

논문 2012-50-6-13

Multi-Level 양자화 기법을 사용한 BTC 영상 압축 알고리즘

(BTC Algorithm Utilizing Multi-Level Quantization Method for Image Compression)

조 문 기*, 윤 영 섭*

(Moonki Cho and Yungsup Yoon[©])

요 약

BTC 영상 압축은 간단하고, 하드웨어 구현이 용이하여, LCD의 오버드라이브에 필요한 영상 압축 기법으로 널리 사용되어 지고 있다. 본 논문에서는 압축 손실을 줄이기 위한 방법으로, MLQ-BTC (Multi-Level Quantization BTC) 알고리즘을 제안 한다. MLQ-BTC 알고리즘은 입력 영상을 Quasi 8-level BTC 방법과 Advanced 2-level BTC 방법으로 압축 및 복원하여, 압축 손실이 적은 알고리즘을 선택하는 과정이다. 시뮬레이션으로 기존의 BTC 알고리즘과 PSNR 및 압축비율의 비교를 통해서, 제안한 알고리즘의 효율성을 확인하였다.

Abstract

BTC image compression is a simple and easy hardware implementation, is widely used in a video compression techniques required for LCD overdrive. In this paper, methods for reducing compression loss, a multi-level quantization BTC (MLQ-BTC) algorithm is proposed. The process of the MLQ-BTC algorithm is, a input image is compressed and decompressed by Quasi 8-level method and Advanced 2-level BTC method, and select the algorithm with the smallest compression loss. Simulation results show that the proposed algorithm is efficient as compared with PSNR and compression ratio of the existing BTC methods.

Keywords : overdrive technique, image compression, block truncation coding, multi-level Quantization

I. 서 론

Liquid Crystal Display (LCD)는 hold-type 디스플레이 방식으로, 액정의 물리적인 상태 변경으로 색을 표현하는 원리를 사용한다. 하지만 액정의 늦은 반응 속도로 인해서, Cathode Ray Tube (CRT)와 같은 임펄스 구동 방식보다 응답속도가 현저히 늦은 단점이 있다. 이와 같이 액정의 늦은 반응속도로 인한, motion-blur 현상을 감소시키는 기술로 액정에 인가되는 데이터의 크기를 변조하는 오버드라이브 방법을 널리

사용되고 있다. 오버드라이브를 적용하기 위해서는 이전 프레임과 현재 프레임 영상을 비교해야 되기 때문에 이전 프레임 영상이 모두 필요하게 되어, 외부 메모리를 사용하여 이전 프레임 데이터를 저장하는 방법이 사용된다. 최근에는 해상도 증가와 함께 원가 절감 및 메모리 대역폭 증가로 인한 클럭 주파수 감소를 위해서, 메모리 용량을 감소시키는 방법으로 오버드라이브를 적용하는 Timing Controller (T-Con)에, 압축 알고리즘을 반영하여, 외부 메모리 크기를 줄이는 연구가 진행되고 있다.

오버드라이브 적용에서는 프레임마다 데이터를 쓰고 읽는 과정이 필요하므로, 실시간으로 데이터 처리가 가능한 저 복잡도 압축 기술이 필요하게 된다. 그리고 오버드라이브 적용에 필요한 압축 알고리즘의 압축비는

* 정회원, 인하대학교 전자공학과
(Div. of Electronic Engineering, Inha University)
※ 이 논문은 인하대학교의 지원에 의하여 발간되었음.
© Corresponding Author(E-mail: ysyoon@inha.ac.kr)
접수일자: 2013년1월26일 수정완료일 2013년5월22일.

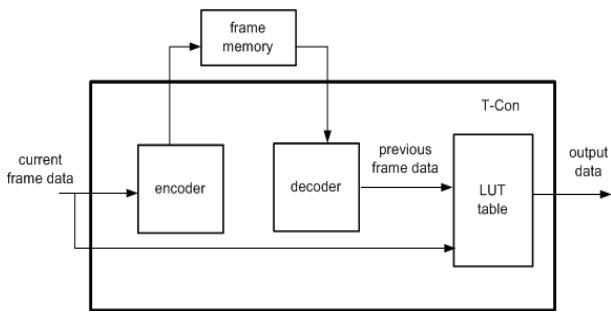


그림 1. 오버드라이브 도식도
Fig. 1. Block diagram of overdrive.

