

## 블록 병합 기법을 이용한 객체 경계 부분 부호화

이희습(李熙習), 김정식(金正植), 김정우(金正祐), 이근영(李根泳)  
성균관대학교 전기전자컴퓨터 공학부  
전화 : (0331) 290-7193 / 팩스 : (0331) 290-7180

### Object Boundary Block Coding Using Block Merging Method

Heesup Lee, Jeongsig Kim, Jeongwoo Kim, Keunyoung Lee  
School of Electrical and Computer Engineering, Sungkyunkwan University  
E-mail : sky@mickey.skku.ac.kr

#### Abstract

Padding is a technique that enables applying conventional discrete cosine transform to encode boundary blocks of arbitrarily shaped objects by assigning imaginary values to the pixels that are not included in the object. Padding prevents the increase of high frequency DCT coefficients. However, in some boundary blocks, too many padded pixels are coded due to a small portion of object pixels. To reduce the number of padded pixels and to improve coding efficiency, we propose a block merging method for texture coding. The proposed method searches the shape information of boundary blocks and excludes the  $4 \times 4$  pixels of  $8 \times 8$  blocks if all the  $4 \times 4$  pixels are in the background region, and merges the remained  $4 \times 4$  pixels into new  $8 \times 8$  blocks. Experimental results show that our proposed method yields a rate-distortion gain about 0.5~1.6dB compared to conventional padding method, LPE.

#### I. 서론

임의의 모양을 가지고 있는 객체를 각각 부호화 하는 기법(Object Based Coding)은 MPEG-4[1,2]에 있어서

가장 주목할만한 특징들 중 하나라고 할 수 있다. 이러한 객체 기반 부호화를 실현시키기 위하여 MPEG-4에서는 기본적으로 두 가지 기법 즉, 모양 적응형 DCT(Shape Adaptive Discrete Cosine Transform : SA-DCT)[2,3]와 패딩(Padding)[3,4] 기법이 선택적으로 사용되고 있다. SA-DCT는 패딩에 비해 높은 화질을 얻을 수 있으나 기존의  $8 \times 8$  DCT를 그대로 사용할 수 없으므로 복잡도가 증가한다. 패딩 기법은 그에 비하여 화질은 떨어지나 기존의 DCT를 그대로 사용할 수 있는 장점이 있다. 패딩 기법은 경계 블록 내에서 객체 바깥영역에 해당되는 화소들에 임의의 적절한 값을 채워 줌으로써 고주파 영역의 DCT 화소 수를 줄이는데 그 목적이 있다. 그러나, 경계 블록 내에서 객체에 해당하는 화소들의 수가 적을 경우 오히려 그보다 더 많은 수의 객체 바깥영역에 해당되는 화소들이 패딩되어 부호화 됨으로써 효율을 저하시키는 단점이 있다.

본 논문에서는 패딩 기법의 효율성을 증가시킬 수 있는 전처리 기법으로써  $4 \times 4$  화소에 기반한 객체 경계 블록 병합 방법을 제안한다. 제안한 기법은 위의 단점을 극복하고자 먼저  $8 \times 8$  블록으로 나누어진 모양 정보를  $4 \times 4$  블록 단위로 검색하여 만일 모든 화소들이 객체 바깥 영역에 해당되는 블록이 검색되면 그 블록을 제외한 나머지 블록들로 다시  $8 \times 8$  블록을 재조합함으로써 패딩되는 경계 블록의 수를 줄여 경계 블록 부호화 효율을 높인다.

2장에서는 기존의 패딩 기법에 대해 간략히 살펴본 후 제안한 기법에 대해서 설명한다. 3장에서는 기존의 패딩 기법[2,3,4]과 실험결과를 비교한 후 4장에서 결론을 맺는다.

## II. 본론

### 1. MPEG-4에 적용되는 패딩 기법

MPEG-4에는 영상내(intraframe) 부호화를 위한 저주파 확장 패딩(Low Pass Extrapolation Padding: LPE Padding)과 영상간(interframe) 부호화를 위한 0값 패딩(Zero Padding)[3,4]이 있다.

