

3차원 비트율-왜곡 최적화 기반 블록 부호화를 이용하는 임베디드 비디오 압축 방법

양 창 모*, 정 광 수°

An Embedded Video Compression Scheme Using a Three-Dimensional Rate-Distortion Optimization Based Block Coder

Chang Mo Yang*, Kwangsue Chung°

요 약

본 논문에서는 3차원 비트율-왜곡 최적화 기반 블록 부호화를 이용하는 새로운 임베디드 비디오 압축 방법을 제안한다. 제안한 방법에서는 입력되는 비디오 프레임에 움직임 보상 시간적 필터링(Motion Compensated Temporal Filtering, MCTF)를 적용하여 비디오의 시간적 중복성을 제거한 후, 비디오 프레임에 2차원 이산 웨이블릿 변환을 수행하여 공간적 중복성을 제거한다. 이러한 방법으로 생성된 3차원 웨이블릿 계수들은 비트율-왜곡 비 기댓값에 따라 정렬되며 3차원 블록분할 부호화 방법을 이용하여 부호화된다. 또한 제안한 방법은 임베디드 특징을 유지하면서도 효과적으로 컬러 비디오를 부호화하는 방법과 효율적인 비트율 제어 방법을 사용한다. 실험 결과는 제안한 방법이 임베디드 비트스트림을 생성하면서도 기존의 비디오 압축 방법과 비교하여 우수한 성능을 제공함을 보여준다.

Key Words : Wavelet, Embedded, Video Compression, Block Coder, MCTF

ABSTRACT

In this paper, we propose a new embedded video compression scheme which uses three-dimensional rate-distortion optimization based block coder. After the proposed scheme removes temporal redundancy by applying the motion compensated temporal filtering(MCTF) on input video frames, two dimensional discrete wavelet transform is applied on video frames to remove spatial redundancy. The three-dimensional wavelet coefficients generated in this way are sorted according to their expected rate-distortion slope and encoded by using the three-dimensional block partition coding method. The proposed scheme also uses both the effective color video coding method which maintains embedded features, and the efficient bit-rate control method. Experimental results demonstrate that the proposed scheme not only produces embedded bit-streams, but also outperforms existing video compression schemes.

* 본 연구는 미래창조과학부 및 정보통신기술진흥센터의 정보통신·방송 연구개발사업의 일환으로 수행되었습니다.[B0101-16-0525, (2세부) 대규모 실시간 비디오 분석에 의한 전역적 다중 관심객체 추적 및 상황 예측 기술 개발]

• First Author : Smart Media Research Center, Korea Electronics Technology Institute, cmyang@keti.re.kr, 정희원

° Corresponding Author : Department of Communications Engineering, Kwangwoon University, kchung@kw.ac.kr, 종신회원

논문번호 : KICS2016-09-257, Received September 13, 2016; Revised September 30, 2016; Accepted October 7, 2016

I. 서 론

최근, 네트워크 접근성이 좋아지고 모바일 디바이스가 보편화됨에 따라 다양한 환경에서 비디오 서비스를 제공하기 위한 압축 및 전송 기술에 대한 관심이 높아지고 있다¹⁻²⁾. 이러한 흐름에 발맞추어 국제 표준화 기구에서 H.26x 권고안이나 MPEG 표준을 제정한 이후, 표준 기술들이 CD(Compact Disk), DVD(Digital Versatile Disk), 화상 회의, 디지털 TV 등의 다양한 응용분야에서 활발히 사용되고 있다. 그러나 기존의 표준 기술들은 비디오 프레임의 일정한 크기를 가지는 블록으로 분할하여 이산 여현 변환(Discrete Cosine Transform, DCT)과 양자화를 수행하기 때문에 높은 비트율에서는 우수한 비디오 압축 성능을 제공하는 장점이 있는 반면 낮은 비트율에서 블록화 현상(Blocking artifacts)이 발생할 뿐만 아니라 압축된 비디오의 프레임을 복호화할 때 이전 프레임 정보를 이용하는 폐루프(Closed loop) 구조를 가짐으로서 비디오 프레임간 에러 전파(Error propagation) 문제가 발생하는 단점을 가진다.

기존 표준 기술들이 가지는 이러한 문제점들을 해결하기 위한 대안으로서, 이산 웨이블릿 변환(Discrete Wavelet Transform, DWT)을 기반으로 비디오를 부호화하는 방법들이 활발히 연구되고 있다. 기존의 표준 기술들이 비디오 프레임간에 존재하는 시간적 중복성을 제거하기 위해 블록 기반 움직임 추정(Motion estimation)과 움직임 보상(Motion compensation)을 수행하는 것과는 달리, 이산 웨이블릿 변환 기반 비디오 압축 방법에서는 1차원 변환을 수행하여 비디오 프레임간에 존재하는 시간적 중복성을 제거하기 때문에 개루프(Open loop) 구조를 가지며 비디오 프레임간 에러 전파 문제가 발생하지 않는다. 또한, 이산 웨이블릿 변환 기반 비디오 압축 방법에서는 시간적 중복성이 제거된 변환 프레임에 2차원 변환을 수행하여 공간적 중복성을 제거한 후 GOP(Group Of Pictures) 단위의 비트평면 부호화를 수행하기 때문에, 낮은 비트율에서도 블록화 현상이 발생하지 않을 뿐만 아니라 GOP 단위로 생성된 비트스트림에 임의의 지점에서 잘림이 발생하더라도 복호기가 잘려진 비트스트림을 이용하여 복호를 수행할 수 있는 임베디드 부호화 특징을 가지게 된다.

일반적으로 이산 웨이블릿 변환 기반 비디오 압축 방법³⁻¹⁹⁾에서는 입력되는 비디오 프레임을 GOP 단위로 분할하여 3차원 이산 웨이블릿 변환을 수행한 후, 3차원 웨이블릿 계수들을 3차원 셋³⁻⁷⁾이나 3차원 블

록⁸⁻¹⁵⁾으로 구성하여 중요도를 판별하고 부호화하는 방법을 사용한다. 이때, 3차원 셋이나 3차원 블록의 중요도를 판별하기 위해서는 지수적으로 감소하는 크기 임계값과의 비교 과정이 수행되며, 3차원 셋이나 3차원 블록이 중요하다고 판단되는 경우 연속적으로 셋분할 혹은 블록분할 과정을 수행한다. 그러나 기존의 비디오 압축 방법들은 간편하면서도 임베디드 비트스트림을 생성하기 수월한 장점이 있는 반면, 부호화 과정이 비트율-왜곡의 관점에서 최적화되어 있지 않은 단점을 가진다¹⁶⁻¹⁹⁾.

본 논문에서는 기존의 임베디드 비디오 압축 방법들이 가지는 단점을 개선하기 위해, “3차원 비트율-왜곡 최적화 기반 블록 부호화(Three-Dimensional Rate-Distortion optimization based Block Coder, 3D-RDBC)”를 이용하는 새로운 임베디드 비디오 압축 방법을 제안한다. 제안하는 비디오 압축 방법에서는 먼저 연속적으로 입력되는 비디오 프레임을 GOP로 구성한 후, 시간적 중복성 제거를 위해서 MCTF를 적용하고 공간적 중복성 제거를 위해서 2차원 이산 웨이블릿 변환을 수행함으로써 3차원 웨이블릿 계수를 생성한다. 다음으로는 이미지 압축을 위해 고안되었던 2D-RDBC²⁰⁾ 부호화 방법을 비디오 부호화를 위해 3차원으로 확장한 3D-RDBC를 이용하여 3차원 웨이블릿 계수들을 부호화한다. 3D-RDBC는 3차원 웨이블릿 계수들을 3차원 블록이나 단일 계수로 구분한 후, 비트율-왜곡비 기댓값(Expected rate-distortion slope)에 따라 3차원 블록이나 계수를 정렬하며 부호화한다. 또한, 3D-RDBC는 임베디드 특징을 유지하면서도 효과적으로 컬러 비디오를 부호화하는 방법과 효율적인 비트율 제어 방법을 사용한다.

