

High Efficient Entropy Coding For Edge Image Compression

Jong-Woo Han*, Do-Hyun Kim**, Yoon Kim***

Abstract

In this paper, we analyse the characteristics of the edge image and propose a new entropy coding optimized to the compression of the edge image. The pixel values of the edge image have the Gaussian distribution around '0', and most of the pixel values are '0'. By using this analysis, the Zero Block technique is utilized in spatial domain. And the Intra Prediction Mode of the edge image is similar to the mode of the surrounding blocks or likely to be the Planar Mode or the Horizontal Mode. In this paper, we make use of the MPM technique that produces the Intra Prediction Mode with high probability modes. By utilizing the above properties, we design a new entropy coding method that is suitable for edge image and perform the compression. In case the existing compression techniques are applied to edge image, compression ratio is low and the algorithm is complicated as more than necessity and the running time is very long, because those techniques are based on the natural images. However, the compression ratio and the running time of the proposed technique is high and very short, respectively, because the proposed algorithm is optimized to the compression of the edge image. Experimental results indicate that the proposed algorithm provides better visual and PSNR performance up to 11 times than the JPEG.

▶Keyword: Edge Image, Entropy Coding, Compression, Context Based Adaptive Binary Arithmetic Coding

I. Introduction

현대 사회는 정보의 취득, 저장, 전송 및 재생이 가능한 시대이다. 또한 IT 기술이 발전하면서 영상은 고해상도와 고화질로 발전하였고 영상의 실제 데이터양은 거대화되었다.

해상도가 1280 x 720 일 때 RGB 형태로 화소가 양자화 되고 각각 8bits로 양자화 된다고 가정하면 데이터양은 $1280 \times 720 \times 3 \times 8 \text{ bit} \approx 22,118,400 \text{ bit}$, 약 2.76 Mbyte가 된다. 기술이 발전하며 저장 및 전송의 용량이 급속히 발전하고 있지만, 여전히 데이터양을 줄이는 것은 중요하다. 이러한 문제를

해결하기 위해 다양한 영상 압축 기술들이 개발되었다.

영상 압축이란 영상을 표현하는데 쓰이는 데이터의 양을 줄이는 것을 말한다. 영상 압축은 방법에 따라 손실 압축과 무손실 압축으로 구별된다. 무손실 압축은 압축 영상을 복원한 영상이 원래 영상과 완전히 일치하는 방법인 반면, 손실 압축은 덜 중요한 부분의 정보를 제거함으로써 압축률을 높이는 압축 방법이다[1-3].

• First Author: Jong-Woo Han, Corresponding Author: Yoon Kim

*Jong-Woo Han(jongwu90@naver.com), Dept. of Computer and Communications Engineering, Kangwon National University

**Do-Hyun Kim(abc12369874@naver.com), Dept. of Computer and Communications Engineering, Kangwon National University

***Yoon Kim(yooni@kangwon.ac.kr), Dept. of Computer and Communications Engineering, Kangwon National University

• Received: 2016. 04. 05, Revised: 2016. 04. 14, Accepted: 2016. 04. 27.

• This study is supported by 2015 Research Grant from Kangwon National University (No. 520150468).

Table 1. Size information of format

Format	Size	Colour Depth	Frame Size
1080p	1920 x 1080	24	6.2MB
2k	2048 x 1080	24	10MB
SHD	3840 x 2160	36	25MB
4k	4096 x 2160	36	40MB
8k	7680 x 4320	36	150MB

대표적으로 많이 사용되는 영상 압축 방식인 JPEG은 24비트의 수백만 가지 색상을 사용할 수 있어 다양한 디자인 및 사진을 압축하는데 효과적인 방식이지만 문자나 선 등 고주파 성분이 많은 이미지의 변환 과정에서 영상의 품질이 저하되는 단점이 있다[4-6]. 이러한 현상을 줄이기 위해 연구된 것이 H.264와 HEVC이다. H.264는 고주파 성분의 손실로 영상의 품질이 저하되는 단점을 줄이기 위해 복잡한 알고리즘을 사용하였다. 따라서 영상의 품질 저하는 줄였지만 알고리즘의 복잡도가 증가하여 알고리즘의 수행 시간이 증가하는 단점이 있다 [7]. HEVC는 H.264보다 더 복잡한 알고리즘을 사용하였다. 따라서 H.264보다 더 좋은 영상의 품질을 보이지만 알고리즘의 복잡도가 증가하여 H.264보다 더 긴 알고리즘의 수행 시간이 걸린다[8].

영상 처리 분야에서 에지 영상은 중요한 의미를 갖는다. 에지 영상이란 일반 영상의 에지들이 모여서 하나의 영상을 이루는 것을 말한다. 영상에서 에지는 영상의 명암, 컬러 등 특성이 급격히 변하는 지점으로 영상에서 중요한 정보를 가지고 있다. 따라서 서로 다른 물체 사이의 에지를 검출하는 것은 매우 유용하다. 에지 정보는 물체의 위치나 모양, 크기 등을 알 수 있으며 이러한 정보를 이용하여 물체를 검출 또는 인식하는데 기초가 된다. 추출된 에지는 도형 분리, 복잡한 인식, 레이블링 등 다양한 영상처리에 활용되며 여러 분야에서 응용이 가능하다. 따라서 에지 맵은 영상에서 매우 중요하며, 효율적으로 관리할 필요가 있다. 또한 에지 영상은 이진 영상으로 본래의 영상에 비해 필요한 메모리가 훨씬 적다.