저주파 확장 패딩에서는 객체의 경계 블록 내부에 있는 화소들의 평균값을 구한 후 객체 바깥에 해당되는 화소들을 그 평균값으로 채우고 난 후 객체 경계 블록의 바로 바깥에 위치한 화소에만 주위 4개 화소의 평균값으로 채우고 DCT를 수행한다. 이때 주위의 4개 화소 중 해당 8×8 블록 외부에 존재하는 것은 제외시킨다. 순서는 왼쪽 상단에서 수평방향으로 오른쪽 하단까지이다.

0값 패딩은 이와는 다르게 단순히 객체 경계 블록 바깥에 있는 모든 화소들을 0으로 채워준다. 이는 부호화에 사용되는 영상간 오차신호의 평균값이 일반적으로 0을 가진다는 성질을 이용한 것이다.

### 2. 제안한 객체 경계 영역 병합 기법

기존의 패딩 기법은 객체의 경계 영역 블록에서 객체 경계 외부에 속하는 화소들에게 임의의 적절한 값을 채워 줌으로써 고주파 영역의 DCT 화소 수를 줄이는데 그 목적이 있다. 그러나, 그림 1의 (a)와 같이 객체의 경계 영역 내부에 있는 화소의 수가 적을 경우 불필요한 화소들이 많이 패딩되어 부호화 되는 단점이 있다. 이러한 단점을 보완하기 위해 제안한 블록 병합 기법을 기존의 패딩 과정 전에 수행함으로써 부호화 효율을 증가시킬 수 있다.

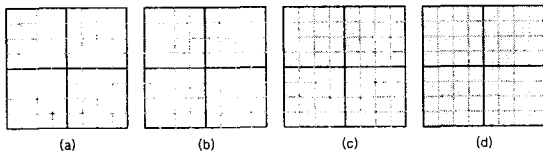


그림 1. 전형적인 물체 경계 블록의 유형

일반적으로 객체 경계 블록은 그림 1과 같이 4가지 유형으로 크게 나뉘어 질 수 있다. 제안한 방법에서는 (a), (b), (c)와 같이 객체 경계 영역 내부에서 비어있

는 4×4 블록이 있을 경우 인접 경계 블록을 순서대로 채워 넣음으로써 8×8 블록을 재구성한다. 예를 들면, 그림 1의 (a), (b), (c) 세 블록을 병합한 결과는 그림 2의 (b)와 같다.

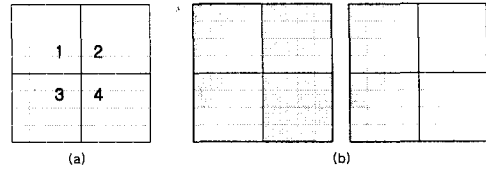


그림 2. 8×8 물체 경계 블록에서의 (a)검색, 병합 순서와 (b)에

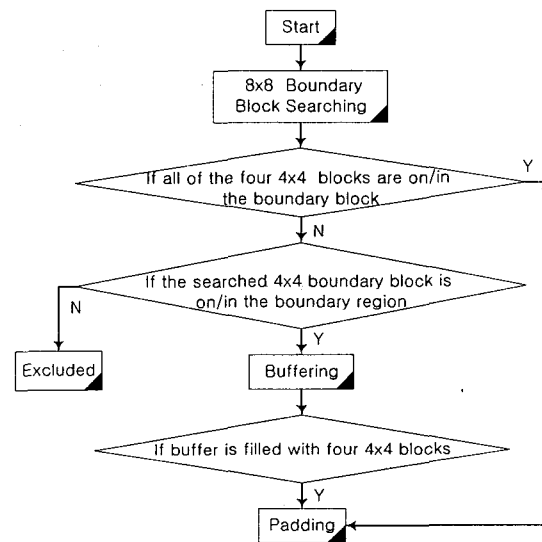


그림 3. 경계 블록 병합 과정의 흐름도

부호기에서 수행되는 전처리 과정을 단계별로 설명하면 다음과 같다.

단계 1. 8×8 블록으로 나누어진 모양 정보를 4×4 블록 단위로 그림 3의 (a)와 같은 순서로 검색하여 만일 그림 1의 (d)와 같이 4개의 4×4 블록 모두 경계 블록 영역에 해당되면 그 8×8 블록은 먼저 기존의 패딩 방법과 동일하게 영상내 부호화하면 저주파 확장 패딩을, 영상간 부호화이면 0값 패딩하고 DCT를 수행한다.

