만, 이러한 중앙값 예측치는 현재 블록의 움직임 벡터가 주변 움직임 벡터와 유사한 패턴을 갖는 경우에는 부호화해야 하는 잔여 움직임 벡터 값이 크게 줄어들어 그 성능이 우수한 반면에, 움직임이 다른 영상 내 물체의 윤곽선 부분을 포함한 블록과 같이, 주변 움직임과 다른 불규칙한 패턴을 갖는 경우에는 그 성능이 제한적이다.

이와 같은 불규칙한 패턴을 갖는 블록에서의 성능 제한 문제를 해결하기 위해, S. Kim<sup>[4]</sup> 등은 중앙값 예측치 대신 세 개의 주변 블록 움직임 벡터들 중에 현재 블록의 움직임 벡터와 가장 유사한 움직임 예측 후보를 선택하여 이를 예측 치로 사용함으로써 잔여 움직임 벡터 값을 효율적으로 줄일 수 있는 경쟁 기반의 움직임 벡터 코딩 기법을 제안하였다. 하지만, 비교적 규칙적인 패턴을 많이 포함하는 영상에 대해서는 선택된 최적의 움직임 예측 후보를 인덱싱하는데 필요한 오버헤드 비트가 잔여 움직임 벡터 값을 줄여 얻을 수 있는 성능 개선 량에 비해 커지게 되어 전체적인 압축 성능이 오히려 떨어질 수 있다는 한계를 갖는다.

J. Jung<sup>[5]</sup> 등은, 이와 같은 한계를 해결하기 위해, 예측 후보를 두 개로 고정시켜 부가적인 인덱싱 비트의 증가를 최소화하고, 스킵 모드의 최적 움직임 예측 후보를 윗-왜곡 관점에서 경쟁적으로 선택하는 기법을 포함하는 향상된 경쟁 기반의 움직임 벡터 코딩 기법을 제안함으로써 움직임 벡터 부호화 효율을 크게

향상시켰다. 하지만, 이 기법은 실험적으로 선택한 두 개의 고정된 예측 후보를 사용하기 때문에, 카메라의 움직임이나 영상 내 물체의 배치, 크기, 움직임 등에 의한 영상의 특성에 따라 그 성능 개선의 정도가 매우 제한적인 경우들이 발생할 수 있다는 문제를 가진다.

본 논문에서는, 이러한 경쟁 기반 움직임 벡터 코딩 기술의 한계를 극복하고 더욱 높은 성능 개선을 이루기 위해, 영상의 특성을 반영하여 적응적으로 효율적인 움직임 예측 후보를 선택하는 기법을 제안한다. 이러한 적응적 움직임 예측 후보의 선택을 위해, 본 논문에서는, 움직임 예측 후보로 사용할 수 있는 높은 움직임 벡터 상관도를 가지는 모든 주변 움직임 벡터들을 중앙값 계열, 그리고 기타 계열의 두 집합으로 분류하고, 카메라의 움직임 여부와 주변 블록의 움직임 벡터 문맥을 기반으로 각 집합에서 하나씩의 움직임 예측 후보를 선택하도록 하였다. 이와 같이 선택된 적응적 움직임 예측 후보를, 비디오 부호기의 인터 모드 중 가장 효율적인 부호화 모드로 알려진 스킵 모드의 움직임 예측 후보 선택에 적용함으로써 종래 방식 중 가장 우수한 성능을 나타내는 경쟁 기반 움직임 벡터 부호화 방식<sup>[5]</sup>의 BD-RATE 성능을 최대 2배까지 개선할 수 있었다.

## II. 경쟁기반 스킵 모드 움직임 벡터 선택 기법

2003년 H.264/AVC의 표준화 이후, ITU-T VCEG(Video Coding Expert Group)에서는 지속적인 연구를 통해 H.264/AVC 표준의 부호화 성능을 개선하여, 제안된 다양한 기법들을 KTA-software (Key Technology Area)에 통합해왔다. 이러한 기법들 중, 움직임 벡터 코딩 부분에서는 그 성능을 인정받아, KTA-software에 채택된 경쟁 기반 스킵 모드 움직임 벡터 선택 기법<sup>[5]</sup>이 있다. 이 기법은 경쟁 기반 움직임 벡터 코딩의 움직임 예측 후보군을 정하여 경쟁 기반 스킵 모드 움직임 벡터를 선택하는 기법인데, 그 구체적인 방법은 다음과 같다.