본 논문에서는 고해상도의 에지 영상을 빠르고 효율적으로 압축하기 위하여 에지의 특성을 분석하고 특성에 적합한 엔트로피 코딩(Entropy Coding)을 제안한다. 지금까지의 압축 방법은 자연 영상을 기반으로 개발되어 에지 영상을 압축할 경우 기대되는 압축률도 낮으며 필요 이상으로 복잡하기 때문에 수행시간이 길다. 따라서 에지 영상의 특성에 맞는 새로운 압축 알고리즘이 필요하다. 에지의 픽셀 값은 '0'을 기준으로 가우시안 분포를 이루며 대부분의 값이 '0'을 갖는다. 이를 이용하여 공간영역에서 zero block 기법을 사용한다. 또한, 에지 영상의 화면 내 예측 모드는 주변 블록과 유사거나 Planar 모드, 혹은 수평 모드가 나올 가능성이 크다. 본 논문에서는 가능성이 높은 모드를 이용하여 예측모드를 만드는 MPM 기법을 사용한다. 위의 특성을 이용한 엔트로피 코딩을 설계하여 에지 영상에 최적인 압축을 수행한다.

본 논문의 구성은 다음과 같다. 2장에서는 엔트로피 코딩 방

법에 대한 전체적인 구성에 대하여 기술하고, 3장에서는 제한한 알고리즘을 기술한다. 4장에서는 기존의 알고리즘과 제한한 알고리즘을 비교하여 알고리즘의 우수성을 입증한다. 최종적으로 5장에서는 결론을 맺는다.

II. Entropy Coding

1. Outline Of The Entropy Coding

엔트로피 코딩은 무손실 압축 기법으로 심볼의 발생 확률을 바탕으로 동작한다. 엔트로피 코딩 중 자주 쓰이는 방법 두 가지는 가변 길이 코딩(VLC : Variable Length Coding)과 산술 부호화이다. 가변 길이 코딩은 심볼을 발생 확률에 따라 길이가 다른 코드 워드로 매핑하는 방법이며 산술 부호화는 하나의 실수 값으로 여러 개의 심볼을 표현하는 방법이다. 산술 코딩은 여러 개의 심볼을 묶어 부호화하는 방법을 사용하여 가변길이 코딩의 한계를 극복한다.

2. Arithmetic Coding

일반적인 산술 코딩에서 범위의 초기 값은 0 ~ 1이며 범위는 심볼의 발생 확률에 따라 구간으로 나누어진다. 현재 부호화할 심볼에 할당된 구간을 선택하고, 선택된 구간이 다음 부호화 과정에서의 범위가 된다. 이러한 과정을 반복하며 최종적으로 범위에 해당하는 값이 산술 코딩의 출력이 된다.

3. Binary Arithmetic Coding

다양한 심볼이 존재할 수 있는 일반적 산술 코딩과 달리 이진 산술 코딩은 '0' 또는 '1', 2개의 심볼만이 존재한다. 따라서 심볼에 대한 확률은 2개만이 존재하며 부/복호화 시 범위를 나누는 간격도 항상 2개만이 존재한다. 그림 1은 이진 산술 부호화의 과정을 보여준다.

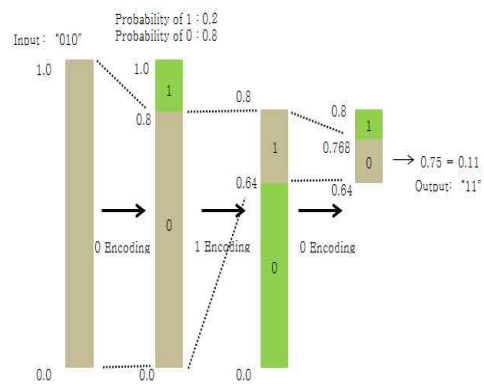


Fig. 1. Binary Arithmetic Coding

4. Adaptive Binary Arithmetic Coding

적응적 이진 산술 코딩은 심볼의 발생 확률이 독립적이지 않을 경우에 변화에 대응할 수 있도록 이전의 부/복호화 된 심볼에 따라 현재 부호화할 심볼의 발생 확률을 결정하여 부호화하

는 방법이다. 그림 2는 적응적 이진 산술 코딩 방법의 과정을 보여준다. 그림 2를 보면 심볼의 발생 확률이 심볼에 따라 변화하는 것을 알 수 있다.

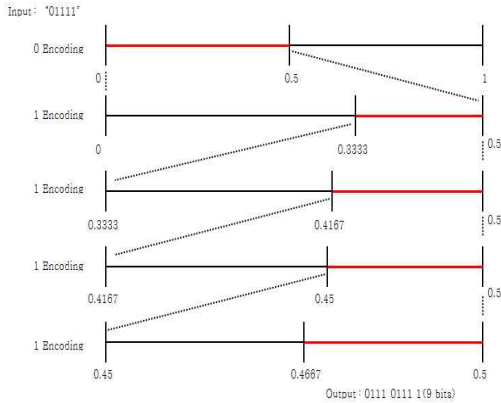


Fig. 2. Adaptive Binary Arithmetic Coding

5. Adaptive Binary Arithmetic Coding With The Changed Initial Probability

그림 3은 초기 확률을 변경한 적응적 이진 산술 부호화 과정을 보여준다. 그림 3은 그림 2와 다르게 초기 확률이 변경되어 있다. 이 초기 확률은 입력신호의 발생 확률과 근사함을 알 수 있다. 더 정확한 초기 확률을 사용한 결과, 동일한 적응적 이진 산술 코딩을 거쳤을 때 그림 2는 9비트인 반면 그림 3의 부호화 길이는 6비트임을 알 수 있다. 따라서 적응적 이진 산술 부호화에서 초기 확률 설정은 중요하다[8-10].