본 논문의 구성은 다음과 같다. 먼저 II장에서는 이산 웨이블릿 변환을 이용하는 기존의 임베디드 비디오 압축 방법에 대해 살펴본다. III장에서는 본 논문에서 제안한 3D-RDBC 기반 비디오 압축 방법에 대해 기술한다. 마지막으로 IV장 및 V장에서는 기존의 비디오 압축 방법과 본 논문에서 제안한 방법의 성능을 평가하고 결론을 맺는다.

II. 관련 연구

초기의 이산 웨이블릿 변환 기반 비디오 압축 방법은 H.26x 권고안이나 MPEG과 같은 표준 기술들에서 사용하는 비디오 압축 방법과 같은 구조를 가지며, 공간적 중복성을 제거하기 위한 블록기반 이산 여현 변환을 이산 웨이블릿 변환으로 대체하고 양자화 과정

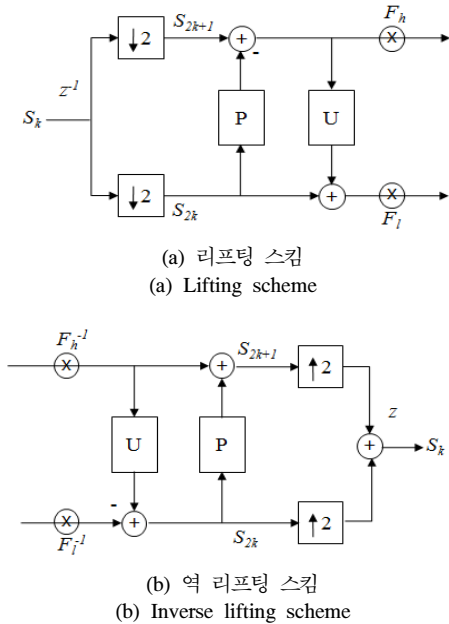


그림 1. 리프팅 기반 MCTF 구조
Fig. 1. Structure of lifting-based MCTF

을 이미지 압축을 위해 고안된 임베디드 부호화 방법으로 대체하는 방법이 사용되었다²¹⁾. 그러나 이러한 방법은 기존의 표준 기술들이 가지는 여러 전과 문제와 블록화 현상을 그대로 가지는 단점이 있을 뿐만 아니라 이산 웨이브렛 변환 기반 부호화의 주요 장점인 임베디드 비트스트림을 생성하지 못하는 단점이 있다.

이러한 단점을 극복하고자 입력되는 비디오 프레임에 이산 웨이브렛 변환과 다중해상도 움직임 예측 및 움직임 보상(Multi-resolution Motion Estimation/Motion Compensation, MRME/MC)²²⁾을 수행한 후, 이미지 압축을 위해 고안된 임베디드 부호화를 수행하는 방법들이 제안되었다. 그러나 이러한 방법은 임베디드 비트스트림을 생성하기에는 용이한 반면, 웨이브렛 변환 영역에서 움직임 예측 및 보상이 수행되므로 비디오 프레임의 시간적 중복성을 충분히 제거하지 못함으로 인해 부호화 효율이 좋지 못한 단점을 가진다.

최근에는 종래의 단점들을 극복하기 위한 3차원 이산 웨이브렛 변환기반 비디오 압축 방법들이 활발히 연구되고 있다. 기존의 표준 기술들과는 달리 3차원 이산 웨이브렛 변환기반 비디오 압축 방법에서는 비디오 프레임의 시간적 중복성을 1차원 변환을 통해 제거한 후, 공간적으로 2차원 변환을 수행하여 3차원

부대역을 생성하는 방법을 사용한다. 이때 시간적 중복성을 제거하기 위한 다양한 방법들이 연구되었으며, 최근에는 그림 1과 같은 리프팅 스킴(Lifting scheme)²³⁾을 기반으로 구현되는 MCTF(Motion Compensated Temporal Filtering) 방법이 활발히 연구되고 있다. MCTF는 웨이브렛 기법을 이용하여 시간 확장성을 제공함과 동시에 프레임간 중복성을 제거하기 위해 고안된 것으로서, 인접한 저주파 프레임간의 움직임 예측을 통해 샘플을 정렬하고 변환을 수행하는 과정을 반복함으로써, 연속되는 비디오 프레임을 여러 개의 주파수 대역으로 분해한다.

비디오의 시간 및 공간적 중복성 제거를 위해 3차원 이산 웨이브렛 변환이 수행되면 3차원 부대역이 형성된다. 이때, 3차원 부대역에 존재하는 웨이브렛 계수들은 임베디드 이미지 압축에 사용된 부모-자식간 상관관계(Parent-children relationship)^{13,20)}나 지역화 특성(Localization property)^{18,20)}과 같은 특징을 3차원 부대역내에서 유지하게 된다. 따라서 3차원 부대역 부호화를 위해서는 이미지 압축에 사용된 임베디드 부호화 방식을 3차원으로 확장 적용하는 것이 일반적이며, 대표적인 부호화 방법으로는 3차원 셋분할 부호화 방법¹³⁻⁷⁾, 3차원 블록분할 부호화 방법⁸⁻¹⁵⁾, PCRD(Post Compression Rate-Distortion) 기반 3차원 부대역 부호화 방법¹⁶⁻¹⁹⁾이 있다.

3차원 셋분할 부호화 방법은 웨이브렛 계수의 대역간 상관관계인 ‘부모-자식간 상관관계’를 이용하는 방법으로서, GOP 단위의 3차원 부대역에 공간적으로 2차원 트리를 형성하고 시간적으로 1차원 트리를 형성함으로써 3차원 트리구조를 완성한 후, 지수적으로 감소하는 크기 임계값을 이용하여 제로트리 분할 부호화를 수행하는 방법이다. 3차원 블록분할 부호화 방법은 대부분의 웨이브렛 계수가 아주 작은 값을 가지며 일부 큰값을 가지는 웨이브렛 계수만이 특정영역에 분포한다는 통계적인 특징인 ‘지역화 특성’을 이용한다. 3차원 블록분할 부호화 방법에서는 웨이브렛 계수들을 큐브형태의 3차원 블록으로 구성한 후, 지수적으로 감소하는 크기 임계값에 따라 3차원 블록의 중요도를 부호화한다. 3차원 블록이 중요하다고 판단되는 경우에는 해당 블록을 여러 개의 3차원 부블록으로 분할하는 과정을 반복함으로써 비디오를 GOP 단위로 부호화한다. PCRD 기반 3차원 부대역 부호화 방법은 일반적으로 두가지 단계로 구성되며, 첫 번째 단계에서는 GOP내의 웨이브렛 계수들을 일정한 크기의 3차원 코드블록(Code-block)으로 구성하여 독립적인 부호화를 수행함으로써 다수개의 비트스트림을 생성한

다. 두 번째 단계에서는 생성된 비트스트림의 임베디드 특징과 비트율-왜곡 최적화 이론을 결합하여 다수 개의 독립적인 비트스트림을 재조합함으로써, 하나의 비트스트림을 생성하는 과정이 수행된다.