### 2.1 경쟁기반 움직임 벡터 코딩의 움직임 예측 후보

일반적으로 움직임 예측 후보는 공간적, 시간적 예측 후보들로 나뉠 수 있는데, 그림 1에서 현재 프레임인 N번째 프레임 내에서 존재하는 움직임 예측 후보를 공간적 움직임 예측 후보라고 하고, 이전 프레임인 N-1번째 프레임 내에 존재하는 움직임 예측 후보를 시간적 움직임 예측 후보라고 한다.

공간적 움직임 예측 후보로 대표적인 움직임 벡터는 그림 1에서와 같이  $mv_a, mv_b, mv_c, mv_d, mv_0$ 와  $mv_{264}, mv_{ext}$ 로 구성될 수 있다.  $mv_a, mv_b, mv_c, mv_d$ 는 현재 블록의 주변에 위치한 이미 부호화 된 블록들이 가지는 움직임 벡터를 의미하고,  $mv_0$ 는 (0,0) 값을 갖는 움직임 벡터이다. 또한,  $mv_{264}$ 는 일반적으로 H.264/AVC에서 사용하는 중앙값( $mv_{med}$ )을 의미하며,  $mv_{med} = median\{mv_a, mv_b, mv_c\}$ 로 구하게 된다. 만일  $mv_c$ 가 존재하지 않는 경우는  $mv_d$ 을 이용하여 계산하며, 스킵의 경우  $a$  또는  $b$  블록이 존재하지 않거나 인트라 모드로 코딩되었을 때,  $mv_{264}$ 는  $mv_0$  값을 갖는다. 마지막으로  $mv_{ext}$ 은 확장된 공간 예측 벡터로  $a, b, c$  블록이 모두 존재하는 경우 중앙값으로,  $a$  블록이 존재하고  $b$  또는  $c$  블록이 존재하지 않는 경우는  $mv_a$ 로 결정하고,  $b$  블록이 존재하고  $a$  또는  $c$  블록이 존재하지 않는 경우는  $mv_b$ ,  $c$  블록이 존재하고  $a$  또는  $b$  블록이 존재하지 않는 경우는  $mv_c$ , 모두 존재하지 않는 경우는  $mv_0$  값을 갖게 된다. 이  $mv_{ext}$ 는  $mv_{264}$ 와 달리  $a$  또는  $b$  블록이 인트라 모드로 코딩되었을 때도 중앙값 예측을 사용하기 때문에 텍스처가 복잡하고 카메라의 움직임이 전반적으로 일어나는 영상들에 대해서 우수한 성능을 보이지만, 카메라가 고정되어 있는 영상에 대해서는  $mv_{264}$ 에 비해 예측 성능이 제한적일 수 있는 특징을 갖는다.

다음으로, 대표적인 시간적 움직임 예측 후보는  $mv, mv_{ti}(i = 1, \dots, 7)$ 로, N-1번째 프레임의 부호화 할 현재 블록과 같은 위치에 있는 블록 및 그 주변 블록들이 가지는 움직임 벡터를 이용한다.  $mv_{col}$ 는 현재 블록과 동일한 위치의 이전 프레임 블록의 움직임

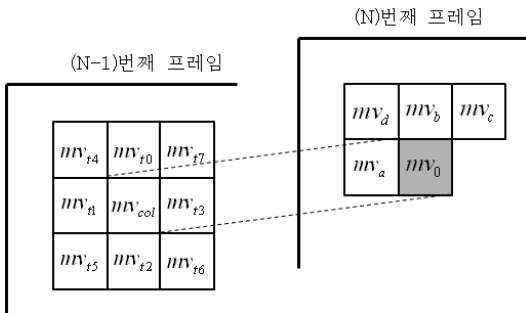


그림 1. 다양한 시공간적 움직임 예측 후보들  
Fig. 1. Various spatial-temporal motion vector candidates

벡터를 의미하며,  $mv_{ti}$ 는 그 주위 블록의 움직임 벡터를 의미한다. 하지만, 공간적 움직임 예측 후보와 달리 시간적 움직임 예측 후보는 이전에 복호된 프레임의 움직임 정보를 이용하기 때문에, 이전 프레임의 전송 에러가 현재 프레임의 비트열 복호과정에 영향을 미치게 되어 디코더 충돌 문제(Decoder Crash Problem)를 야기할 수 있으므로, 추가적 성능개선이 가능함에도 불구하고 본 논문에서는 시간적 움직임 예측 후보의 사용을 배제하였다.