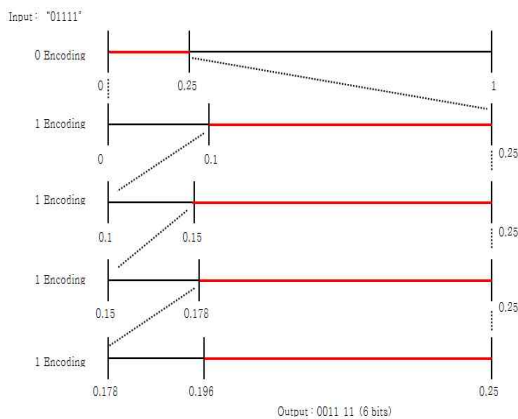


Fig. 3. Adaptive Binary Arithmetic Coding with the Changed Initial Probability

6. Context Based Adaptive Binary Arithmetic Coding

6.1 Outline Of The Context Based Adaptive Binary Arithmetic Coding

문맥 기반의 적응적 이진 산술 부호화(CABAC : Context based Adaptive Binary Arithmetic Coding)은 신택스의 문맥과 이전에 발생한 심볼을 보고 적응적으로 업데이트 된 확률

모델을 사용하는 방법이다. 신택스 요소들은 이진화 과정을 거쳐 '0' 또는 '1'으로 구성된 빈(Bin)의 배열인 빈 스트링(Bin string)으로 변환된다. 각각의 빈은 정규 또는 우회 코딩을 적용한다. CABAC은 4단계로 구성되어있다. 첫 번째는 이진화 단계이다. 이진화 단계에서는 신택스 요소들을 이진화 과정을 거쳐 '0' 또는 '1'으로 구성된 빈(Bin)의 배열인 빈 스트링(Bin string)으로 변환된다. 빈은 CABAC의 기본 단위이다. 두 번째는 문맥 모델링 단계이다. 문맥 모델링 단계는 정규 부호화 모드를 수행할 경우 수행되는 단계이다. 문맥 모델링 단계에서는 확률 모델을 결정하며 확률 모델이 결정되면 세 번째 단계인 이진 산술 부호화 단계가 수행된다. 마지막으로 네 번째 단계는 확률 업데이트 단계이다. 확률 업데이트 단계는 확률을 업데이트하여 압축 효율을 높인다. 우회 코딩을 수행할 경우 첫 번째 단계인 이진화와 세 번째 단계인 이진 산술 부호화만 수행한다.

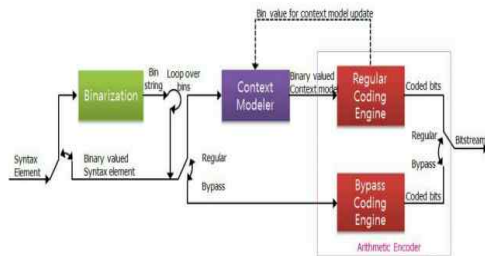


Fig. 4. Encoding Block Diagram of CABAC

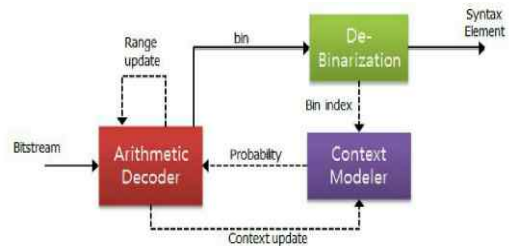


Fig. 5. Decoding Block Diagram of CABAC

6.2 Binarization

신택스는 다양한 형태가 존재하지만 CABAC의 입력은 이진 심볼만 가능하다. 따라서 다양한 형태의 신택스를 이진수 형태로 변환해야한다. 이진화 하는 방법에는 단항 이진화, 고정길이 이진화, Exp-Golomb 이진화 등 다양한 방법이 존재한다.

6.3 Context Modeling

이진화 단계의 수행 결과로 빈 스트링을 출력한다. 빈 스트링은 '0'과 '1'의 값을 갖는 빈으로 이루어졌다. 문맥은 심볼의 발생 확률에 관한 정보로서, 문맥 모델링은 이진화 결과인 빈을 입력으로 하여 이진 산술 코딩에 필요한 빈의 확률을 추정하는 과정이다.

6.4 Binary Arithmetic Coding

이진 산술 부호화는 문맥 모델링으로부터 얻은 빈의 확률 정

보를 이용하여 빈을 부/복호화한다. 이진 값의 확률에 따라 정규화 모드, 우회 모드(Bypass Mode)를 수행한다. 정규 코딩은 문맥 모델링을 사용하여 빈의 확률을 예측하는 적응적 이진 산술 코딩이며, 우회 코딩은 이진화된 빈 스트링을 그대로 비트 스트림으로 출력하는 방법으로 부/복호화기의 복잡도 감소를 위해 특정 선택스 요소에 적용되는 단순 코딩이다.

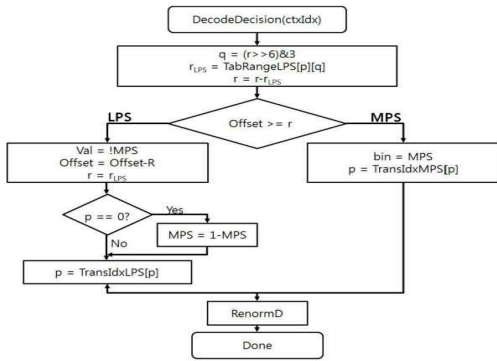


Fig. 6. Regular Mode

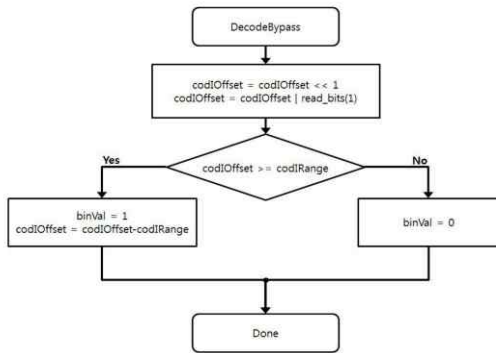


Fig. 7. Bypass Mode

6.5 Probability Update

확률 업데이트는 적응적인 확률 업데이트를 통해 빈의 확률을 더 정확히 추정한다. 현재 코딩한 값을 보고 빈의 확률을 조정하여 다음 빈의 코딩에 사용한다. 확률을 업데이트할 때 정의된 확률 이행 테이블을 이용한다. 확률 이행 테이블은 빈이 LPS (Least Probable Symbol)와 같을 경우와 MPS (Most Probable Symbol)와 같을 경우에 대해 다음 빈을 위한 상태 값을 정의하고 있다[11-14].