3차원 부대역 부호화의 기본 개념은 이미지 압축에 사용된 임베디드 부호화 방법과 매우 유사하지만, 가변 크기의 GOP에 따라 비정형적인 시간축의 신호를 이용하여 3차원 셋이나 블록을 형성하는 방법^[5,7,8], 3차원 셋이나 블록을 효율적으로 분할하는 방법^[4,9], GOP 단위의 비트스트림에 대한 비트율 제어 방법^[10,16] 등의 차이점이 있어, 관련 연구가 활발히 수행되고 있다.

III. 제안한 임베디드 비디오 압축 방법

3.1 비디오 압축 구조

기존의 비디오 압축에서 사용된 3차원 셋분할 부호화 방법 및 3차원 블록분할 부호화 방법들은 3차원 웨이브렛 변환 계수를 비트평면 단위로 부호함으로써 빠른 압축이 가능한 반면 비트율-왜곡의 관점에서 최적화되어 있지 않은 단점이 있다. 반면 PCRD를 수행하여 비트율-왜곡 최적화를 수행하는 3차원 부대역 부호화 방법들은 코드블록들을 독립적으로 부호화해야 함으로 엔트로피 부호화를 위한 확률 샘플이 적어진다는 단점을 가지며, 이미지 압축에서 PCRD 기반 부호화 방법이 보인 우수한 성능과 비교하여 3차원 확장 부호화 방법에서는 좋은 성능을 보여주지 못하고 있다^[8]. 이러한 단점을 개선하기 위해 본 논문에서는 이미지 압축을 위한 고안된 2D-RDBC를 비디오 부호화를 위해 3차원으로 확장한 3D-RDBC 기반의 비디오 압축 방법을 제안한다.

그림 2는 본 논문에서 제안한 3D-RDBC 기반의 비디오 압축 구조를 보여준다. 그림 2에서 볼 수 있듯이, 제안하는 비디오 압축 방법에서는 비디오의 시간적 중복성과 공간적 중복성을 제거하기 위해 GOP 단위의 MCTF와 2차원 이산 웨이브렛 변환을 수행한다. 그런 후에, MCTF와 2차원 이산 웨이브렛 변환을 통

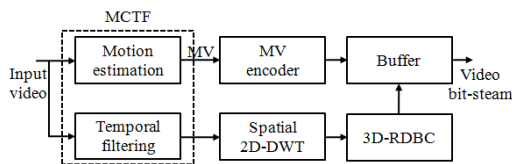


그림 2. 3D-RDBC 기반의 비디오 압축 구조
Fig. 2. Structure of 3D-RDBC based video compression

해 생성된 3차원 웨이브렛 계수들을 3D-RDBC를 이용하여 부호화한다. 3D-RDBC에서는 3차원 웨이브렛 계수들을 3차원 블록이나 단일 계수들로 구성된 후, 3차원 블록과 계수들을 비트율-왜곡비 기댓값에 따라 정렬하는 동시에 비트평면 부호화를 수행함으로써 비트율-왜곡 최적화된 임베디드 부호화 과정을 수행한다. 비트율-왜곡비 기댓값에 따라 3차원 블록이나 계수들을 정렬하는 방법으로는 일정한 비율로 감소하는 비트율-왜곡 임계값이 사용되며, 3차원 블록이나 계수들의 비트율-왜곡비 기댓값이 비트율-왜곡 임계값과 비교하여 큰 경우에만 비트평면 부호화를 수행한다. 이러한 과정을 통해 부호화된 비디오 압축 비트스트림은 비트평면 부호화 과정을 통해 생성된 것이므로 임베디드 특징을 가지게 된다.

3.2 2D-RDBC의 3차원 확장

3.2.1 3차원 블록 분할 부호화

2차원 블록분할 부호화 방법을 사용하는 2D-RDBC와는 달리, 3D-RDBC에서는 웨이브렛 계수들이 가지는 지역화 특성을 이용하여 0으로 부호화되는 대부분의 웨이브렛 계수들을 적은 비트로 부호화하기 위해서 3차원 블록분할 부호화를 사용한다. 3D-RDBC에서는 먼저 GOP 단위로 생성되는 3차원 부대역을 여러 개의 3차원 블록으로 구성한다. 3차원 블록분할 부호화를 수행할 때에는 3차원 블록에 대한 중요도를 판별하고 부호화하는데, 블록이 중요하다고 판별되는 경우 그림 3과 같은 3차원 옥트리(Octree) 구조의 블록분할을 수행함으로써 3차원 블록 B는 8개의 3차원 부블럭 $B_i(i \in \{1,2,3,4,5,6,7,8\})$ 로 분할된다. 이때, 3차원 블록 B의 중요도는 수식 (1)과 같은 방법으로 판별된다.

$$S_{T_k}(B) = \begin{cases} 1, & \text{if } T_k \leq \max_{(i,j,t) \in B} |c_{i,j,t}| < T_{k-1} \\ 0, & \text{else} \end{cases} \quad (1)$$

수식 (1)에서 T_k 는 비트평면 부호화를 위한 k번째 크기 임계값을 의미하며, $c_{i,j,t}$ 는 GOP단위의 3차원 부대역에서 (i,j,t) 좌표에 위치한 웨이브렛 계수를 의미한다. 또한, $S_{T_k}(B)$ 는 크기 임계값 T_k 에 대한 3차원 블록 B의 중요도를 나타내는데, 1과 0이 각각 ‘중요 블록’ 및 ‘비중요 블록’의 의미를 가진다. 이러한 3차원 블록 부호화 과정은 지속적으로 감소하는 크기 임계값 T_k 에 대해 반복적으로 수행된다.

또한 3D-RDBC에서는 더 이상 블록분할을 수행할

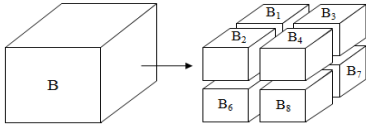


그림 3. 3차원 옥트리 블록 분할
Fig. 3. Three-dimensional octree block partitioning

수 없는 경우나 3차원 블록분할 부호화를 수행하는 것보다는 3차원 블록내의 웨이브렛 계수들을 독립적으로 부호화하는 것이 더 이득이라고 판단되는 경우, 웨이브렛 계수들의 중요도를 수식 (2)와 같은 방법으로 판별하고 부호화를 수행한다. 수식 (2)에서 1은 ‘중요 계수’를 0은 ‘비중요 계수’를 의미한다.

$$S_{T_k}(c_{i,j,t}) = \begin{cases} 1, & \text{if } T_k \leq |c_{i,j,t}| < T_{k-1} \\ 0, & \text{else} \end{cases} \quad (2)$$

수식 (1)과 (2)에서 보듯이, 3D-RDBC는 크기 임계값을 사용하여 3차원 블록이나 웨이브렛 계수들의 중요도를 판별하고 부호화하는데, 크기 임계값 T_k 를 2의 승수의 값으로 선택함으로써 비트평면 부호화가 수행된다.

