

H.264/AVC 무손실 부호화를 위한 향상된 CABAC 부호화 방법

김승환, 호요성

광주과학기술원 정보통신공학과

kshkim@gist.ac.kr, hoyo@gist.ac.kr

I. 서론

H.264/AVC 비디오 부호화 표준은 손실(lossy) 압축 기능과 무손실(lossless) 압축 기능 모두를 제공한다. 하지만, H.264/AVC는 주로 손실(lossy) 압축 기반으로 연구되어왔기 때문에 무손실 압축의 경우 부호화 효율이 비교적 낮다. 현재 H.264/AVC에 구현된 인트라 화면의 무손실 압축의 경우 원 영상과 예측된 영상의 차분인 잔여 영상 신호에 대하여 변환 (Transform)과 양자화(Quantization)과정을 수행하지 않고 엔트로피 부호화 과정만을 수행한다.

본 논문에서는 무손실 압축환경에서의 잔여 신호의 통계적인 특성을 고려하여 이를 효율적으로 부호화할 수 있는 향상된 CABAC을 설계하는 방법을 제안한다. 그림 1은 H.264/AVC 부호화 표준에서 매크로 블록 부호화에 사용되는 구문을 나타낸다. 그림 1에서 회색으로 처리된 구문들은 잔여 신호를 부호화 하기 위해 사용되는 구문이다. 본 논문에서는 회색으로 처리된 구문들에 대한 부호화 과정과 방법을 새롭게 제안했다.

<i>mb_type</i>
<i>mb_qp_delta</i>
<i>intra_chrom_pred_mode</i>
<i>rem_intra4x4_pred_mode_flag</i>
<i>mb_field_decoding_flag</i>
<i>coded_block_pattern</i>
<i>coded_block_flag</i>
<i>significant_coeff_flag</i>
<i>last_significant_coeff_flag</i>
<i>coeff_ahs_level_minus1</i>

그림 1. 매크로 블록부호화에 사용되는 구문

II. 제안한 방법

그림 2에서는 제안한 CABAC 기반의 잔여 신호 부호화 방법을 잘 나타내고 있다. 제안한 방법에서는 각 블록내 모든 계수의 절대값이 0 인 All_zero 블록을 부호화하기 위해 coded_block_flag 대신 significance Map 부호화 과정을 변경하여 이를 부호화했으며 Significance Map 정보를 부호화 하기 위해 significant_coeff_flag 만을 이용하여 부호화 하도록 새롭게 제안했다. 또한, 레벨 무손실 부호화에서 레벨의 통계 분포에 맞도록 새로운 이진화 방법도 제안했다.

1. 잔여 신호 부호화

주어진 하위 블록 내 계수들의 절대 값이 모두 0 인 All_zero 블록의 발생 확률은 양자화 계수의 크기에 비례한다. 무손실 압축 부호화 경우 All_zero 블록의 발생 확률이 매우 낮으므로 coded_block_flag 를 부호화 하지 않고 항상 coded_block_flag =1 이라 가정한다. 그

리고 Significance Map 을 부호화할 때 그림 2에 명시된 대로 All_zero 블록도 함께 부호화한다.

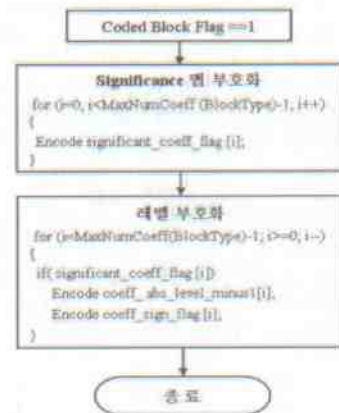


그림 2. 제안하는 잔여 신호 부호화 과정

2. Significance Map 부호화

무손실 압축의 경우 last_significant_coeff_flag가 대부분 스캔의 마지막 부분에 분포하므로 이를 사용하는 것은 무손실 압축의 부호화 효율에 많은 저하를 초래하게 된다. 따라서, 본 논문에서는 블록 내 모든 스캔 위치에서 significant_coeff_flag만 이용하여 부호화 하도록 새롭게 제안했다.

3. 레벨(level) 부호화

무손실 압축의 경우 '0' 이 아닌 레벨 값들의 분포가 손실 압축과는 많은 차이가 있음을 고려하여 기존의 'cut-off' 값을 14에서 9로 바꾸어 무손실 부호화에서 레벨의 통계 분포에 맞도록 새로운 이진화 방법을 제안했다.