III. The Proposed Scheme

1. Edge Image

입력 영상인 에지 영상을 생성하기 위하여 원본 영상과 Blur된 영상의 차이를 구한다. 이 과정을 식으로 나타내면 식 (1)과 같다.

$$I_{Edge} = I_{Input} - I_{GaussianBlur} \quad (식 1)$$

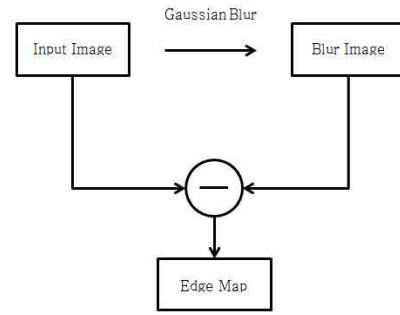


Fig. 8. Block Diagram Generating Edge Image

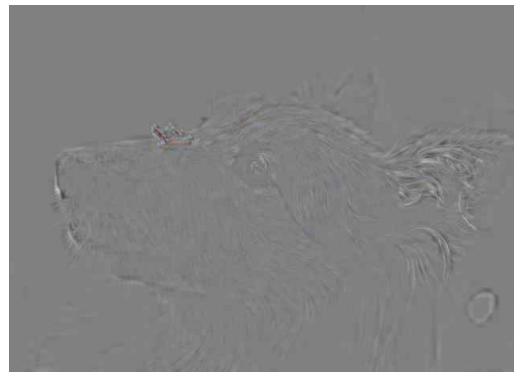


Fig. 9. Edge Image (Pixel Value + 128)

식 1의 방법으로 에지 영상을 생성하면 9 비트 영상이 되며 그림 9는 에지 영상을 표현하기 위해 각 픽셀 값에 128을 더한 영상이다.

2. Zero Block Coding

에지 영상은 그림 10과 같이 '0'을 중심으로 가우시안 분포를 가지며 픽셀 값이 '0'인 경우가 많다. 따라서 픽셀 영역에서 블록의 픽셀 값이 '0'인 경우가 다수 존재하기 때문에 블록의 픽셀 값을 체크하여 모두 '0'인 경우에는 이후의 알고리즘을 수행하지 않고 '1'이라는 flag를 전송한다. 즉, 현재 블록의 모든 픽셀 값이 '0'인지 아닌지를 Zero Block Flag로 표현한다. Zero Block Flag가 '1'이면 모든 픽셀 값이 '0'을 의미하며, 플래그가 '0'이면 '0'이 아닌 픽셀 값이 존재함을 의미한다. 따라서 수행시간이 감소하며 소요되는 bit 수 또한 줄일 수 있다. Zero Block Flag는 심볼의 발생 확률을 예측할 수 있기 때문에 정규 코딩을 수행한다.

3. Prediction Flag

Zero Block Flag가 '0'일 경우에 예측 단계를 수행한다. 이때 예측 단계에서 예측을 수행할 것인지에 대하여 판단하며 Prediction Flag는 예측 수행 여부를 나타낸다. Prediction Flag가 '1'일 경우 예측을 수행하며 '0'일 경우 예측을 수행하지 않는다. Prediction Flag는 확률을 예측할 수 있기 때문에 정규 코딩을 수행한다.

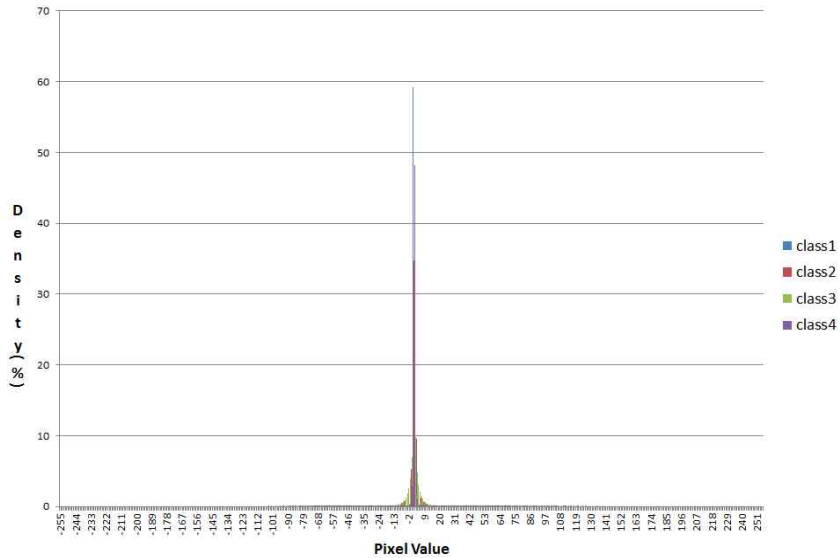


Fig. 10. Histogram of Edge Image

4. Prediction Mode

Prediction Flag가 '1'일 경우 예측을 수행한다. 예측을 수행할 경우 예측 모드(Prediction Mode)를 전송하여야 한다. 이때 사용한 방법이 MPM (Most Probable Mode)이다. MPM은 주변 블록의 예측 모드를 참조하여 현재 블록의 예측 모드를 예측하는 방법이다. MPM을 부호화하기 위하여 MPM Flag와 모드를 사용한다. MPM Flag는 후보 모드와 일치하는지를 나타내며, 모드는 후보 모드와 일치할 경우에는 후보 모드의 번호를 나타내고 후보 모드와 일치하지 않을 경우 예측 모드의 번호를 나타낸다. MPM 방식을 사용할 때 후보 모드로 3개의 모드를 사용하며, 이때 후보군에 매칭이 될 경우에는 1 bit 또는 2 bit로 표현하고 후보 모드에 매칭이 되지 않을 경우 총 5bit(32가지 모드)의 고정 비트를 할당하여 표현하게 된다.