3.2.2 비트율-왜곡 최적화

2D-RDBC 부호화 방법과 유사하게, 3D-RDBC는 3차원 블록이나 웨이브렛 계수들을 부호화함으로써 감소하는 왜곡의 기댓값 $E[\Delta D]$ 와 증가하는 비트율의 기댓값 $E[\Delta R]$ 을 사용하여 수식 (3)과 같이 정의되는 비트율-왜곡비 기댓값에 따라 3차원 블록이나 웨이브렛 계수들을 정렬함으로써 비트율-왜곡 최적화를 수행한다. 비트율-왜곡비 기댓값에 따라 3차원 블록이나 웨이브렛 계수들을 정렬하기 위해서는 일정한 비율로 감소하는 비트율-왜곡 임계값이 사용되며, 3차원 블록이나 웨이브렛 계수들의 비트율-왜곡비 기댓값이 비트율-왜곡 임계값보다 큰 경우에만 부호화를 수행한다.

$$\lambda = \frac{E[\Delta D]}{E[\Delta R]} \quad (3)$$

3D-RDBC를 이용하여 3차원 웨이브렛 계수를 부호화하는 과정은 웨이브렛 계수의 MSB(Most Significant Bit)를 부호화하는 단계, 3차원 블록을 부호화하는 단계, 웨이브렛 계수의 정제비트를 부호화하는 단계로 구성된다. 이때, 3차원 블록이나 웨이브렛 계수를 비트율-왜곡비 기댓값에 따라 정렬하기 위해서는 각 부호화 단계에서 비트율-왜곡비 기댓값을 계

산하는 것이 필수적이다. 웨이브렛 계수의 MSB를 부호화하는 단계 및 정제비트를 부호화하는 단계에서는 웨이브렛 계수의 좌표가 3차원으로 확장된 것을 제외하고는 2D-RDBC와 동일한 과정을 수행한다. 따라서 웨이브렛 계수 $c_{i,j,t}$ 의 MSB를 부호화함으로써 얻을 수 있는 비트율-왜곡비 기댓값 λ_p 및 정제비트를 부호화함으로써 얻을 수 있는 비트율-왜곡비 기댓값 λ_r 은 $c_{i,j,t}$ 가 수식 (2)에 의해 1로 판단될 확률 p_s , 부호가 +일 확률 p_+ , 정제비트가 1일 확률 p_r 을 이용하여 수식 (4) 및 (5)와 같이 계산된다.

$$\begin{aligned} \lambda_p &= \frac{E[\Delta D_p]}{E[\Delta R_p]} = \frac{p_s \cdot 2.25 T_k^2}{p_s H(p_+) + H(p_s)} \\ &= \frac{2.25 T_k^2}{H(p_+) + H(p_s)/p_s} \end{aligned} \quad (4)$$

$$\lambda_r = \frac{E[\Delta D_r]}{E[\Delta R_r]} = \frac{0.25 T_k^2}{H(p_r)} \quad (5)$$

3차원 블록을 부호화함으로써 얻을 수 있는 비트율-왜곡비 기댓값 λ_b 는 블록에 속해있는 계수들을 독립적으로 부호화했을 때 얻어지는 비트율-왜곡비 기댓값 λ_c 와 3차원 옥트리 블록분할 부호화를 수행했을 때 얻어지는 비트율-왜곡비 기댓값 λ_o 를 비교하여 큰 값을 λ_b 로 결정하는 방법이 사용되며, 3차원 블록을 부호화하는 과정에서는 λ_b 의 값에 따라 3차원 블록분할 부호화 혹은 3차원 블록내 계수들에 대한 독립적인 부호화 과정이 수행된다. λ_b 는 3차원 블록에 속한 계수들을 독립적으로 부호화했을 때 감소하는 왜곡의 기댓값의 합 $E[\Delta D_s]$ 와 증가하는 비트율의 기댓값의 합 $E[\Delta R_s]$ 를 이용하여 수식 (6)과 같이 계산된다.

$$\lambda_s = \frac{E[\Delta D_s]}{E[\Delta R_s]} = \frac{\sum_{(i,j,t) \in B} E[\Delta D_p]}{\sum_{(i,j,t) \in B} E[\Delta R_p]} \quad (6)$$

λ_o 는 2D-RDBC의 쿼드트리 분할기반 비트율-왜곡비 기댓값 계산방법을 3차원으로 확장함으로써, 수식 (7)과 같이 계산된다.

$$\lambda_o = \frac{\sum_{i=1}^8 E[\Delta D_{B_i}] + \psi \lambda_c}{\sum_{i=1}^8 E[\Delta R_{B_i}] + 2} \quad (7)$$

수식 (7)에서 $E[\Delta D_{Bi}]$ 및 $E[\Delta R_{Bi}]$ 는 3차원 블록 B 가 그림 3과 같이 8개의 3차원 부블록 $B_i(i \in \{1,2,3,4,5,6,7,8\})$ 로 분할된 후, 각 부블록 B_i 를 부호화함으로서 감소하는 왜곡의 기댓값과 증가하는 비트율의 기댓값을 의미하며, Ψ 는 블록 B 의 중요도가 0인 경우 얻을 수 있는 비트율 이득을, λ_c 는 비트율 이득 Ψ 를 이용해서 다른 블록이나 계수들을 부호화함으로서 얻어지는 평균 비트율-왜곡비를 의미한다. Ψ 와 λ_c 는 2D-RDBC와 동일한 방법을 이용하여 수식 (8) 및 (9)와 같이 계산된다.

$$\Psi = \sum_{(i,j,t) \in B} (p_s - 1) \log_2(1 - p_s) - 1 \quad (8)$$

$$\lambda_c = \frac{\gamma_k + \gamma_{k-1}}{2} = 1.125 \gamma_k \quad (9)$$

수식 (7)에서 알 수 있듯이, 3차원 블록분할 부호화를 통해 얻을 수 있는 비트율-왜곡비 기댓값 λ_c 를 구하기 위해서는 3차원 블록분할을 반복적으로 수행하며, $E[\Delta D_{Bi}]$ 와 $E[\Delta R_{Bi}]$ 를 계산해야 한다. 3D-RDBC에서는 분할-합병(Split-merge) 개념을 이용하여 $E[\Delta D_{Bi}]$ 와 $E[\Delta R_{Bi}]$ 를 계산한다. 즉, 3차원 블록에 대한 3차원 옥트리 분할 과정을 반복적으로 수행한 후 트리노드 중단에서부터 상단 방향으로 트리노드 합병을 수행하며 각 트리노드의 $E[\Delta D_{Bi}]$ 와 $E[\Delta R_{Bi}]$ 를 계산하는 방법이 사용된다.

3.2.3 비트율 제어 및 임베디드 컬러 부호화

3D-RDBC에서는 비디오 GOP 단위의 임베디드 부호화 과정을 수행한다. 따라서 GOP 단위의 부호화 및 복호화 과정에서 목표 비트율에 도달하면 어떤 지점에서도 부호화나 복호화 과정을 중단할 수 있는 임베디드 특징을 가진다. 이러한 GOP 단위의 임베디드 특징을 이용하여, 3D-RDBC는 수식 (10)과 같은 방법을 이용하여 GOP 단위의 비트율 제어를 수행한다.

$$R_g = N_g \times R_t / F_r \quad (10)$$

수식 (10)에서 R_g 와 N_g 는 GOP를 부호화하기 위한 할당 비트수(Bit budget)와 GOP내의 프레임 개수를 의미하며, R_t 와 F_r 는 목표 압축 비트율과 비디오의 프레임율을 의미한다.

3D-RDBC는 각각의 컬러 정보를 별도로 부호화하는 기존의 표준 기술과는 달리, 모든 컬러 정보를 동일하게 취급하며 하나의 부호화 시스템에서 컬러 정

보를 동시에 부호화하는 임베디드 컬러 부호화 방식을 사용한다. 3D-RDBC에서는 입력되는 비디오의 컬러 정보에 대해 3차원 부대역을 생성한 후, 생성된 3차원 부대역을 컬러에 관계없이 3차원 블록이나 계수들로 구성하고 임베디드 부호화 과정을 수행한다.