IV. 결론

본 논문에서는 H.264/AVC에서 손실 압축과 무손실 압축 환경에서의 잔여 신호의 통계적인 특성의 차이를 분석했다. 이를 기반으로, 무손실 압축 환경에 적합한 향상된 CABAC기반의 잔여 신호 부호화 방법을 제안했다. 제안한 방법에서는 잔여신호의 효율적인 부호화를 위한 significance map 부호화 방법과 새로운 이진화 방법을 제안했다. 실험 결과로부터 제안한 방법이 무손실 부호화 압축의 경우 현재 H.264/AVC에 사용되고 있는 CABAC에 비해 평균적으로 약 16% 비트 수를 감소시키는 것을 확인했다.

H.264/AVC 무손실 부호화를 위한 향상된 CABAC 부호화 방법

김승환, 호요성
광주과학기술원 정보통신공학과

Improved CABAC for H.264/AVC Lossless Coding

Seung-Hwan, Yo-Sung Ho
Gwangju Institute of Science and Technology
kshkim@gist.ac.kr, hoyo@gist.ac.kr

Abstract: H.264/AVC 비디오 압축 표준에서 엔트로피 부호화 방법 중 하나인 문맥 적응적 이진 산술 부호화(CABAC: context-adaptive binary arithmetic coding)는 주로 손실(lossy) 압축기반의 부호화 환경에 맞추어 설계 되었다. 또한 H.264/AVC 고급 프로파일(Advanced 4:4:4)에서 지원하는 무손실(lossless) 부호화의 경우 손실 압축기반으로 만들어진 CABAC 이 그대로 사용되므로 최적의 압축성능을 제공하지 못하고 있다. 본 논문에서는 H.264/AVC 에서 손실 압축과 무손실 압축환경에서의 잔여 신호의 통계적인 특성에 분명한 차이가 있음을 보인다. 또한, 무손실 압축 환경에서 잔여 신호의 통계적 특성을 고려하여 향상된 CABAC 기반의 잔여 신호 부호화 방법을 제안한다. 실험 결과로부터 본 논문에서 제안하는 향상된 CABAC 부호화 방법이 무손실 부호화 압축의 경우 기존의 CABAC 에 비해 평균적으로 약 16% 비트 수를 감소시키는 것을 확인했다.

Keywords: H.264/AVC, CABAC, 무손실 부호화

I. 서론

H.264/AVC 비디오 부호화 표준은 국제전기 통신 연합(International Telecommunication Union, ITU) 산하의 ITU-T와 국제 표준화 기구(International Standard Organization) 산하의 MPEG (Motion Picture Expert Group) 두 기관의 공동 작업으로 최근에 완성되었다. H.264/AVC는 이전의 비디오 부호화 표준인 MPEG-2/4, H.263 등에 비해 높은 부호화 효율을 제공하는데, 이는 가변 블록기반의 움직임 보상, 복수 참조 영상, 1/4 화소 단위 부화소 움직임 예측, 문맥 적응적 엔트로피 부호화인 CABAC (Context-based Adaptive Binary Arithmetic Coding)과 같은 부호화 효율이 좋은 기술들이 사용되기 때문이다 [1] [2].

H.264/AVC 비디오 부호화 표준은 손실(lossy) 압축 기능과 무손실(lossless) 압축 기능 모두를 제공한다. 하지만, H.264/AVC는 주로 손실(lossy) 압축 기반으로 연구되어왔기 때문에 기존의 무손실 압축 기반으로 연구된 JPEG-LS, JPEG 2000에 구현된 무손실 알고리즘에 비해 비교적 낮다. 따라서 H.264/AVC 기반의 무손실 압축 성능향상에 대한 많은 연구가 필요하다. 현재

H.264/AVC에 구현된 인트라 화면의 무손실 압축의 경우 원 영상과 예측된 영상의 차분인 잔여 영상 신호에 대하여 변환 (Transform)과 양자화(Quantization)과정을 수행하지 않고 엔트로피 부호화 과정만을 수행한다.

따라서, 무손실 압축의 잔여 영상 신호와 손실 압축의 잔여 영상 신호간에는 명확한 통계적 분포의 차이가 있다. 본 논문에서는 무 손실 압축환경에서의 잔여 신호의 통계적인 특성을 고려하여 이를 효율적으로 부호화할 수 있는 향상된 CABAC을 설계하는 방법을 제안한다. 그림 1은 H.264/AVC 부호화 표준에서 매크로블록 부호화에 사용되는 구문을 나타낸다. 그림 1에서 회색으로 처리된 구문들은 잔여 신호를 부호화 하기 위해 사용되는 구문이다. 따라서, 본 논문에서는 회색으로 처리된 구문들에 대한 부호화 과정과 방법을 새롭게 제안했다.