### 2.2 경쟁기반 스킵모드 움직임 벡터 선택 방식

H.264/AVC에서 스킵 모드에 관해서는 울-왜곡 비용  $J_{skip}$ 은 다음 식 (1)과 같이 정의된다.

$$J_{skip} = D_{skip} + \lambda R_{skip} \quad (1)$$

여기서  $D_{skip}$ 는 현재 매크로블록과  $mv_{264}$ 의 위치에 해당하는 이전 프레임의 매크로블록과의 SSE (Sum of Squared Error) 값이며, 스킵의 경우는 움직임 예측 값을 스킵 모드의 움직임 벡터로 사용하기 때문에  $R_{skip}$ 는 스킵 모드를 인덱싱하기 위해 필요한 비트만을 포함한다. 또한,  $\lambda$ 는 양자화 파라미터에 따라 변하는 왜곡과 비트율 사이의 관계를 적절히 반영하기 위한 라그랑지 승수이다<sup>6)</sup>.

경쟁 기반 움직임 벡터 코딩 기법<sup>5)</sup>은 여러 개의 스킵 움직임 예측 후보들 간의 최적 후보 선택을 통해 H.264/AVC의 스킵 결정 기법을 보다 개선하였다. 다음 식 (2)는 경쟁 기반 움직임 벡터 코딩 기법에서 스킵 모드를 결정하기 위한 수정된 울-왜곡 판정식이다.

$$J_{skip}^{best} = D_{skip}^j + \lambda(R_{skip} + R_{mv_j}) \quad (2)$$

$$\{j = 1(mv_{ext}), 2(mv_a)\}$$

여기서  $D_{skip}^j$ 는  $mv_j$ 의 위치에 해당하는 이전 프레임의 매크로블록과의 SSE 값이며,  $R_{mv_j}$ 는  $mv_j$ 를 인덱싱하기 위해 필요한 비트량이다. 또한  $R_{mv_j}$ 는  $j$ 의 최대값이 2이기 때문에 고정 길이 부호화를 할 때,  $\log_2 \lceil 2 \rceil = 1$  비트가 된다. 만일  $mv_{ext}$ 와  $mv_a$ 가 같은 값을 가지면, 디코더에서도 해석 가능하기 때문에  $R_{mv_j}$ 를 전송하지 않아도 된다. 하지만, 식 (3)과 같이 경쟁 기반 움직임 벡터 코딩 기법에서 최적으로

골라진  $J_{skip}^{best}$  가 식 (1)과 같이 원래 H.264/AVC에서 계산하는  $J_{skip}$  보다 커지는 경우가 발생하면, 오히려 H.264/AVC의 경우보다 성능이 나빠질 수 있다.

$$J_{skip}^{best} < J_{skip} \quad (3)$$

이는 경쟁적 선택 과정에 제공된 두 개의 움직임 예측 후보 ( $mv_{ext}, mv_a$ )에 의해서 얻어지는  $D_{skip}^j$  가 H.264/AVC에서 제공하는  $mv_{264}$ 에 의한  $D_{skip}$  비해서 큰 이득을 내지 못하거나, 오히려 증가하기 때문이다. 실제로 주변 움직임 벡터는 카메라의 움직임이나 객체(Object)의 움직임에 따라서 현재 블록과의 상관도가 달라지기 때문에, 두 개의 움직임 예측 후보를 영상의 특성을 고려하여 적절히 제공할 수 있어야만 기존의 경쟁적 선택 기법이 가지고 있는 한계를 극복하고, 나아가 코딩 성능의 향상을 가져올 수 있게 된다.

### III. 제안된 스킵 모드 움직임 예측 후보 선택 기법

2장에서 언급한 기존의 경쟁 기반 움직임 벡터 코딩 기법이 가진 한계점을 해결하기 위해, 본 장에서는 움직임 예측 후보로 사용할 수 있는 기존의 모든 움직임 벡터들을 그 특성에 따라 중앙값 계열 및 기타 계열의 두 집합으로 분류하고, 선택된 두 개의 후보 집합에서 각각 적합한 후보를 하나씩 선택하는 적응적 문맥 기반 움직임 예측 후보 선택 기법을 제안한다.

#### 3.1 체계적인 움직임 예측 후보 집합 분류 방법

일반적으로 영상 내의 움직임은 카메라에 의해서 발생하는 전역 움직임과 객체의 움직임에서 발생하는 국소적인 움직임으로 구분할 수 있다. 주로 전역 움직임에 영향을 받는 블록은 현재 블록의 움직임 벡터가 주변 블록의 움직임 벡터와 유사한 값을 갖기 때문에 중앙 예측 값으로 모델링하여 해결할 수 있다. 하지만, 객체의 움직임에서 발생하는 국소적인 움직임은 현재 블록의 움직임 벡터와 주변 블록의 움직임 벡터 간의 상관도에 따라서 다르기 때문에 주변 블록의 상황에 따라서 다른 움직임 예측 후보를 선택해야 한다. 따라서 제안하는 스킵 모드 움직임 예측 후보 선택 기법은 앞서 2장에서 언급한 움직임 예측 후보군을 다음 그림 2와 같이 분류해서 카메라의 전역 움직임을 효율적으로 해결할 수 있는 움직임 벡터 집합 A와 객체