3.3 3D-RDBC 구조

그림 4는 3D-RDBC의 부호화 구조를 보여준다. 3D-RDBC는 GOP내의 웨이브렛 계수나 3차원 블록을 구분하기 위해 LIP(List of Insignificant Pixels), LIB(List of Insignificant Blocks), LSP(List of Significant Pixels)의 세가지 리스트를 사용한다. LIP는 크기 임계값을 기준으로 중요하지 않다고 분류된 계수들을 포함하는 리스트이며, LIB는 중요하지 않다고 분류된 3차원 블록들을 포함하는 리스트이고, LSP는 중요하다고 판단된 웨이브렛 계수들을 포함하는 리스트이다. 세가지 리스트를 이용하여, 3D-RDBC는 초기화(Initialization), LIP 정렬(LIP sorting), LIB 정렬(LIB sorting), LSP 정제(LSP refinement), 갱신(Update), 엔트로피 부호화(Entropy coding)의 각 단계를 수행하여 웨이브렛 계수나 3차원 블록을 부호화한다. GOP를 부호화하기 위한 할당 비트수는 수식 (10)을 이용하여 초기화 단계에서 결정되며, 각 단계의 부호화 과정에서 할당 비트수에 도달하면 부호화 과정이 중단된다.

3.3.1 초기화 단계

초기화 단계에서는 GOP 부호화에 사용되는 할당 비트수를 결정하는 과정, 3차원 블록이나 웨이브렛 계수의 중요도를 판단하는 기준인 크기 임계값의 초기값 T_0 를 결정하는 과정, 3차원 블록이나 웨이브렛 계수의 부호화 수행여부의 판단 기준이 되는 비트율-왜곡 임계값의 초기값 γ_0 를 결정하는 과정, 각 리스트에 들어갈 항목을 결정하는 과정이 수행된다. 먼저, GOP 부호화에 사용되는 할당 비트수는 MCTF를 통해 생성된 움직임 벡터(Motion Vection, MV)를 부호화하

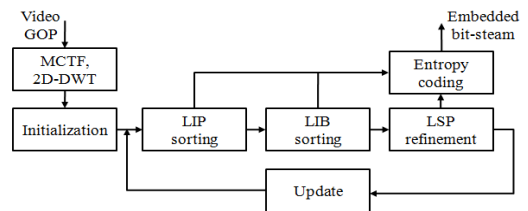


그림 4. 3D-RDBC의 부호화 구조
Fig. 4. Encoding structure of 3D-RDBC

는 데에 사용된 비트수를 포함하여 수식 (10)을 이용하여 결정된다. T_0 는 모든 컬러 정보들을 포함한 GOP 내의 웨이브렛 계수 $c_{i,j,t}$ 를 이용하여 수식 (11)과 같이 결정되며, Υ_0 는 수식 (12)와 같이 결정된다.

$$T_0 = 2^n, \quad n = \left\lfloor \log_2 \left(\max_{(i,j,t) \in GOP} |c_{i,j,t}| \right) \right\rfloor \quad (11)$$

$$\gamma_0 = \frac{1}{2} T_0^2 \quad (12)$$

T_0 와 Υ_0 가 결정되면 각 리스트에 들어갈 항목을 결정한다. 첫 번째 프레임의 공간적 부대역을 기준으로 시간적으로 GOP 길이만큼의 3차원 블록을 형성한 후 LIB에 초기화한다. 이때, GOP에 포함되어 있는 모든 컬러정보에 대해 같은 과정을 수행함으로써, 모든 컬러정보에 대한 3차원 블록들이 LIB에 초기화된다. LIB에 3차원 블록을 초기화할 때에는 T_0 를 함께 저장한다. 초기화 단계에서는 LIP 및 LSP는 리스트 항목을 가지지 않는다.

3.3.2 LIP 정렬 단계

LIP 정렬 단계에서는 리스트에 포함된 각 웨이브렛 계수 $c_{i,j,t}$ 의 비트율-왜곡비 기댓값 λ_p 를 수식 (4)에 의해 계산한 후, λ_p 의 값이 비트율-왜곡 임계값 Υ_k 보다 큰 리스트 계수의 부호화를 수행한다. 리스트 계수를 부호화할 때에는 먼저 수식 (2)를 이용하여 리스트 계수의 중요도를 부호화한 후, 리스트 계수가 중요하다고 판별된 경우 리스트 계수의 부호를 부호화한다. 부호화된 리스트 계수의 크기 임계값은 절반으로 갱신되며, 만약 해당 리스트 계수가 중요하다고 판별된 경우에는 리스트 계수를 LSP로 이동시킨다.

3.3.3 LIB 정렬 단계

LIB 정렬 단계에서는 수식 (6)과 (7)을 이용하여 리스트에 포함된 3차원 블록의 λ_b 와 λ_o 를 계산한 후, 큰 값을 3차원 블록의 비트율-왜곡비 기댓값 λ_b 로 결정한다. λ_b 가 결정되면 λ_b 의 값이 Υ_k 보다 큰 경우 3차원 블록 부호화 과정을 수행한다. 블록 부호화 과정에서는 λ_b 의 값이 λ_o 와 같은 경우에 3차원 블록에 포함된 모든 웨이브렛 계수들에 대한 독립 부호화가 수행되며, 해당 블록은 LIB에서 삭제된다. 3차원 블록에 포함된 웨이브렛 계수들의 독립 부호화를 수행할 때에는 LIP 정렬 단계에서 수행되었던 일련의 과정이 수행되며, 독립 부호화가 수행되는 웨이브렛 계수의 비트율-왜곡비 기댓값 λ_p 가 Υ_k 보다 작거나 0으로 부

호화된 웨이브렛 계수들은 LIP에 추가된다. 이때, λ_p 가 Υ_k 보다 작은 경우에는 3차원 블록의 크기 임계값이 해당 웨이브렛 계수의 크기 임계값으로 할당되며, 중요도가 0으로 부호화된 웨이브렛 계수들은 크기 임계값을 절반으로 갱신한다. 독립 부호화가 수행되는 웨이브렛 계수들 중 중요도가 1로 부호화된 계수들은 크기 임계값을 절반으로 갱신한 후 LSP에 추가한다.

λ_b 의 값이 λ_o 와 같은 경우에는 수식 (1)에 의해 판별되는 블록의 중요도가 부호화되고 블록의 크기 임계값이 절반으로 갱신된다. 블록의 중요도가 1인 경우에는 그림 3과 같은 3차원 옥트리 블록분할이 수행되며, 분할된 8개의 3차원 부블록들을 LIB에 추가한 후 현재의 블록을 LIB에서 삭제한다.

3.3.4 LSP 정제 단계

LSP 정제 단계에서는 리스트에 포함된 웨이브렛 계수의 정제 부호화를 통해 얻어지는 비트율-왜곡비 λ_r 를 수식 (5)에 의해 계산한 후, λ_r 의 값이 Υ_k 보다 큰 경우 정제 부호화를 수행한다. 정제 부호화에서는 리스트에 포함되어 있는 웨이브렛 계수 $c_{i,j,t}$ 의 크기 임계값 T_k 를 기반으로 $|c_{i,j,t}|$ 의 $\log_2 T_k$ 번째 비트가 부호화되며, 부호화된 웨이브렛 계수들의 크기 임계값은 절반으로 갱신된다.