예지 영상의 경우 현재 블록의 예측 모드가 왼쪽 블록의 예측 모드와 일치하는 경우가 가장 많이 존재한다. 따라서 첫 번째 후보 모드는 왼쪽 블록의 예측 모드로 설정한다. 두 번째로 많이 존재하는 경우가 위쪽 예측 모드와 일치하는 경우이다. 따라서 두 번째 후보 모드는 위쪽 예측 모드로 설정한다. 표 2를 보면 예지 영상에서 많이 존재하는 예측 모드는 Planar 모드와 수평 모드이다. 현재 블록의 예측 모드가 주변 블록의 예측 모드와 다르다면 Planar 모드, 혹은 수평 모드와 일치할 확률이 높다. 따라서 본 논문에서는 세 번째 후보 모드를 Planar 모드, 혹은 수평 모드로 설정한다. 표 3과 표 4는 MPM 후보 모드에 일치할 경우와 일치하지 않을 경우에 대한 이진화 테이블이다. 이진화가 끝나면 MPM Flag는 확률을 예측할 수 있기 때문에 정규 코딩을 수행하며 모드는 우회 코딩을 수행한다.

Table 2. Ratio of prediction mode in edge image

Prediction Mode	Ratio
Horizontal Mode	25 %
Planar Mode	24 %
...	...
Angular 34 Mode	0%

Table 3. Binary table of MPM mode

MPM Nominee Mode	Binary
0	0
1	00
2	01

Table 4. Rest mode except for MPM mode and binary table

Prediction Mode	Binary
0	00000
1	00001
2	00010
...	...
31	11111

5. Coefficient Coding

양자화가 끝난 후의 값을 계수(Coefficient)라고 하며 계수를 부호화하기 위하여 7단계의 과정을 수행한다. 이때 4 x 4의 서브 블록으로 분할하여 처리한다. 따라서 4 x 4 변환, 8 x 8 변환, 16 x 16 변환, 32 x 32 변환을 수행할 경우 각각 1개, 4개, 16개, 64개의 서브 블록으로 분할된다.

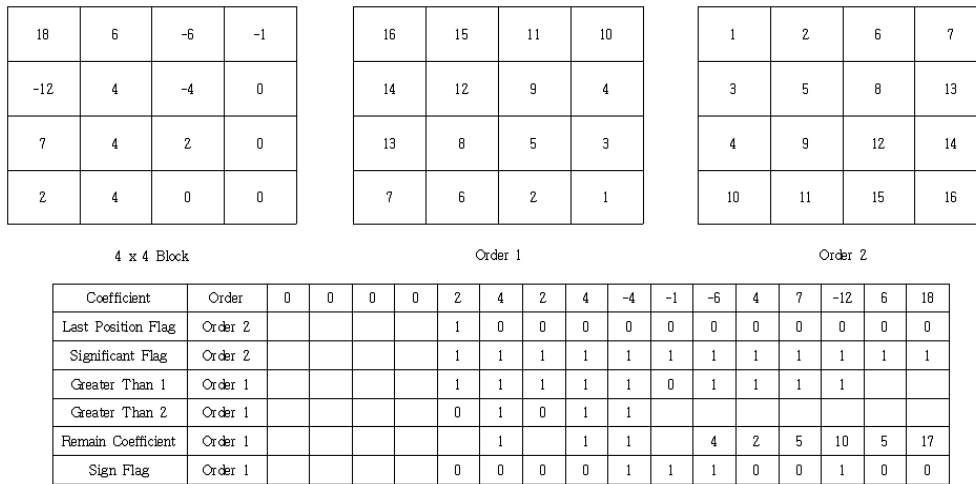


Fig. 11. Coefficient Coding

5.1 Sub Block Zero Block Flag

Sub Block Zero Block Flag는 4 x 4 블록의 모든 계수 값이 '0'인지 아닌지를 나타낸다. 이 플래그를 사용하면 계수가 모두 '0'일 경우 1개의 bit를 사용하여 계수 부호화를 표현하고 나머지 단계를 수행하지 않기 때문에 복잡도 또한 줄일 수 있다. Sub Block Zero Block Flag가 '1'일 경우 블록의 모든 계수가 '0'이며 플래그가 '0'일 경우 '0'이 아닌 계수가 존재함을 의미한다. Sub Block Zero Block Flag는 확률을 예측할 수 있기 때문에 정규 코딩을 수행한다.

5.2 Last Position Flag

Last Position Flag는 스캔을 하며 현재 계수 위치가 마지막 '0'이 아닌 계수인지에 대한 여부를 나타낸다. Last Position Flag를 사용하는 이유는 마지막 '0'이 아닌 계수의 위치를 나타내 블록의 모든 계수를 부호화하지 않아도 되므로 부호화하는 bit 수를 줄일 수 있기 때문이다. Last Position Flag가 '1'인 경우 마지막 '0'이 아닌 계수를 의미하며 '0'일 경우 반대를 의미한다. Last Position Flag는 확률을 예측할 수 있기 때문에 정규 코딩을 수행한다.

5.3 Significant Flag

Significant Flag는 계수의 값이 '0'인지 아닌지에 대한 여부를 나타낸다. 플래그가 '0'일 경우 계수는 '0'이며 플래그가 '1'일 경우 계수는 '0'이 아닌 값을 갖는다. 플래그가 '0'일 경우 더 이상 부호화를 수행하지 않기 때문에 bit 수를 줄일 수 있다. Significant Flag는 확률을 예측할 수 있기 때문에 정규 코딩을 수행한다.