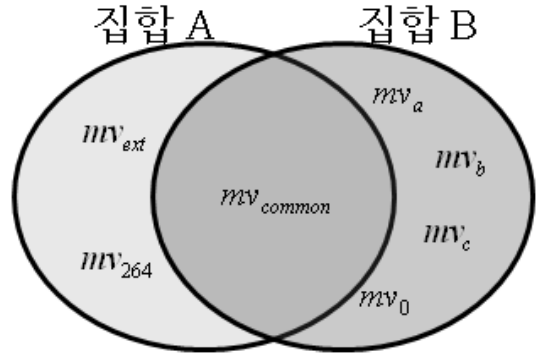


그림 2. 움직임 벡터 후보군의 분류  
Fig. 2. Candidate motion vector sets and classifications

의 국소적인 움직임을 효율적으로 해결할 수 있는 움직임 벡터 집합 B로 구분한다. (단, 그림 2의 집합 B에서 시간적 예측 후보는 디코더 충돌 때문에 제외하였다.) 만일 부분적으로 동일한 객체에 속해 있는 블록이거나 배경과 같이 주변의 블록과 유사한 움직임 패턴을 갖는 경우, 집합 A와 B 간의 후보 움직임 벡터들은 서로 동일한 값을 가질 수 있는데, 이 부분은 그림 2에서 A와 B의 교집합 부분( $A \cap B, mv_{common}$ )에 해당한다. 각 집합에서 대표 움직임 벡터는 카메라의 움직임이나 주변 블록의 특성을 고려해서 적응적으로 선택되는데, 여기서 두 개의 선택된 대표 움직임 벡터가 경쟁 기반 움직임 코딩 기법에서의 후보 움직임 예측 벡터가 된다.

집합 A에는 영상의 전역 움직임을 해결하기 위해 중앙값 계열 움직임 벡터인  $mv_{264}, mv_{ext}$ 가 포함되고, 이 두 개의 움직임 벡터는 카메라의 움직임 변화를 고려하여 적응적으로 선택된다. 카메라가 움직이지 않는 경우는 움직임 벡터가 주로  $mv_0$  또는  $mv_{med}$ 을 갖게 된다. 반면에 카메라가 움직이는 경우는 주변 블록이 복잡한 텍스처를 갖게 되어 인트라 블록으로 코딩이 되더라도 주변 블록의 움직임 벡터의 평균적인 움직임( $mv_{med}$ )을 반영해야 한다.  $mv_{264}$ 와  $mv_{ext}$ 는 2장에서 언급한 것과 같이 주변 블록의 상태에 따라서 각각  $mv_0$ 와  $mv_{med}$ 을 적응적으로 가질 수 있기 때문에 현재 블록의 이전에 부호화된 블록들의 움직임 벡터가 가지는  $mv_0$ 값의 비율이 낮을 경우 (전역 움직임이 있는 것으로 판단되는 경우) 집합 A에서의 대표 움직임 후보 벡터로 선택된다. 집합 A의 대표 움직임 후보 벡터 선택을 위한 판별식은 다음 (4)와 같다.

$$\begin{cases} mv_{264}, p(mv_0) \geq TH \\ mv_{ext}, p(mv_0) < TH \end{cases} \quad (4)$$

여기서  $p(mv_0)$ 는 현재 블록 이전에 부/복호된 행 (row)의 움직임 벡터가 갖는  $mv_0$ 의 비율을 의미한다. 제안하는 기법에서는  $TH$ 는 실험적으로 0.6으로 구하였다. 즉, 카메라의 움직임이 정지된 것으로 판단되는 경우 ( $p(mv_0) \geq 0.6$ )는  $mv_0$ 와  $mv_{med}$ 를 포함하는  $mv_{264}$ 를 대표 후보 벡터로 선택하고, 카메라의 움직임이 발생한 것으로 판단되는 경우 ( $p(mv_0) < 0.6$ )는 주변 블록이 복잡한 텍스처를 갖게 되어 인트라 블록으로 코딩되더라도  $mv_0$  대신에 전역 움직임을 반영할 수 있는  $mv_{med}$ 으로 결정되는  $mv_{ext}$ 를 선택한다.