3.3.5 갱신 단계

갱신 단계에서는 비트율-왜곡 임계값 Υ_k 를 수식 (13)과 같이 갱신한 후, LIP 정렬 부호화 단계로 되돌아가는 과정을 수행한다.

$$\gamma_k = \gamma_{k-1} / 1.25 \quad (13)$$

3.3.6 엔트로피 부호화 단계

3D-RDBC의 엔트로피 부호화에는 적응적 산술 부호기(Adaptive arithmetic coder)가 사용된다. 적응적 산술 부호기를 이용하여 웨이브렛 계수들의 중요도를 부호화하기 위해서는 EBCOT^[24] 알고리즘에서 사용된 컨텍스트(Context)에 웨이브렛 계수들의 부모 및 사촌 노드 계수를 조합하여 27개의 컨텍스트를 정의하고 이를 엔트로피 부호화에 사용한다. 또한, 계수들의 부호 및 정제 비트의 엔트로피 부호화를 위해서는 EBCOT와 동일한 컨텍스트를 사용하였으며, 블록의 중요도는 1비트를 사용하여 부호화한다.

IV. 실험 결과

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) dB \quad (14)$$

본 논문에서 제안한 3D-RDBC 기반 비디오 압축 방법을 소프트웨어로 구현하여 실험하였다. 비디오 압축을 위해 3D-RDBC와 결합되는 MCTF 방법으로는 Enhanced MC-EZBC^[11]에서 사용한 LGT 5/3 필터 및 Haar 2/2 필터 기반 적응적 MCTF(Adaptive MCTF) 방법이 사용되었다. MCTF에 사용되는 움직임 예측 및 보상을 위해서는 1/4 픽셀 정확도와 ±8의 탐색 영역을 기반으로 수행되는 계층적 가변 크기 블록 매칭(Hierarchical Variable Size Block Matching, HVSBM)이 적용되었으며, HVSBM에는 4×4에서 64×64 크기의 가변 크기 블록이 사용되었다. 2차원 공간 변환을 위한 이산 웨이블릿 변환 필터로는 Daub 9/7 필터가 사용되었으며, 비디오의 GOP는 16개의 프레임으로 구성하였다. 또한, 실험을 위한 비디오 시퀀스로는 96개의 프레임을 가지며 YUV 4:2:0 컬러 형식으로 구성된 SIF(352×240) 크기의 Mobile Calendar 및 Flower Garden과 CIF(352×288) 크기의 Bus 및 Football이 사용되었다. 부호화 성능을 측정하는 방법으로는 원 비디오와 복원된 비디오의 MSE(Mean Squared Error)를 기반으로 수식 (14)와 같이 정의되는 PSNR(Peak Signal-to-Noise Ratio)을 사용하였다.

표 1은 본 논문에서 제안한 3D-RDBC 기반 비디오 압축 방법과 MCTF를 기반으로 3차원 부대역 부호화를 수행하는 대표적인 임베디드 비디오 압축 방법인 MC-EZBC^[10] 및 Enhanced MC-EZBC^[11]의 컬러 부호화 성능을 비교한 것이다. MC-EZBC와 Enhanced MC-EZBC의 GOP 크기 및 MCTF 조건은 제안한 압축 방법과 동일 조건으로 실험하였다. 성능 비교를 위해서는 SIF 및 CIF 크기의 비디오 시퀀스를 30fps의 프레임율로 300kbps와 730kbps로 부호화하였다. 표 1에서 보듯이, 본 논문에서 제안한 비디오 압축 방법은 MC-EZBC 및 Enhanced MC-EZBC와 비교하여 우수한 컬러 부호화 성능을 제공한다. 4개의 비디오 시퀀스를 부호화함으로써 얻을 수 있는 YUV 컬러 부호화 이득을 평균하여 보면, 제안한 비디오 압축 방법은 MC-EZBC와 비교하여 300kbps에서 약 0~0.485dB의 성능향상을 보이며, 730kbps에서 약 0.1325~0.4275dB의 성능향상을 보인다. 같은 방법으로 제안한 비디오 압축 방법을 Enhanced MC-EZBC와 비교하여 보면, 제안한 비디오 압축 방법은 Enhanced MC-EZBC와 비교하여 300kbps에서 약 0.3525~0.3775dB의 성능향상을 보이며, 730kbps에서

표 1. 300kbps와 730kbps에서의 제안 방법과 다른 압축 방법과의 컬러 PSNR 비교
Table 1. Colour PSNR comparisons of the proposed scheme and other compression schemes at 300kbps and 730kbps

Video sequences	PSNR(dB) at 300kbps				Video sequences	PSNR(dB) at 730kbps			
	Compression schemes	Y	U	V		Compression schemes	Y	U	V
Mobile Calendar (SIF)	MC-EZBC	23.77	27.59	27.27	Mobile Calendar (SIF)	MC-EZBC	29.05	32.87	32.47
	Enh. MC-EZBC	23.79	27.15	26.84		Enh. MC-EZBC	28.93	32.51	32.17
	Proposed	24.22	27.21	27.07		Proposed	29.26	33.29	32.85
Flower Garden (SIF)	MC-EZBC	24.09	28.68	30.61	Flower Garden (SIF)	MC-EZBC	28.82	33.70	34.42
	Enh. MC-EZBC	24.30	28.16	30.08		Enh. MC-EZBC	29.07	33.25	33.75
	Proposed	24.53	28.39	30.04		Proposed	29.19	33.80	34.55
Bus (CIF)	MC-EZBC	27.80	37.01	38.39	Bus (CIF)	MC-EZBC	32.61	40.58	42.10
	Enh. MC-EZBC	27.74	36.74	37.96		Enh. MC-EZBC	32.74	39.80	41.41
	Proposed	28.32	37.08	38.38		Proposed	33.05	39.97	41.45
Football (CIF)	MC-EZBC	26.06	30.13	33.59	Football (CIF)	MC-EZBC	31.37	35.56	37.98
	Enh. MC-EZBC	26.32	30.32	33.57		Enh. MC-EZBC	31.59	36.04	38.19
	Proposed	26.59	31.07	34.37		Proposed	32.06	36.60	38.65