5.4 Greater Than 1 Flag

Greater Than 1 Flag는 계수의 절댓값이 '1'보다 크지 또는 작거나 같은지를 나타내는 플래그이다. 계수 값이 '1'보다 작거나 같을 경우 이 플래그를 이용하여 표현하고 더 이상 부호화

를 진행하지 않기 때문에 bit 수를 줄일 수 있으며, 또한 '1'보다 클 경우 (계수-2)를 하여 부호화하기 때문에 더 적은 bit 수로 계수를 부호화할 수 있다. Greater Than 1 Flag는 예지 영상에서 압축률을 높이기 위해 1개의 Sub Block에 대하여 10개의 flag를 사용하며 Greater Than 1 Flag는 확률을 예측할 수 있기 때문에 정규 코딩을 수행한다.

5.5 Greater Than 2 Flag

Greater Than 2 Flag는 계수의 절댓값이 '2'보다 크지 또는 작거나 같은지를 나타내는 플래그이다. 계수 값이 '2'보다 작거나 같을 경우 이 플래그를 이용하여 표현하고 더 이상 부호화를 진행하지 않기 때문에 bit 수를 줄일 수 있으며, 또한 '2'보다 클 경우 (계수-3)을 하여 부호화하기 때문에 더 적은 bit 수로 계수를 부호화할 수 있다. Greater Than 2 Flag는 Edge Map에서 압축률을 높이기 위해 1개의 Sub Block에 대하여 5개의 flag를 사용하며 Greater Than 2 Flag는 확률을 예측할 수 있기 때문에 정규 코딩을 수행한다.

5.6 Remain Coefficient Coding

위의 모든 과정을 수행한 후, 잔여 계수를 부호화하기 위해 Exp-Golomb 이진화를 사용하며 우회 코딩을 수행한다. Exp-Golomb 이진화는 점두사와 점미사로 이루어져 있다. 표 5는 정수 값에 대한 Exp-Golomb 이진화 테이블이다. 식 (2)를 보면 n은 양의 정수를, ID는 n이 속한 Group의 ID를 의미한다. 식 (3)은 점두사를 구하는 식이다. UnaryCode는 입력 받은 ID를 Unary 이진화 방식으로 변환하는 함수이다. 식 (4)는 점두사의 Index를 구하는 식이다. 식 (4)의 결과로 Index를 구하면 식 (5)를 이용해 최종적으로 점미사를 표현한다. WriteBits는 ID 크기만큼의 비트 수를 할당하고 Index를 이진수로 변환하는 함수이다[15].

Table 5. Exp - Golomb binarization

Num	Prefix	Suffix	Code (Prefix) + (Suffix)
0	0		0
1	10	0	100
2	10	1	101
3	110	00	11000
4	110	01	11001
5	110	10	11010
6	110	11	11011
7	1110	000	1110000
8	1110	001	1110001
9	1110	010	1110010
10	1110	011	1110011
11	1110	100	1110100

$$ID = \lfloor \log_2(n+1) \rfloor \quad (\text{식 } 2)$$

$$prefix = UnaryCode(ID) \quad (\text{식 } 3)$$

$$Index = n - ((1 \ll ID) - 1) \quad (\text{식 } 4)$$

$$suffix = WriteBits(Index, ID) \quad (\text{식 } 5)$$

5.7 Sign Flag

Sign Flag는 계수 값의 부호를 나타내는 플래그이다. 플래그의 값이 '0'일 경우 계수의 부호는 양수를 의미하며, 플래그의 값이 '1'일 경우 계수의 부호는 음수를 의미한다. Sign Flag는 확률을 예측하기 어렵다. 따라서 우회 코딩을 수행한다.

6. Context Design Utilizing The Characteristics Of The Edge Image

특성에 맞는 문맥 모델을 설계하면 알고리즘의 수행 시간은 줄고 영상의 압축률은 높아진다. 따라서 에지 영상의 특성에 맞게 문맥을 설계하는 것은 중요하다.

문맥의 개수는 알고리즘의 초기 설정 시간에 영향을 미친다. 문맥의 개수가 많으면 각 문맥을 초기화하는데 드는 시간이 증가하여 최종적으로 알고리즘 수행 시간이 증가한다. 따라서 문맥의 개수를 최대한 작게 해야 한다. 본 논문에서 문맥의 개수를 줄이기 위하여 다양한 에지 영상을 바탕으로 각 심볼의 발생 확률을 수집하며 확률이 비슷한 심볼들을 묶어 동일한 문맥 모델을 사용한다.

문맥 모델의 초기 확률 설정은 2.5절에서 서술했듯이 압축률에 영향을 미친다. 에지 영상은 자연영상과 달리 에지 영상의 특성이 존재하여 심볼의 발생 확률이 자연영상과 다르다. 따라서 자연영상에서 사용하던 초기 확률을 사용하게 되면 심볼의 발생 확률에 차이가 존재하여 높은 압축률을 기대하기 어렵다. 본 논문에서 압축률을 높이기 위하여 다양한 에지 영상을 바탕으로 각 심볼의 발생 확률을 수집하며 확률의 평균값을 계산한다. 계산한 평균값은 심볼의 발생 확률과 유사하다. 따라서 평균값을 초기 값으로 설정하여 압축률을 높인다.

IV. Results

본 논문에서는 에지 영상의 특성을 분석하여 에지 영상에 특화된 Entropy Coding 알고리즘을 제안하였다. 알고리즘의 성능 평가를 위해서 640x480, 1280x720, 1920x1080 크기의 영상을 사용하였다. 기존의 압축 방법인 JPEG과 제안한 방법의 결과를 비교하여 제안한 알고리즘의 우수성을 입증하였다. 실험에 사용된 컴퓨터의 사양은 아래와 같다.