하지만, 현재 블록이 객체의 경계에 위치하여 주변 블록의 움직임 벡터와는 다른 불규칙한 움직임을 갖게 되는 경우는 집합 A의 대표 벡터로는 적절한 예측치를 설정할 수 없기 때문에, 그림 2의 집합 B와 같이 불규칙한 움직임을 해결할 수 있는 후보 움직임 벡터들을 주변의 블록의 상황에 맞게 적응적으로 선택해서 해결해야 한다. 따라서 본 논문에서는 이러한 불규칙한 움직임을 해결하기 위하여 적절한 움직임 벡터 후보를 선택하는 적응적 문맥 기반 움직임 예측 후보 선택 기법을 제안한다.

### 3.2 적응적 문맥 기반 움직임 예측 후보 선택 기법

제안하는 기법은, 현재 블록이 객체의 경계에 위치하여 전역 움직임으로만 움직임 벡터 부호화를 위한 적절한 예측치를 발견할 수 없는 경우, 이러한 문제를 해결할 수 있는 움직임 예측 후보를 선택하는 기법이다. 이 선택 기법은 주변 블록의 움직임 벡터 간의 거리 관계를 통해 적합한 후보 움직임 벡터를 선택하는 문맥 모델링 기법으로, 그림 2의 집합 B와 같이 불규칙한 움직임을 효율적으로 해결할 수 있는 후보 집합에서 현재 블록의 주변 상황에 적합한 후보 움직임 벡터를 선택하도록 하는 기법이다.

문맥 모델링은 그림 3과 같이 현재 블록 ( $x$ )의 주변 블록 ( $a, b, c$ )의 움직임 벡터와 그 주변 블록 (1, 2, 3, 0)의 움직임 벡터 간의 거리 관계에 의해서 이루어진다. 현재 블록의 주변 블록 ( $n$ )의 움직임 벡터와 가까운 움직임 벡터를 (1), (2), (3), (0) 블록 ((0) 블록은  $mv_0$ 를 갖는 자신 블록을 의미한다.)의 움직임 벡터 중에서 다음 식 (5)와 같이 움직임 벡터

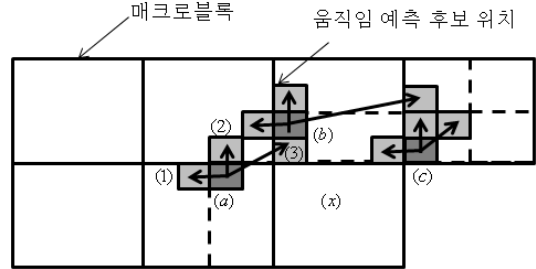


그림 3. 주변 블록의 움직임 벡터의 거리 관계  
Fig. 3. Relations between neighboring motion vectors

간의 거리가 최소가 되는 블록  $M$ 을 찾는다.

$$M^n = \arg \min_r \{ |mv_x^r - mv_x^n| + |mv_y^r - mv_y^n| \} \quad (5)$$

for  $\{r = 1(a), 2(b), 3(c), 4(0)\}$

여기서  $(mv_x^n, mv_y^n)$ 은 주변 블록 ( $n$ )의 움직임 벡터이고,  $(mv_x^r, mv_y^r)$ 은 주변 블록 ( $n$ )의 주변 블록들 ( $r$ )의 움직임 벡터이다.  $M^n$ 은 수식 (5)를 통해 얻을 수 있는 임의의 주변 블록 ( $n$ )에 대한 최적 모드이고,  $M^n$ 은  $\{n = 1(a), 2(b), 3(c)\}$ 에 대해서 구하게 된다. 따라서 현재 블록  $x$ 의 문맥은 주변 블록의 가장 가까운 모드들의 조합 ( $M^1, M^2, M^3$ )으로 구성된다. 예를 들면, ( $M^1, M^2, M^3$ )가 (1, 3, 2)의 값을 가지면, 이 문맥은 현재 블록  $x$ 의 왼쪽 블록인 ( $a$ ) 블록의 경우, 그 왼쪽 블록인 (1)과 움직임 벡터 관점에서 가장 가까운 값을 가지고, ( $b$ ) 블록의 경우는 오른쪽 상단에 위치한 블록인 (3)과, ( $c$ ) 블록의 경우는 위쪽에 위치한 (2)의 블록과 가장 가깝다는 것을 의미한다.