입력 영상의 데이터 분포에 따라서 달라지는 방법보다 일정한 압축비로 처리하는 방법이 메모리 컨트롤러 설계에도 유리하게 되어, 구현이 간단하고 압축비가 일정한 Block Truncation Coding (BTC)이 널리 사용되고 있다.^[1~2] BTC 압축방법은 1979년 Delp, Mitchell 그리고 Calton 등이 발표한 공간 코딩 방법으로 비교적 부호화를 위한 연산이 적어 하드웨어로 구현이 간단한 압축 알고리즘으로 알려져 있다.^[3] 표준 BTC 알고리즘은 입력 영상을 서브 블록으로 나누고, 평균, 표준편차 그리고 비트맵 데이터를 전송하는 방법으로, 1차 및 2차 모멘트 보존 원리를 근간하여 영상을 재구성하는 원리이다. 하지만, 표준 BTC는 제공된 연산이 필요함으로서, 최근에는 제공된 연산이 없어 하드웨어 구현이 간단하고 화질 측면에서도 우수한 AMBTC (Absolute Moment Block Truncation Coding)가 널리 사용되어 지고 있다.^[4] AMBTC는 1984년 Lema 와 Mitchell에 의해서 발표된 방법으로 1차 및 2차 모멘트 보존 원리 대신 1차 절대 모멘트를 보존하는 방법이다. 1차 절대 모멘트는 평균과 각 화소 값과의 차이 정도를 나타내는 정보를 가지고 있다. 일반적으로 BTC 압축에서 압축손실을 줄이기 위해서는, 적용하는 블록크기를 줄이거나, 높은 양자화 레벨을 사용하여야 한다. 하지만, 위 2 가지 요소를 적용할 경우 적은 압축손실로 화질은 개선되나, 압축비가 현저히 저하된다. 본 논문에서는 적용하는 압축 알고리즘에 따라서, 블록크기를 달리하고 양자화 레벨에 따라 비트맵 및 양자화 데이터를 압축하는 BTC 압축 알고리즘을 적용하여 전체적으로 6:1의 일정한 압축률을 가지도록 하였다. 다양한 영상으로 시뮬레이션으로 PSNR 결과를 확인한 결과, 4:1의 압축률을 갖는 4x4 블록의 2-level AMBTC 결과보다 평균 2.12 dB 높은 PSNR 결과를 확인되어, 본 논문에서 제안하는 알고리즘의 효율성을 확인할 수 있었다.

II. 이론적 배경

2.1. 2-level AMBTC

AMBTC의 비트맵 양자화 레벨은 2-level 또는 4-level 이상으로 구분이 가능하다. 2-level 양자화 방법을 설명하면, 그림 2와 같이, 전체 영상을 일정한 크기로 분할하여, 분할 된 블록 마다, 평균값을 계산하여 각 화소 값과 비교하여 평균 이상이면 1, 미만이면 0을 할당한다. 2-level AMBTC는 입력 영상을 단순히 평균을 기준으로 높은 값과 낮은 값으로만 복원 하게 되며, 4x4의 블록 크기 기준으로 4:1의 압축률을 가지게 된다.

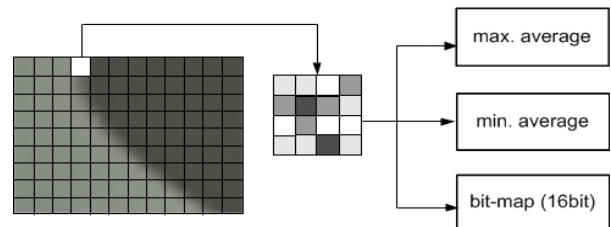


그림 2. 4x4 블록의 2-level AMBTC 알고리즘
Fig. 2. 2-level AMBTC algorithm of 4x4 block.

2.2. 8-level AMBTC

일반적인 8-level AMBTC 압축은, 7개의 임계값을 기준으로 8개의 영역으로 분리한 다음, 블록내의 데이터를 111, 110, 101, 100, 011, 010, 001, 000(binary)의 비트맵으로 표현하여, 수신측에 전달하고, 비트맵 정보로 각각의 레벨에 맞는 데이터 값으로 복원 할 수 있도록 하는 방법이다. 본 논문에서는 7개의 임계값을 적용하는 방법은 압축비의 현저한 저하를 가져오게 된다. 그래서 본 논문에서는 압축률 향상을 위해서, 1개의 절대 중심 모멘트만을 활용한 방법으로 진행하였다.^[5] 본 논문에서 사용한 8-level BTC 압축 방법은 다음과 같다. 우선, 동작범위의 임계값을 구하기 위해서, 블록내의 데이터들의 상위평균을 구하고 구해진 상위 평균에 속하는 데이터들의 상위평균(T_H)을 구한다. 동일한 방법으로, 하위평균(T_L)을 구한다. 구해진 T_H 와 T_L 값으로, 블록내의 평균 (\bar{B})과 동작범위 (dynamic range, \bar{D})를 아래 식에 의해서 구할 수 있다.

$$\bar{B} = \left[\frac{1}{e} \sum_{B(i) \geq T_H} B(i) + \frac{1}{f} \sum_{B(i) < T_L} B(i) \right] / 2 \quad (1)$$

$$\bar{D} = \left[\frac{1}{e} \sum_{B(i) \geq T_H} B(i) - \frac{1}{f} \sum_{B(i) < T_L} B(i) \right] \quad (2)$$

식(1), 식(2)에서, B(i)는 블록내의 영상 데이터이다. 그리고 e는 블록 내에서 T_H 보다 크거나 같은 데이터의 수이고, f는 블록 내에서 T_L 보다 작은 데이터의 수이다. 식(1)와 식(2)에 의해서 구해진 평균과 동작범위를 활용하여, 각각의 블록 7개의 임계값을 아래와 같이 구할 수 있다.