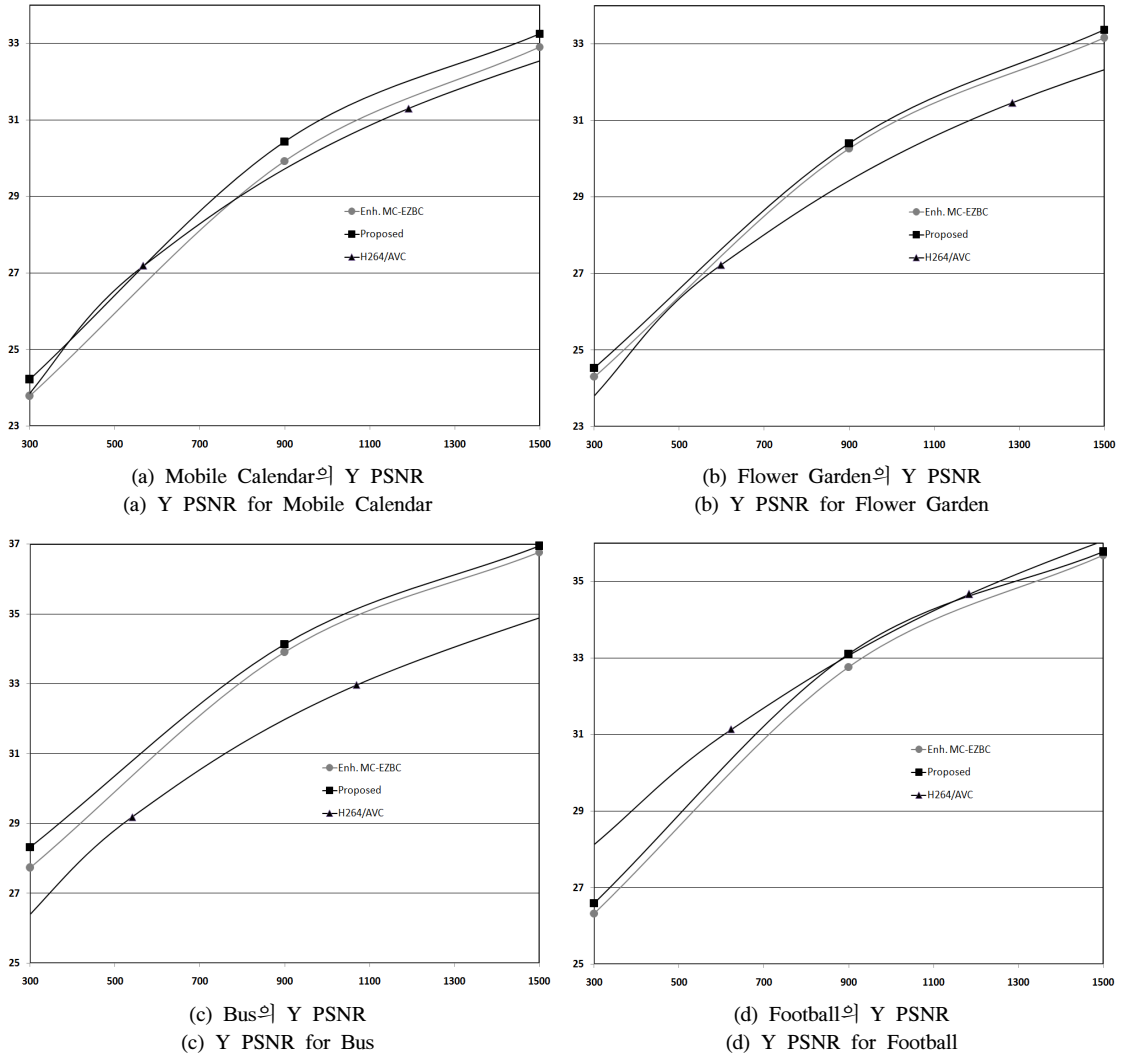


그림 5. 다양한 비트율에서 제안 방법과 다른 압축 방법과의 Y PSNR 비교

Fig. 5. Y PSNR comparisons of the proposed scheme and other compression schemes at various bit-rates

약 0.3075~0.515dB의 성능향상을 보인다.

그림 5는 본 논문에서 제안한 비디오 압축 방법과 Enhanced MC-EZBC 및 H.264/AVC(Advanced Video Coding)^[25]를 이용하여 비디오 시퀀스를 부호화한 결과 중, Y컬러에 대한 성능을 비교한 것이다. 그림 5에서 가로 및 세로축은 압축된 비트스트림의 비트율(kbps)과 PSNR을 의미한다. 성능비교를 위해 Enhanced MC-EZBC의 GOP 크기 및 MCTF 조건은 제안한 압축 방법과 동일 조건으로 실험하였다. 한편, H.264/AVC는 GOP 크기를 16으로 설정하고 움직임 예측 및 보상을 위해서는 1/4 픽셀 정확도와 ±8의 탐색 영역을 설정하였으며, 부호화되는 프레임은 IBBBPBBBP의 순서로 부호화하였다. 또한,

H.264/AVC는 임베디드 압축 방법이 아니므로 다양한 비트율에서 압축 성능을 확인하기 위해 양자화 파라미터를 변경하며 성능 시험을 수행하였다.

그림 5에서 보듯이, 제안한 비디오 압축 방법은 Enhanced MC-EZBC와 비교하여 우수한 성능을 제공하며, 특히 낮은 비트율에서 높은 부호화 이득을 제공한다. 이러한 현상은 제안 방법과 Enhanced MC-EZBC의 부호화 방법의 차이에서 기인한다. 제안 방법과 비교하여 Enhanced MC-EZBC는 비트율-왜곡 최적화 방법을 사용하지 않는 반면 엔트로피 부호화를 위한 고위(High-order) 컨텍스트 모델을 사용한다. 따라서 낮은 비트율에서는 비트율-왜곡 최적화 방법을 사용하는 제안 방법의 부호화 이득이 큰 반면, 비

트율이 높아짐에 따라 고위 컨텍스트 모델을 사용하는 Enhanced MC-EZBC와의 성능차이가 줄어들어 것으로 분석된다. 또한, 그림 5는 제안한 비디오 압축 방법이 H.264/AVC와 비교하여 평균적으로 우수한 성능을 제공함을 보여준다. 특히, 움직임이 적은 비디오 시퀀스에 대해 우수한 성능을 제공함을 알 수 있다. 반면에 Football과 같이 객체의 움직임이 많은 비디오 시퀀스에 대해서는 H.264/AVC가 제안한 비디오 압축방법과 비교하여 상대적으로 높은 부호화 이득을 제공한다. 이러한 현상은 H.264/AVC가 사용하는 움직임 예측 및 보상 방법과 제안한 비디오 압축 방법이 사용하는 MCTF의 차이에서 기인한다. H.264/AVC에서 사용하는 움직임 예측 및 보상 방법은 비디오 시퀀스 내에 객체들의 움직임이 많은 경우에도 움직임 예측 및 보상의 정확도가 어느 정도 보장되는 반면 MCTF는 그 정확도가 상대적으로 낮음으로 인해 전체적인 부호화 효율에 영향을 주는 것으로 분석된다.

V. 결 론

본 논문에서는 이미지 압축을 위해 고안되었던 2D-RDBC를 비디오 압축을 위해 3차원으로 확장한 3D-RDBC 기반 비디오 압축 방법을 제안하였다. 제안한 비디오 압축 방법에서는 MCTF와 2차원 이산 웨이블릿 변환을 이용하여 입력되는 비디오의 시간 및 공간적 중복성을 제거한 후, 3차원 부대역에 존재하는 웨이블릿 계수들을 3D-RDBC를 이용하여 부호화한다. 3D-RDBC를 이용하여 웨이블릿 계수들을 부호화할 때에는 비트율-왜곡비 기댓값에 따라 3차원 블록이나 계수를 정렬하는 동시에 비트평면 부호화를 수행함으로써, 임베디드 특징을 유지하면서도 비트율-왜곡 최적화를 수행한다. 또한, 3D-RDBC는 임베디드 컬러 부호화 방법과 간단하면서도 효율적인 비트율 제어 방법을 사용한다. 실험 결과는 제안한 비디오 압축 방법이 임베디드 특징을 유지하면서도 기존의 대표적인 임베디드 비디오 압축 방법과 비교하여 우수한 성능을 제공함을 보여주며, 특히 낮은 비트율에서 기존의 비디오 압축 방법과 비교하여 상대적으로 높은 부호화 이득이 발생함을 보여준다. 또한 임베디드 부호화 방식이 아닌 H.264/SVC와 비교하여 평균적으로 우수한 성능을 제공함을 보여준다. 특히 비디오 시퀀스 내에 객체의 움직임이 많은 경우에는 H.264/SVC와 비교하여 상대적으로 낮은 부호화 성능을 제공하는 반면 평활한 비디오 시퀀스에 대해서는

상대적으로 우수한 성능을 제공함을 보여준다.