불규칙한 움직임을 해결하기 위한 보다 정확한 문맥 ( $M^1, M^2, M^3$ )을 모델링하기 위해서, 그림 2의 집합 A의 원소로 충분히 해결할 수 있는 영역 ( $A \cap B, mv_{common}$ )은 제외하고 집합 B와 공통부분이 아닌 ( $B - A$ )의 영역을 따로 분류하여 모델링을 수행하였다. 따라서 주변 블록의 움직임 벡터와 그 이전의 부호화 또는 복호화 된 블록의 움직임 벡터와의 상관도를 통해 얻어진 문맥은 현재 블록 주위의 국소 움직임 패턴을 반영할 수 있기 때문에 현재 블록의 불규칙 움직임에 적절한 후보 벡터를 제공할 수 있게 되고, 이는 부호화 효율의 개선으로 이어질 수 있게 된다.

이상의 설명을 바탕으로, 최종 제안 기법을 정리하면 그림 4와 같다. 영상의 전역 움직임을 효율적으로

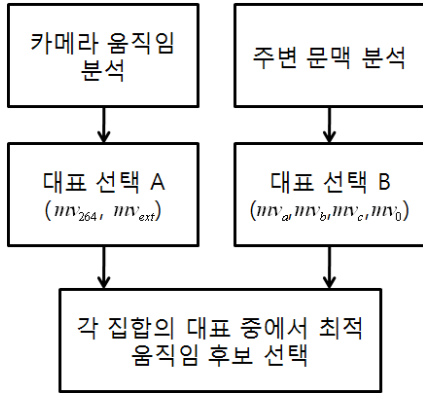


그림 4. 제안하는 움직임 후보 선택 방식의 블록도  
Fig. 4. Overall block diagram of the proposed algorithm

해결하기 위해서 카메라의 움직임을 분석하고, 객체에 의한 국소적인 움직임을 해결하기 위해서 주변 블록의 문맥을 분석한다. 집합 A의 경우, 이전 부/복호된 영상의 한 행에서 차지하는  $mv_0$ 의 비율에 따라서  $mv_{264}, mv_{ext}$  중에 적절히 하나를 선택하고, 집합 B의 경우, 부호화하고자하는 블록의 주변 문맥에 따라서  $mv_a, mv_b, mv_c, mv_0$  중에 적절히 선택하도록 한다. 마지막으로 이렇게 선택된 두 개의 대표 움직임 후보 벡터 중 율-왜곡 관점에서 최적의 움직임 후보를 선택하여 스킵 모드를 부호화하고, 이에 대한 인덱싱 비트를 추가적으로 부호화하는 것으로 제안하는 기법이 수행된다.

#### IV. 실험 결과

제안하는 기법의 부호화 성능을 비교 평가하기 위해서 H.264/AVC 참조 모델인 JM11.0 기반의 KTA 2.4 소프트웨어에 기존의 경쟁 기반 움직임 벡터 코딩 기법<sup>[5]</sup>과 제안하는 기법을 구현하였다. 본 실험에서는 VCEG 공통 실험 조건<sup>[8]</sup>하에서 Baseline 프로파일과 IPPP 구조를 사용하였다. 단, 참조 프레임은 1이다. 두 기법의 성능 평가는 BD-PSNR과 BD-RATE<sup>[9]</sup>를 통해 비교의 기준이 되는 KTA 2.4 소프트웨어와 상대적인 비교를 통해 이루어졌다. 표 1은 기존의 코딩 기법과 제안하는 코딩 기법의 성능 평가를 나타낸 표이다. 제안하는 기법은 기존의 코딩 기법에 비해서 비트율 측면에서 평균적으로 약 0.7% 정도의 비트를 절감하였으며, PSNR 측면에서는 0.03dB 정도의 향상을 가져왔다.

이 결과는 제안하는 코딩 기법이 기존의 코딩 기법

표 1. 기존의 코딩 기법과 제안하는 코딩 기법의 성능 평가  
Table 1. Performance comparisons between the conventional and the proposed coding method

실험 영상		기존의 코딩 기법 [5]		제안하는 코딩 기법	
		BD-PSNR	BD-RATE	BD-PSNR	BD-RATE
QCIF	Container	-0.032	0.712	0.050	-1.113
	Silent	-0.016	0.241	0.048	-0.841
CIF	Paris	0.021	-0.357	0.044	-0.771
	Foreman	0.108	-2.861	0.111	-2.987
720p	Shuttlestart	0.111	-3.230	0.116	-3.393
	Night	0.002	-0.064	0.033	-0.881
	Bigships	0.105	-4.024	0.107	-4.046
평균 성능		0.043	-1.369	0.073	-2.005

과는 달리 영상의 특성을 고려하여 두 개의 적응적인 움직임 후보 벡터를 제공하기 때문에 나타나는 것으로, 기존의 코딩 기법의 성능 향상 정도(-1.369%)를 고려했을 때 제안하는 알고리즘은 약 0.636%(=2.005-1.369) 비트율을 개선하였으므로 평균 46%(0.636/1.369\*100=46%)의 성능 효과에 해당한다. 특별히 기존의 코딩 기법은 Container, Silent와 같은 카메라가 고정되어 있는 영상에 대해서 적절하지 않은 움직임 후보 벡터를 제공하게 되어 H.264/AVC 앵커에 비해 그 성능이 열화 되는 문제를 가지는 반면에, 제안하는 기법은 기존 기법 대비 대략 최대 2%까지 비트율 절감 효과를 가져왔다.