<i>mb_type</i>
<i>mb_qp_delta</i>
<i>intra_chrom_pred_mode</i>
<i>rem_intra4x4_pred_mode_flag</i>
<i>mb_field_decoding_flag</i>
<i>coded_block_pattern</i>
<i>coded_block_flag</i>
<i>significant_coeff_flag</i>
<i>last_significant_coeff_flag</i>
<i>coeff_abs_level_minus1</i>

그림 1. 매크로 블록부호화에 사용되는 구문

II. CABAC 에서의 잔여 영상 부호화

그림 2에서는 주어진 매크로블록내 하위 블록(4×4 또는 8×8)에 대한 잔여 영상 부호화 과정을 나타내고 있다. 그림 2에 명시된 coded block flag 는 현재 블록이 '0'이 아닌 계수들을 가지고 있다는 것을 나타내는 일종의 플래그(flag)와 같다. 만약 coded block flag 가 '0'이면, 주어진 블록에 대해 전송할 계수가 존재하지 않다는 것을 의미한다. 따라서 coded block flag 가 '0'이 아닌 경우에만 significance map 부호화 과정과 레벨 부호화 과정을 통해 '0'이 아닌 계수들을 부호화한다. 이때,

significant_coeff_flag 는 각 지그재그 스캔위치에서 해당 계수의 절대 값이 '0'인지 아닌지를 나타내며 last_significant_coeff_flag 는 해당 지그재그 스캔 위치 이후에 '0'이 아닌 계수 값이 존재하는가를 나타낸다. 따라서 last_significant_coeff_flag =1 인 경우에 significance map 부호화가 종료된다.

레벨 부호화 과정에서는 significance map 부호화 과정을 통해 이미 0 이 아닌 계수들의 위치를 알고 있으므로, 해당 계수들에 대한 절대값과 부호를 부호화한다. CABAC 에서는 이진화된 심볼만이 산술 부호화를 통해 부호화되도록 설계되었으므로 계수의 절대값을 UEG0(0-th order Unary Exponential Golomb code) 이진화 과정을 통해 이진화한다. 이때, 각 계수들의 절대 값들은 역으로 지그재그스캔 된 순서에 따라 전송된다 [3].

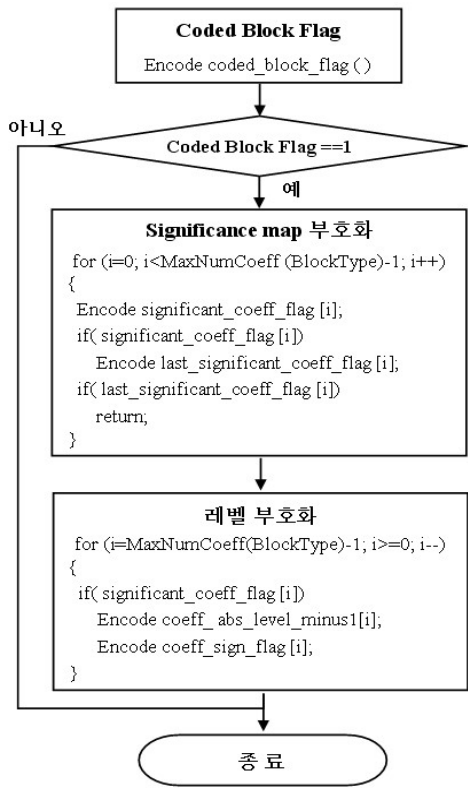


그림 2. 잔여신호 부호화 구조 (CABAC)

III. 제안한 방법

그림 3 에서는 제안한 CABAC 기반의 잔여 신호 부호화 방법을 잘 나타내고 있다. 제안한 방법에서는 각 블록내 모든 계수의 절대값이 '0'인 'All_zero' 블록을 부호화하기 위해 coded_block_flag 대신 'significance Map' 부호화 과정을 변경하여 이를 부호화 했으며 Significance Map 정보를 부호화 하기 위해 significant_coeff_flag 만을 이용하여 부호화 하도록 새롭게 제안했다. 또한, 레벨 무손실 부호화에서 레벨의 통계 분포에 맞도록 새로운 이진화 방법도 제안했다.