본 논문에서 제안한 비디오 압축 방법에서는 웨이블릿 계수나 3차원 블록을 부호화하기 위해 비트율-왜곡비 기댓값을 계산해야 함으로 기존의 비디오 압축 방법과 비교하여 상대적으로 많은 연산량을 가진다. 향후의 연구 과제는 제안한 방법의 성능을 저하시키지 않으면서도 연산량을 감소시키는 것이다.

References

- [1] N. Kim, T. Song, W. Kim, and S. Pack, "A scalable video coding(SVC)-aware retransmission scheme," *J. KICS*, vol. 39, no. 02, pp. 95-101, Feb. 2014.
- [2] H. Lee, H. J. Shim, Y. Park, and B. Jeon, "Fast motion estimation using adaptive search range for HEVC," *J. KICS*, vol. 39, no. 4, pp. 209-211, Apr. 2014.
- [3] B. J. Kim, Z. Xiong, and W. A. Pearlman, "Low bit-rate scalable video coding with 3-D set partitioning in hierarchical trees (3-D SPIHT)," *IEEE Trans. Cir. and Syst. for Video Technol.*, vol. 10, no. 8, pp. 1374-1387, Dec. 2000.
- [4] E. Khan and M. Ghanbari, "Wavelet-based video coding with early-predicted zerotrees," *IET Image Process.*, vol. 1, no. 1, pp. 95-102, Mar. 2007.
- [5] L. Zhang, D. Wang, and A. Vincent, "Decoupled 3-D zerotree structure for wavelet-based video coding," *IEEE Trans. Broadcasting*, vol. 54, no. 3, pp. 430-436, Jul. 2008.
- [6] C. Naveen, V. R. Satpute, A. G. Keskar, and K. D. Kulat, "Comparative analysis of video compression mechanisms based on EZW coded 3D-DWT and AVI," *IEEE Int. Symp. Signal Process. and Inf. Technol.*, pp. 96-101, Dec. 2014.
- [7] K. Cheng and J. Dill, "Lossless to lossy dual-tree BEZW compression for hyperspectral images," *IEEE Trans. Geoscience and Remote Sensing*, vol. 52, no. 9, pp. 5765-5770, Sept. 2014.
- [8] X. Tang, W. A. Pearlman, and J. W. Modestino,

- “Hyperspectral image compression using three-dimensional wavelet coding: a lossy-to-lossless solution,” *Image and Video Commun. and Process.*, pp. 1-22, Mar. 2004.
- [9] M. J. H. Loomans, C. J. Koeleman, and P. H. N. With, “Low-complexity wavelet based scalable image & video coding for home-use surveillance,” *IEEE Trans. Consumer Electron.*, vol. 57, no. 2, pp. 507-515, May 2011.
- [10] P. Chen and J. W. Woods, “Bidirectional MC-EZBC with lifting implementation,” *IEEE Trans. Cir. and Syst. for Video Technol.*, vol. 14, no. 10, pp. 1183-1194, Oct. 2004.
- [11] Y. Wu, K. Hanke, T. Ruser, and J. W. Woods, “Enhanced MC-EZBC scalable video coder,” *IEEE Trans. Cir. and Syst. for Video Technol.*, vol. 18, no. 10, pp. 1432-1436, Jun. 2008.
- [12] Y. Chen and G. Liu, “Improved overlapped block motion compensation using adaptive weighting matrices design for 3D wavelet-based scalable video coding,” *IEEE Int. Conf. Signal Process., Commun. and Computing*, pp. 1-6, Sept. 2015.
- [13] Y. Wu and J. W. Woods, “Scalable motion vector coding based on CABAC for MC-EZBC,” *IEEE Trans. Cir. and Syst. for Video Technol.*, vol. 17, no. 6, pp. 790-795, Jun. 2007.
- [14] A. Golwelkar and J. W. Woods, “Motion-compensated temporal filtering and motion vector coding using biorthogonal filters,” *IEEE Trans. Cir. and Syst. for Video Technol.*, vol. 17, no. 4, pp. 417-428, Apr. 2007.
- [15] C. Tillier, B. P. Popescu, and M. Schaar, “3-band motion-compensated temporal structures for scalable video coding,” *IEEE Trans. Image Process.*, vol. 15, no. 9, pp. 2545-2557, Sept. 2006.
- [16] J. Xu, Z. Xiong, S. Li, and Y. Q. Zhang, “Three-dimensional embedded subband coding with optimized truncation (3-D ESCOT),” *Applied and Computational Harmonic Anal.*, vol. 10, no. 3, pp. 290-315, May 2001.
- [17] E. Belyaev, K. Egiastian, and M. Gabbouj, “A low-complexity bit-plane entropy coding and rate control for 3-D DWT based video coding,” *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 1786-1799, Dec. 2013.
- [18] H. Xiong, L. Zhu, N. Ma, and Y. F. Zheng, “Scalable video compression framework with adaptive orientational multiresolution transform and nonuniform directional filterbank design,” *IEEE Trans. Cir. and Syst. for Video Technol.*, vol. 21, no. 8, pp. 1085- 1099, Aug. 2011.
- [19] B. B. Fradj and A. O. Zaid, “Scalable video coding using motion-compensated temporal filtering,” *Eur. Wksp. Visual Info. Process.*, pp. 50-55, Jul. 2011.
- [20] C. M. Yang and K. Chung, “Embedded image compression scheme using rate-distortion optimized block coding of wavelet coefficients,” *J. KICS*, vol. 39, no. 11, pp. 625-636, Nov. 2014.
- [21] J. E. Fowler and B. P. Popescu, “An overview on wavelets in source coding, communications and networks,” *EURASIP J. Image and Video Process.*, vol. 2007, pp. 1-27, Jun. 2007.
- [22] S. Zafar, Y. Zhang, and B. Jabbari, “Multi-scale video representation using multi-resolution motion compensation and wavelet decomposition,” *IEEE J. Sel. Areas in Commun.*, vol. 11, no. 1, pp. 24-35, Jan. 1993.
- [23] J. Heo, and C. Ryu, “Contents adaptive MCTF using JND,” *J. KICS*, vol. 34, no. 1, pp. 48-55, Jan. 2009.
- [24] D. Taubman, “High performance scalable image compression with EBCOT,” *IEEE Trans. Image Process.*, vol. 9, no. 7, pp. 1158 - 1170, Jul. 2000.
- [25] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Trans. Cir. and Syst. for Video Technol.*, vol. 13, no. 7, pp. 560-576, Jul. 2003.

양 창 모 (Chang Mo Yang)



1998년 2월 : 한국항공대학교 항공전자공학과 졸업
2000년 2월 : 광주과학기술원 정보통신공학과 석사
2002년 2월 : 광주과학기술원 정보통신공학과 박사수료
2002년 3월~현재 : 전자부품연구원 책임연구원

2012년 9월~현재 : 광운대학교 전자통신공학과 박사과정

<관심분야> 영상 압축, 실시간 비디오 스트리밍

정 광 수 (Kwangsue Chung)



1981년 2월 : 한양대학교 전자공학과 졸업
1983년 2월 : 한국과학기술원 전기 및 전자공학과 석사
1991년 2월 : University of Florida 전기공학과 박사
1983년 3월~1993년 2월 : 한국전자통신연구원 선임연구원

1993년 3월~현재 : 광운대학교 전자통신공학과 교수
<관심분야> 인터넷 QoS, 유·무선 비디오 스트리밍, 센서 네트워크