CPU : Intel Core i7 4.00GHz

RAM : 16.0GB

O S : Windows 7 Ultimate K

표 6은 동일 PSNR일 때 JPEG과 제안 알고리즘의 압축 용량을 비교한 표이다. 총 8개의 이미지에 대하여 실험을 수행하였다. 제안 방법은 JPEG보다 이미지 A에서는 약 2.8배, 이미지 B에서는 약 11배 정도의 좋은 압축 성능을 보인다. 모든 경우에서 JPEG보다 더 좋은 압축률을 보이며 적게는 약 2.8배, 많게는 약 11배의 압축 성능을 보인다. 표를 통해 제안 방법이 JPEG 보다 에지 영상에 대한 압축 성능이 우수함을 알 수 있다.

Table 6. Comparison of the data capacity compared to the same PSNR

Image	Capacity of JPEG (KB)	Capacity of Proposed Algorithm (KB)
Image A	110	38.56
Image B	100	8.99
Image C	105	25.97
Image D	101	13.16
Image E	70	17.23
Image F	101	13.87
Image G	100	14.33
Image H	102	31.41

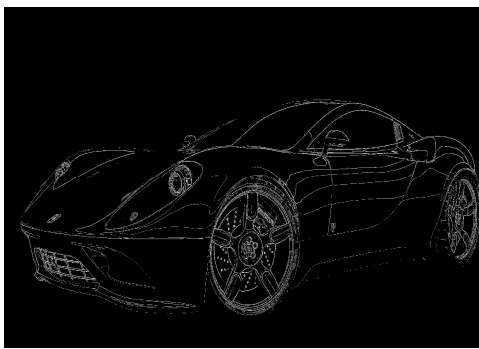
Table 7. Comparison of PSNR compared to the same data capacity

Image	Average PSNR of JPEG (dB)	Average PSNR of Proposed Algorithm (dB)
Image A	42.21	54.28
Image B	39.35	54.34
Image C	42.58	53.82
Image D	42.79	49.56
Image E	43.94	62.21

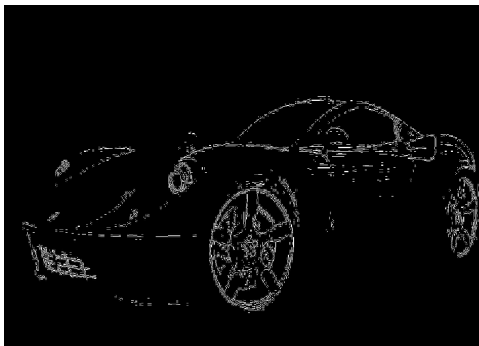
표 7은 동일 용량일 때 JPEG과 제안 알고리즘의 PSNR을 비교한 표이다. 총 5개의 이미지에 대하여 실험을 수행하였다. 제안 방법은 JPEG보다 이미지 D에서는 약 6.7 dB, 이미지 E에서는 약 18.2 dB 정도의 좋은 화질을 보인다. 모든 경우에서

JPEG보다 더 좋은 화질을 보이며 적게는 약 6.7 dB, 많게는 약 18.2 dB의 우수성을 보인다. 표를 통해 제안 방법이 JPEG보다 에지 영상에 대하여 화질이 우수함을 알 수 있다.

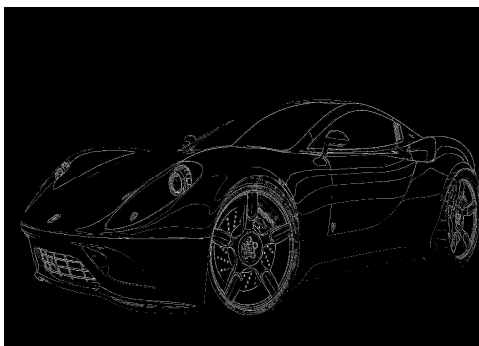
그림 12와 13은 각 압축 방법들의 결과를 나타낸 것이다. 그림 12와 13에서 JPEG의 결과는 원본영상이 가진 에지 성분들의 손실이 발생하는 것을 보여준다. 반면 제안 방법은 이러한 에지 성분의 손실이 JPEG보다 많이 개선되었으며 원본의 에지 영상과 차이가 적음을 육안으로 확인할 수 있다. 따라서 제안 방법의 에지 영상은 화질이 우수하여 활용 및 다양한 분야에 응용이 가능하다.



(a)

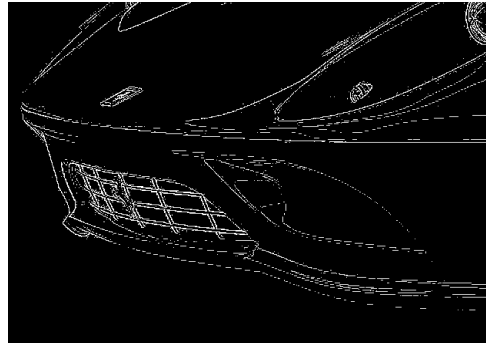


(b)

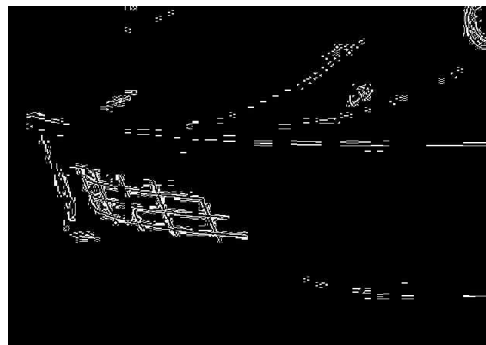


(c)