표 1. 양자화 계수에 따른 'All-zero'블록의 발생 확률

영상 \ QP	32	24	12	무손실
Paris	0.2669	0.1375	0.0165	0.0004
Mobile	0.2064	0.1084	0.0197	0.0001
Tempete	0.2048	0.1271	0.0062	0.0001
Foreman	0.2895	0.1492	0.0369	0.0001
Silent	0.2067	0.0998	0.0219	0.0011

1. 잔여 신호 부호화

주어진 하위 블록 내 계수들의 절대 값이 모두 '0' 인 'All_zero'블록의 발생 확률은 양자화 계수의 크기에 비례한다. 표 1에서는 양자화 계수에 따른 'All_zero'블록의 발생 확률 나타내었다. 표 1에서 알 수 있듯이 무손실 압축 부호화 경우 'All_zero' 블록의 발생 확률이 매우 낮으므로 coded_block_flag를 부호화하지 않고 항상 coded_block_flag =1이라 가정한다. 그리고 Significance Map을 부호화할 때 그림 3에 명시된 대로 'All_zero'블록도 함께 부호화한다.

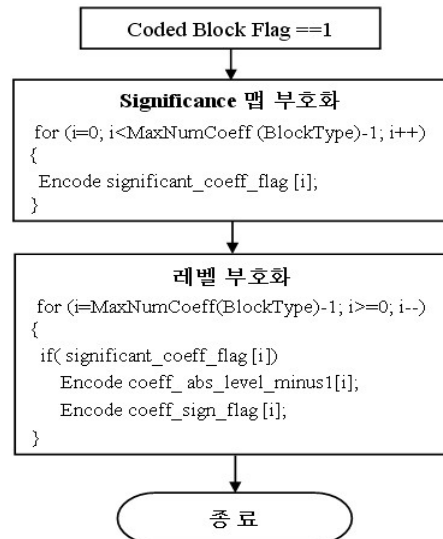


그림 3. 제안하는 잔여 신호 부호화 과정

2. Significance Map 부호화

주어진 하위 블록 내 '0'이 아닌 계수들의 위치를 부호화 하기 위해 사용되는 'significance map' 을 효율적으로 부호화 하기 위해 기존에 사용하던 significant_coeff_flag와 last_significant_coeff_flag를 동시에 사용했다. 하지만, 그림 4에 나타낸 바와 같이 무손실 압축의 경우 last_significant_coeff_flag가 대부분 스캔의 마지막 부분에 분포하므로 이를 사용하는 것은 무손실 압축의 부호화 효율에 많은 저하를 초래하게 된다. 따라서, 본 논문에서는 블록 내 모든 스캔 위치에서 significant_coeff_flag만 이용하여 부호화 하도록 새롭게 제안했다.

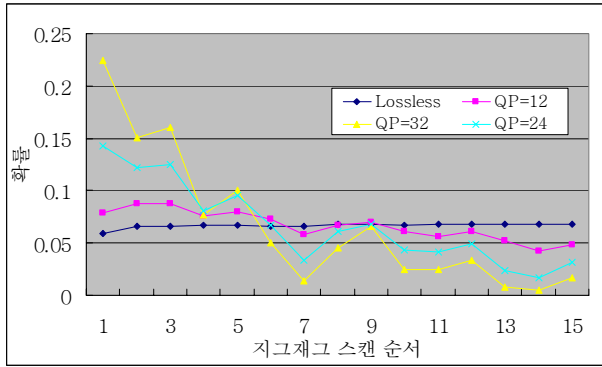


그림 4. 지그재그 스캔 순서와 계수 존재 확률

3. 레벨(level) 부호화

그림 5에 나타낸 바와 같이 무손실 압축의 경우 부호화할 각 계수들의 레벨 분포가 손실 압축과는 많은 차이가 있다. 무손실 압축의 경우 '0'이 아닌 레벨 값들의 분포가 손실 압축과는 많은 차이가 있음을 고려하여 기존의 'cut-off' 값을 14에서 9로 바꾸어 무손실 부호화에서 레벨의 통계 분포에 맞도록 새로운 이진화 방법을 제안했다.