단계 2. 만일 모든 화소들이 객체 경계 외부에 속하는 4×4 블록이면 그 블록은 제외시키고 그 외의 4×4 블록은 순서대로 버퍼에 저장한다.

단계 3. 검색할 때와 같은 순서로, 버퍼에 저장된 4×4 블록들을 8×8 블록으로 재조합 한 후

단계 1과 같은 방법으로 패딩하고 DCT를 수행한다.

복호기에서는 전송되어진 모양 정보를 검색하여 부호기와는 역순으로 임의로 합병되어진 블록들을 다시 원래의 블록으로 재배치한다.

제안한 기법의 장점은 이 블록 병합 과정을 수행함에 있어서 기존의 모양정보를 이용하여 정해진 순서대로 검색하고, 병합한 후 재배치하므로 별도의 부가 정보가 필요 없고, 객체의 경계 블록에 포함되지 않는 4×4 화소들을 제거함으로써 부호화 되는 경계블록의 수를 줄여 부호화 효율을 높일 수 있다는데 있다.

### III. 실험 결과 및 고찰

제안된 블록 병합 전처리 기법의 성능을 알아보기 위하여 MPEG-4의 패딩 기법[1,2,4]을 사용하였다. 본 논문에서는 영상내 부호화시 제안된 블록 병합 기법을 사용하였을 때와 사용하지 않았을 때의 결과를 다양한 실험영상을 통하여 비교하였다. 실험 영상으로서는 "Weather"(QCIF, CIF, 300 frames) (그림4), "Children2"(QCIF, CIF, 300 frames)(그림 5)를 사용하였다.

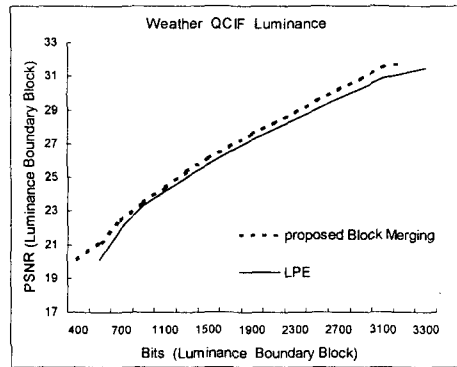


그림 4. Weather CIF, QCIF

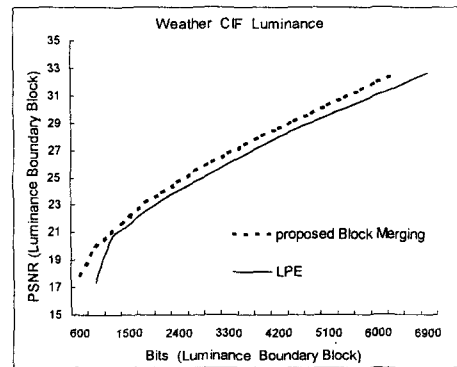


그림 5. Children2 CIF, QCIF

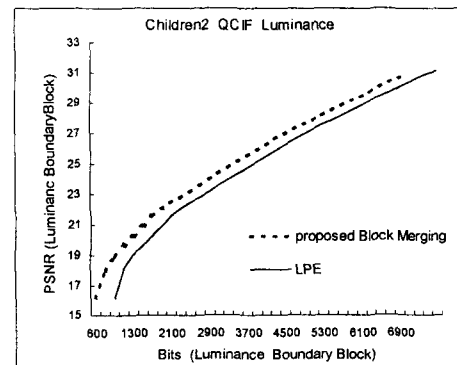
그림 6에서 영상들에 대한 실험 결과 R-D(Rate-distortion)곡선들을 나타내었는데 그 값들은 각각의 실험 영상들을 60 프레임씩 양자화 스텝 수를 가변해 가면서 영상내 부호화를 수행하여 구한 평균값이다. 수평축은 객체 경계 블록의 평균 비트 수를 나타내고 수직축은 평균 PSNR을 나타낸다. 실험결과(그림 6, 표 1)에 나타난 바와 같이, 실험에 사용한 두 영상들에 대해 제안한 블록 병합 기법을 수행함으로써 동일한 비트 수에서 PSNR이 0.5~1.6dB 정도 향상됨을 알 수 있다. 또한 CIF와 QCIF간의 영상 크기 변화에도 큰 영향 없이 동일한 정도로 PSNR이 향상(0.5~1.6dB)됨을 볼 수 있다.