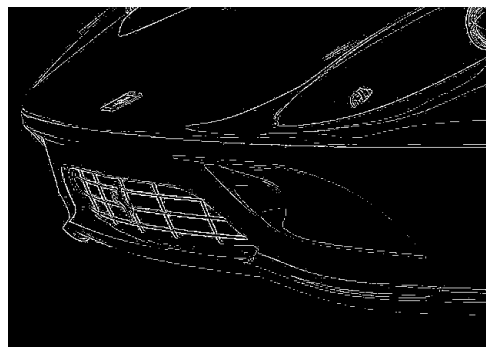
Fig. 12. (a) Original Edge Image (b) JPEG (c) Proposed Algorithm



(a)



(b)



(c)

Fig. 13. (a) Zoomed Image of the Original Edge Image (b) Zoomed Image of JPEG (c) Zoomed Image of the Proposed Algorithm

V. Conclusions

본 논문에서는 에지 영상의 특성을 분석하여 에지 영상에 특화된 Entropy Coding 알고리즘을 제안하였다. 지금까지의 압축 알고리즘은 자연영상을 기반으로 개발되었기 때문에 에지 영상을 압축할 경우 압축률이 낮고 알고리즘이 복잡하여 수행

시간이 길다. 따라서 에지 영상에 적합한 알고리즘이 필요하다. 에지 영상은 pixel 값이 '0'을 기준으로 가우시안 분포를 이루며 대부분의 값이 '0'을 갖는다. 따라서 zero block 기법을 사용한다. 또한, 에지 영상의 화면 내 예측 모드는 주변 블록과 유사하거나 Planar 모드, 혹은 수평 모드가 나올 가능성이 크다. 본 논문에서는 가능성이 높은 모드를 이용하여 예측모드를 만드는 MPM 기법을 사용하고, 이를 활용하여 에지 영상의 특징에 적합한 Entropy Coding을 설계하였다. 제안 방법은 동일 PSNR일 경우에는 JPEG보다 평균 5.9배의 높은 압축효율을 보이며, 동일 용량일 경우에는 JPEG보다 평균 12.6 dB의 우수한 화질을 보인다. 본 연구를 통해 개발한 에지 영상 압축 성능이 우수함을 확인할 수 있다.

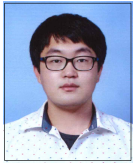
REFERENCES

- [1] P. B. Khobragade, S. S. Thakare, "Image Compression Techniques- A Review," *International Journal of Computer Science and Information Technologies*, Vol. 5(1), 2014.
- [2] A. Lata, P. Singh, "Review of Image Compression Techniques" *International Journal of Emerging Technology and Advanced Engineering*, Vol. 3, Issue 7, Jul. 2013.
- [3] M. Sathya Deepa and Dr. N. Sujatha, "Image Compression Using MH Encoding," *International Journal of Computer Trends and Technology (IJCTT)*, Vol. 13, NO. 2, Jul. 2014.
- [4] M. AL-Ani and F. Awad, "The JPEG Image Compression Algorithm," *International Journal of Advances in Engineering & Technology (IJAET)*, Vol. 6, Issue 3, pp. 1055-1062, Jul. 2013.
- [5] K. Wallace, "The JPEG Still Picture Compression Standard," *IEEE Transactions on Consumer Electronics*, Vol. 38, NO. 1, Feb. 1992.
- [6] T. Rabie, "Lossless Quality Steganographic Color Image Compression," *International Journal of Advanced Computer Science and Applications*, Vol. 6, NO. 4, 2015.
- [7] D. Marpe, H. Schwarz, and T. Wiegand, "Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard," *IEEE Transactions Circuits and Systems for Video Technology*, Vol. 13, NO. 7, pp. 620-636, Jul. 2003.
- [8] G. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 22, NO. 12, Dec. 2012.
- [9] V. Sze, and M. Budagavi, "High Throughput CABAC Entropy Coding in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 22, NO. 12, Dec. 2012.
- [10] D. Marpe, G. Blättermann, T. Wiegand, R. Kurceren, M. Karczewicz, and J. Lainema, "New Results on Improved CABAC," in *Joint Video Team of ISO/IEC JTC1/SC29/WG11 & ITU-T SG16/Q.6 Doc. JVTB101*, Geneva, Switzerland, Feb. 2002.
- [11] D. Marpe, H. Schwarz, G. Blättermann, and T. Wiegand, "Final CABAC Cleanup," in *Joint Video Team of ISO/IEC JTC1/SC29/WG11 & ITU-T SG16/Q.6 Doc. JVT-F039*, Awaji, Japan, Dec. 2002.
- [12] T. Wiegand, G. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, NO. 7, Jul. 2003.
- [13] H. Kim, "A Study on the Hardware Design of CABAC for Performance Improvement of HEVC Decoder," *Master's Thesis of Hanbat University of Technology*, 2013.
- [14] Y. Jo, "A Design of Rescheduling and Parallel Architecture for CABAC Decoder in HEVC," *Master's Thesis of Kwangwoon University of Technology*, 2011.
- [15] S. Xue and B. Oelmann "Hybrid Golomb Codes for a Group of Quantised GG Sources," *IEE Proceedings - Vision, Image and Signal Processing*, Vol. 150, pp. 256-260, Aug. 2003.

Authors



Jong Woo Han is now a B.S. and M.S. candidate in Department of Computer and Communications Engineering at Kangwon National University. His research interests are in the areas of computer vision, parallel processing on GPU, and image compression.



Do Hyun Kim is now a B.S. and M.S. candidate in Department of Computer and Communications Engineering at Kangwon National University. His research interests are in the areas of image compression, machine learning, and computer vision.



Yoon Kim received a B.S. degree in 1993, an M.S. degree in 1995, and a Ph.D. degree in 2003, in electronic engineering with the Department of Electronic Engineering from Korea University.

In 2004, he joined the Department of Computer and Communications Engineering, Kangwon National University, where he is currently an professor. From 1995 to 1999, he was with the LG-Philips LCD Co., where he was involved in research and development on digital image equipment. His research interests are in the areas of video signal processing, multimedia communications, and wireless sensor networks.