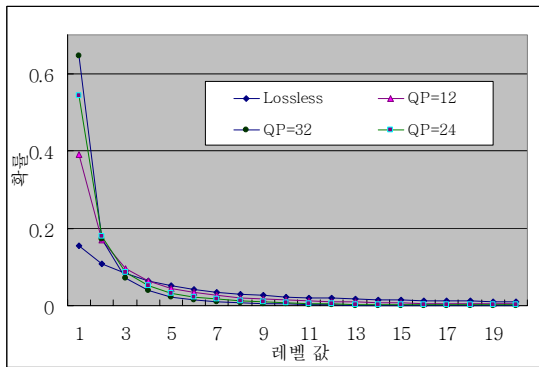


그림 5. 양자화 계수에 따른 레벨의 확률 분포

표 2. 제안한 UEGO 이진화 (cut-off S=9)

레벨 값 (abs_level)	이진 열 (Bin string)	
	TU prefix	EGO suffix
1	0	
2	10	
3	110	
...	...	
9	111111110	
10	111111110	0
11	111111110	100
12	111111110	101
...

4. 제안하는 문맥 모델링

위에서 이미 언급한 대로 무손실 압축의 경우에 잔여 영상의 통계적 분포가 손실 압축과 다르며 CABAC에서 각 이진화된 신택스들을 부호화하기 위해 사용하는 문맥 모델링의 경우에도 새로운 통계적 분포를 고려해서 결정해야 한다. 따라서, 본 논문에서는 주어진 신택스 요소의 이진화된 각 이진수와 유일하게 대응되는 문맥 지표 (Context Index)를 얻기 위하여 다음과 같은 문맥 모델링 방법을 제안한다.

$$\gamma = \Gamma_s + \Delta_s(ctx_cat) \tag{1}$$

식 (1)에서 γ 는 각 이진(bin) 심볼과 대응되는 문맥 지표를 나타낸다. Γ_s 는 각 신택스 요소에 대한 문맥 지표 오프셋(Context Index Offset)을 의미하며, $\Delta_s(ctx_cat)$ 는 잔여 영상 데이터의 경우에 해당 블록의 타입에 따라 문맥 별도의 문맥 지표를 제공하기 위해 사용되는 문맥 목록(Context Category)을 나타낸다. 블록 타입에 따른 문맥 목록의 자세한 내용은 아래 표 3에 나타내었다.

표 3. 블록 타입에 따른 문맥 목록(context category)

블록 형태(Type)	MaxNumCoeff	$\Delta_s(ctx_cat)$
휘도DC블록:I16x16	16	0
휘도AC블록:I16x16	15	1
휘도블록-I4x4	16	2
u-색차DC블록	4	3
v-색차DC블록	4	
u-색차AC블록	4	4
v-색차AC블록	4	
휘도블록:I8x8	64	5

IV. 실험 결과

본 논문은 H.264 표준 소프트웨어 JM 13.2 [4]를 이용하여 실험했다. 제안한 알고리즘의 효율성을 보이고자, 4개의 QCIF (176×144) 해상도의 비디오 영상들(News, Container, Foreman, Silent)과 3개의 CIF (352×288) 해상도의 비디오 영상들(Paris, Mobile, Tempete)을 이용하여 인트라 환경에서 부호화했다. 자세한 실험 환경 설정은 표 4에 나타내었다.

표 4에서 'QPPrimeYZeroTransformByPassFlag'는 변환과 양자화 과정을 생략하고 곧 바로 엔트로피 부호화 과정을 통해 잔여 영상 신호를 부호화하고자 하는 무손실 부호화를 위한 설정변수이다. 단 위 설정 변수는 양자화 계수 값을 0으로 사용했을 경우에만 유효하게 된다. 현재 실험을 위해 사용된 H.264 공개 소프트웨어 JM13.2에는 차분 펄스 변조 방식에 기초한 인트라 예측 방식이 구현되어 있다[5][6].

표 4. 실험 환경 및 변수 설정

환경 변수	설정 값
ProfileIDC (프로파일)	244 (High 4:4:4)
IntraPeriod (인트라 화면 주기)	0
QPISlice (양자화 계수)	0
SymbolMode (엔트로피)	1
QPPrimeYZeroTransformBypasFlag	1

제안한 방법과 H.264 참조 소프트웨어 방법의 성능 비교를 위해 압축 비와 비트 수 이득을 비교했다. 표 5는 성능 비교를 위해 실험한 결과를 나타낸다. 또한, 제안한 잔여신호 부호화 방법들 각각의 효율성을 비교하고자 다음과 같이 각각의 성능을 비교했다.