제안하는 기법의 성능 향상 원인을 보다 구체적으로 설명하기 위해, 기존의 기법과 제안 방식의 경쟁 기반 움직임 벡터 예측치를  $D_{tcp}$  (euclidean Distance between True Motion Vector to best Candidate motion Predictor)관점에서 분석해보았다<sup>[7]</sup>.  $D_{tcp}$ 는 식 (6)과 같이 최적으로 골라진 후보 벡터(Best candidate motion vector)와 실제 선택된 움직임 벡터 값(True motion vector)과의 차이로, 실제 움직임 벡터는  $D_{tcp}$ 의 측정을 위해 이상적으로 구한 값이다. 만일 후보 벡터들이 적절히 제공되어 실제 움직임 벡터와 차이가 크지 않다면,  $D_{tcp}$ 는 작은 값을 가지게 되고, 반면에 크다면  $D_{tcp}$ 는 큰 값을 가지게 된다.

$$D_{tcp} = \frac{1}{N} \sum_{n=1}^N \{ (mv_x^T - mv_x^c)^2 + (mv_y^T - mv_y^c)^2 \} \quad (6)$$

여기서  $(mv_x^T, mv_y^T)$ 는 현재 블록의 실제 움직임 벡터이고,  $(mv_x^c, mv_y^c)$ 은 움직임 예측 후보들 중에

가장 우수하다고 선택된 예측 움직임 벡터이다.  $N$ 은 한 프레임의 매크로블록 총 개수를 의미하며, 따라서  $D_{tcp}$ 는 한 프레임 당 평균 거리를 의미하게 된다.

그림 5는 Silent, Paris 영상에 대한 프레임 당 평균  $D_{tcp}$  변화량을 나타낸 그래프이다. 가로축은 프레임 번호를 의미하며, 세로축은 프레임 당 평균  $D_{tcp}$ 를 의미한다. 그림 5의 점선으로 표기된 부분은 기존의 코딩 기법을 의미하며, 실선으로 표기된 부분은 제안하는 기법의  $D_{tcp}$  변화량을 의미한다. 그림 5의 (a)와 (b)를 통해 알 수 있듯이, 제안하는 기법의 전반적인  $D_{tcp}$ 의 값이 종래의 기법에 비해서 상대적으로 낮은 수치를 기록하고 있다. 따라서 이를 통해 제안하는 알고리즘이 기존의 코딩 기법에 비해 상대적으로 적절한 움직임 예측 후보 벡터들을 제공해준다는 것을 확인할 수 있다.