(a)



(b)



(c)

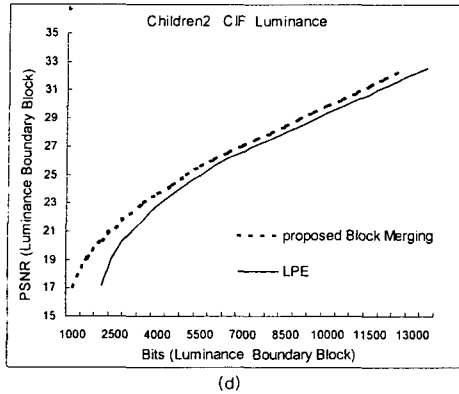


그림 6. 영상간 부호화시 물체 경계 부분에서의 R-D 곡선  
 (a) Weather QCIF Luminance  
 (b) Weather CIF Luminance  
 (c) Children2 QCIF Luminance  
 (d) Children2 CIF Luminance

표 1은 각각의 실험 영상에 대한 블록 분포를 나타낸 것인데, 제안한 블록 병합 기법을 사용함으로써 평균 26%~33%의 경계 블록 수를 줄일 수 있었다. 그러나, Weather에서의 객체 부분에 대한 경계 블록의 비율이 CIF에서는 17%, QCIF에서는 31%인데 비해 Children2의 예에서는 각각 47%와 73%로 제안한 기법의 효과가 더욱 크게 나타남을 알 수 있다. 이는 객체의 크기가 작으면 작을수록 제안한 알고리즘이 더욱 효율적임을 반영한다.

표 1. 실험 영상들의 블록 분포

Sequences	블록 수	객체 영역 전체	객체 경계 영역		
			블록 합병 전	블록 합병 후	병합된 블록의 비율
Weather	CIF	465	80	54	33%
	QCIF	126	39	28	28%
Children2	CIF	327	154	111	28%
	QCIF	102	74	51	31%

#### IV. 결론

본 논문에서는 MPEG-4에서 채택된 패딩 기법을 더욱 효과적으로 사용할 수 있는 블록 병합 기법을 제안하였다. 제안한 기법에서는 부호기와 복호기에서 동일한 객체 모양 정보를 이용하므로 별도의 부가 정보가 필요 없고, 객체 경계 영역 내부에 포함되지 않는 4x4 화소 수를 최대한 줄여 부호화 경계 블록의 수를 줄일 수 있었다. 실험 결과를 살펴보면, 그 효과는

두 실험 영상의 종류와 크기에 거의 무관하였으나 영상내 객체의 크기가 작을수록 더욱 효율적이었다. 제안한 기법은 영상내 부호화시 기존의 저주파 확장 패딩을 단독으로 실시하는 것 보다 0.5~1.6dB 정도의 PSNR이 향상되었고, 경계 블록의 수는 영상에 따라 28~33% 정도 줄일 수 있었다.

본 논문에서는 영상내 부호화의 저주파 확장 패딩의 경우만 실험하였으나 근본적으로 저주파 확장 패딩과 영상간 부호화의 0값 패딩간에는 차이가 없으므로 제안된 기법과 0값 패딩을 결합한다면 영상내 부호화의 결과와 마찬가지로 좋은 효과를 얻을 수 있으리라 예상된다.

#### 참고문헌

- [1] ISO/IEC 14496-2 Proposed Draft Amendment 1, "Information technology-Generic coding of audio -visual objects-Part 2 : Visual," March 1999.
- [2] ISO/IEC JTC1/SC29/WG11, "MPEG-4 video verification model version 11.0," Doc. N2172, March 1998.
- [3] T. Sikora, S. Bauer and B. Makai, "Efficiency of Shape-Adaptive Transforms for coding of Arbitrarily Shaped Image Segments," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 5, No. 3, June 1995.
- [4] André Kaup, "Object-Based Texture Coding of Moving Video in MPEG-4," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 9, NO. 1, February 1999.
- [5] André Kaup, "Adaptive low-pass extrapolation for object-based texture coding of moving video," in Proc. Visual Communication Image Processing '97, SPIE, Vol. 3024, February 1997.