- 방법 1: 제안한 significance map 부호화
- 방법 2: 방법 1 + 제안한 문맥 모델링
- 방법 3: 방법 2 + 제안한 계수 레벨 이진화

표 5. 제안한 알고리즘 성능

영상	방법	비트 수	압축 비	비트 수 이득(%)
News	H.264	41708864	2.1875062	0
	방법 1	37064736	2.4615958	11.13463
	방법 2	36505968	2.4992735	12.47431
	방법 3	35798656	2.5486543	14.17014
Contain.	H.264	43593416	2.0929399	0
	방법 1	38806632	2.3511032	10.98052
	방법 2	38269512	2.3841014	12.21263
	방법 3	36544168	2.4966610	16.17044
Foreman	H.264	42082632	2.1680773	0
	방법 1	37336112	2.4437038	11.27904
	방법 2	36829000	2.4773520	12.48408
	방법 3	36954120	2.4689642	12.18676
Silent	H.264	48389729	1.8854910	0
	방법 1	43200520	2.1119745	10.72378
	방법 2	42819376	2.1307736	11.51143
	방법 3	40594920	2.2475324	16.10839
Paris	H.264	195598960	1.8658258	0
	방법 1	175018320	2.0852308	10.52185
	방법 2	172658856	2.1137265	11.72813
	방법 3	160943952	2.2675819	17.71737
Mobile	H.264	285134272	1.2799359	0
	방법 1	263217832	1.3865078	7.68635
	방법 2	258082984	1.4140940	9.48721
	방법 3	229797288	1.5881545	19.40734
Tempete	H.264	202624344	1.5609828	0
	방법 1	184422088	1.7150500	8.98325
	방법 2	181594104	1.7417587	10.37893
	방법 3	165366376	1.9126809	18.38770
평균	H.264		1.8629655	0
	방법 1		2.0793094	10.18760
	방법 2		2.1087257	11.46810
	방법 3		2.2186042	16.30687

표 5에 제시된 실험 결과로부터, 제안한 방법이 기존의 H.264 방법과 비교하여 평균적으로 약 16% 비트 수를 감소시킴을 알 수 있다. 다음은 표 5에서 비교한 비트 수 이득과 압축비에 관한 식이다.

$$\text{비트수 이득} = \frac{\text{비트수}(H.264/AVC) - \text{비트수}(\text{해당알고리즘})}{\text{비트수}(H.264/AVC)} \times 100\% \quad (2)$$

$$\text{압축비} = \frac{\text{비트 수(원영상)}}{\text{비트 수(해당알고리즘)}} \quad (3)$$

V. 결론

본 논문에서는 H.264/AVC에서 손실 압축과 무손실 압축 환경에서의 잔여 신호의 통계적인 특성의 차이를 분석했다. 이를 기반으로, 무손실 압축 환경에 적절한 향상된 CABAC기반의 잔여 신호 부호화 방법을 제안했다. 제안한 방법에서는 All_zero 블록들의 효율적인 부호화를 위해 coded_block_flag를 제거한 새로운 significance map 부호화 방법과 0이 아닌 계수들의 절대값을 효과적으로 부호화 하기 위해 새로운 이진화 방법을 제안했다. 마지막으로, 무손실 압축 잔여 데이터 특성에 맞게 단순화된 문맥 모델링을 제안했다. 실험 결과로부터 본 논문에서 제안하는 향상된 CABAC 부호화 방법이 무손실 부호화 압축의 경우 현재 H.264/AVC에 사용되고 있는 CABAC에 비해 평균적으로 약 16% 비트 수를 감소시키는 것을 확인했다.

감사의 글

본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터(ITRC)의 지원에 의한 것입니다.

참고문헌

- [1] ITU-T Recommendation H.264 and ISO/IEC 14496-10, Advanced Video Coding for Generic Audiovisual Services, May 2003
- [2] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. CSVT*, vol. 13, no. 7, pp. 560-576, July 2003
- [3] D. Marpe, H. Schwarz, and T. Wiegand, "Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression," *IEEE Trans. CSVT*, vol. 13, no. 7, pp. 620-636, July 2003
- [4] JVT 참조 소프트웨어, available online at: <http://iphome.hhi.de/suehring/tml/download/>
- [5] Y. L. Lee, K. H. Han, and S. C. Lim, Lossless Intra Coding for Improved 4:4:4 Coding in H.264/MPEG-4 AVC, ISO/IEC JTC1/SC29/WG11 and ITU-T Q6/SG16 Joint Video Team document JVT-P016, July 2005
- [6] Y. L. Lee, K. H. Han, and G. J. Sullivan, "Improved lossless intra coding for H.264/MPEG-4 AVC," *IEEE Transactions on Image Processing*, vol. 15, no. 9, pp. 2610-2615, Sept. 2006