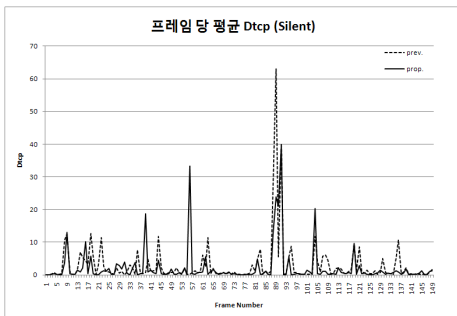
### V. 결 론

본 논문에서는 고정된 움직임 예측 후보 제공으로 인해 발생하는 경쟁 기반 움직임 벡터 코딩 기술의 한

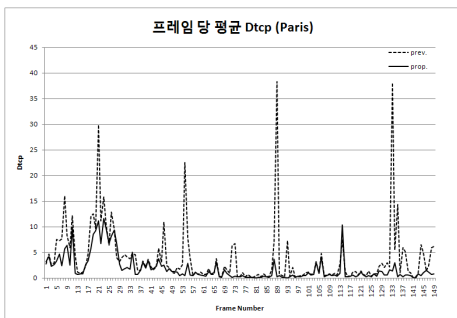
계를 해결하기 위해, 영상의 특성을 반영하여 효율적으로 움직임 예측 후보를 제공하기 위한 체계적인 접근 방법을 제안하였다. 영상의 움직임은 크게 영상의 대부분 영역에서 규칙적인 전역 움직임과 국소적으로 발생하는 불규칙적인 움직임으로 구분할 수 있기 때문에 본 논문에서는 이러한 움직임들을 적절히 반영할 수 있도록 움직임 예측 후보 벡터들을 두 개의 후보 집합으로 분류하고, 두 개의 집합에서 대표 움직임 예측 후보를 영상의 특성을 반영한 카메라의 움직임 여부와 문맥 기반의 모델링 기법으로 선택하도록 하였다. 이와 같은 선택 기법을 인터 모드 중 가장 효율적인 부호화 모드로 알려진 스킵 모드의 움직임 예측 후보 선택에 적용하여 그 성능을 대표적 기존 방식과 비교 분석하였다. 결과적으로 제안하는 기법은 BD-RATE 관점에서 종래의 가장 우수한 기법에 비해서 추가적으로 최대 2배까지 성능 개선을 이룰 수 있었는데, 이는 제안 방식이 선택한 예측 후보 움직임 벡터가 기존 방식의 그것과 비교하여 실제 움직임에 보다 가깝기 때문임을 확인할 수 있었다.

### 참 고 문 헌

- [1] ITU-T Recommendation H.264 and ISO/IEC 14496-10 AVC, "Advanced video coding for generic audiovisual services," version 3: 2005.
- [2] ISO/IEC JTC1/SC29/WG11, "Draft Call for Proposals on High-Performance Video Coding," *ISO/IEC JTC1/SC29/WG11 Doc. N10926*, Xi'an, Oct. 2009.
- [3] ITU-T Recommendation H.263, "Video Coding for Low Bitrate Communication," Jun. 1996.
- [4] S. D. Kim and J. B. Ra, "An efficient motion vector coding scheme based on minimum bitrate prediction," *IEEE Trans. on Image. Proc.*, Vol.8, No.8, pp.1117-1120, Aug. 1999.
- [5] J. Jung, and G. Laroche, "Competition-Based Scheme for Motion Vector Selection and Coding", *ITU-T SG16/Q6 Doc. VCEG-AC06*, Klagenfurt, Jul. 2006.
- [6] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, G. J. Sullivan, "Rate-constrained coder control and comparison of video



(a) Silent 영상에 대한 프레임 당 평균  $D_{tcp}$



(b) Paris 영상에 대한 프레임 당 평균  $D_{tcp}$

그림 5. 각 영상에 대한 프레임 당 평균  $D_{tcp}$  변화  
Fig. 5. Average  $D_{tcp}$  Variations per frame for each sequence.

coding standards”, IEEE Trans. on Circuits and System for Video Tech., Vol.13, No.7, pp.688-703, Jul. 2003.

- [7] Y.G. Kim, Y. Choe, “Candidate motion vectors for error concealment of video signals,” IEICE Trans. Information and systems, Vol.E84D, No.3, pp.427-431, Mar. 2001.
- [8] T. Tan, G. Sullivan, and T. Wedi, “Recommended Simulation Common Conditions for Coding Efficiency Experiments Revision 3,” ITU-T SG16/Q6 Doc. VCEG-A110, Berlin, Jul. 2008.
- [9] G. Bjontegaard, “Calculation of Average PSNR Differences Between RD-Curves,” ITU-T SG16/Q6 Doc. VCEG-M33, Texas, Arp. 2001.

김 성 제 (Sungjei Kim)

정회원



2004년 9월 연세대학교 기 전  
전자공학과 학사  
2006년 9월 연세대학교 전기전  
자공학과 석사  
2006년 9월~현재 연세대학교  
전기전자공학과 박사 재학 중

<관심분야> 비디오 코덱 개발, HW/SW co-design

김 용 구 (Yong-Goo Kim)

정회원



1993년 2월 연세대학교 전기  
공학과 학사  
1995년 8월 연세대학교 전기  
및컴퓨터공학과 석사  
2001년 8월 연세대학교 전기  
전자공학과 박사  
2002년 9월~2006년 2월 (주)  
온타임텍 연구소장

2009년 3월~현재 : KGIT 뉴미디어학부 교수

<관심분야> 비디오/영상 통신, 멀티미디어 신호 처  
리 및 응용

최 윤 식 (Yoonsik Choe)

중신회원



1979년 2월 연세대학교 전기공  
학과 학사  
1984년 5월 Case Western  
Reserve 대학 시스템 공학  
과 석사  
1987년 5월 Pennsylvania 주립  
대학 전기공학과 석사

1990년 12월 Purdue 대학 전기공학부 박사

1993년 3월~현재 연세대학교 전기전자공학 교수

<관심분야> 비디오/영상 신호 처리 및 해석