$$TH_N = \bar{B} + \frac{N-4}{6} \bar{D} \quad (N=1,2,3,4,5,6,7) \quad (3)$$

구해진 TH_N 은 기준 임계값이 7개이므로, 8개의 영역으로 표현이 가능하다. 수신측에는 평균과 동작범위 그리고, 8-level 비트맵으로 복원 시, 아래 수식에 따라서 복원하게 된다.

$$Brec(i) = \bar{B} + \frac{2^* Q(i) - 7}{14} \bar{D} \quad (4)$$

식(4)에서, Q(i)는 8-level 비트맵을 나타낸다. Brec(i)는 복원 데이터의 표현으로서, 블록내의 평균 (\bar{B})과 동작범위 (dynamic range, \bar{D})와 Q(i) 값으로 복호가 가능하다.

2.3. adaptive quantization coding (AQC)

AQC는 Wang 논문^[6]에서 소개되었던 압축 기법으로 BTC와 유사한 압축방법이다. 복호기에 복호에 필요한 신호는 $m \times n$ 의 압축 단위 블록에서 구해진 min. 화소값(min.v)과 비트맵(Bmp) 그리고, 양자화 구간(Q_{step})이다. 양자화 구간(Q_{step})은 max. 화소값(max.v)에서 min. 화소값(min.v) 차이를 원하는 AQC의 양자화 레벨(Q_{level})에서 1을 뺀 값으로 나눈 값이고, 비트맵은 식(6)에서 알 수 있듯이, $m \times n$ 의 입력 데이터(I_{data})에서 min.v의 차이에 Q_{step} 으로 나눈 값으로 비트맵을 구할 수 있다.

$$Q_{step} = (\max.v - \min.v) / (Q_{level} - 1) \quad (5)$$

$$Bmp = (I_{data} - \min.v) / Q_{step} \quad (6)$$

III. 관련 기존 연구

3.1. CE-BTC algorithm

서론에서 언급한 바와 같이, BTC에서 적은 압축 손실을 얻기 위해서는 크기가 작은 블록에서 BTC를 적용하거나, 양자화 레벨을 높이는 방법이 필요하다. CE-BTC^[8] 압축 알고리즘은 압축 손실을 최소화하기 위해서 적용하는 블록의 크기를 줄이는 방법을 사용하였다. 그런데 적용하는 블록의 크기를 감소시키면 압축비가 낮아지게 때문에 CE-BTC 압축 알고리즘에서는 3x3 블록 4개에서 출력되는 상위평균과 하위평균들을 압축하여 원하는 압축비를 얻는 방법을 사용하였다. CE-BTC 압축 방법을 자세히 설명하면, 입력 데이터를 12x3 블록으로 구분한 후에, 12x3 블록을 3x3 서브블록으로 세분화한다. 그리고 3x3 서브블록에서의 서브그레이 데이터들을 휘도 데이터(Y)로 변환한 후, Y에서 2-level BTC를 수행하여 2-level 비트맵을 얻는다. 구해진 Y의 비트맵을 서브 그레이들의 비트맵으로 치환하여 서브블록들에서의 서브 그레이의 상위평균과 하위평균들을 구한다. 그리고 6:1의 압축비를 얻기 위해서 서브 그레이별로 출력된 상위평균과 하위평균들을 그대로 복호기로 전송하지 않고 서브 그레이별로 AQC를 활용하여 상위평균과 하위평균들을 압축하는 방법을 사용하였다.

3.2. FBTC algorithm

FBTC^[5] 압축 알고리즘은 입력되는 서브 그레이 데이터들을 YCbCr으로 변환하여 압축을 진행한다. 일반적으로 YCbCr 변환 방법은 색차 데이터(CbCr)를 휘도 데이터(Y)에 비해서 적은 양의 데이터로 변환하는 방법을 적용하여, YCbCr의 변환 방법만으로 일정한 압축비를 얻을 수 있다. FBTC 압축에서도 입력되는 서브 그레이 데이터들을 YCbCr로 변환하는 방법을 사용하였다. Y와 CbCr의 비트 할당을 달리하는 2개의 압축모드를 설정하고 복호 과정을 통해서 원본 데이터와 오차가 적은 모드를 선택하는 방법을 사용하였다. 2개의 압축 모드 중에, 모드-1은 Y를 4x4 블록에서 4-level BTC 압축 방법을 적용하여 블록마다 48 비트를 할당하여 압축을 진행하였고, CbCr은 4x4 블록에서 각 8 비트를 할당하여 압축비가 6:1이 되도록 구성하였다. 모드-2에서는 Y는 4x4 블록에서 2-level BTC를 사용하여 블록마다 32 비트를 할당하고, CbCr은 8x4 블록에서 각 32 비

트를 사용하여 모드-1과 동일한 6:1의 압축비를 갖도록 하였다.

IV. MLQ-BTC algorithm

4.1. Quasi 8-level BTC algorithm

본 논문에서 제안하는 MLQ-BTC의 비트맵은 압축비 향상을 위해서, 기존 논문에서 소개된 방법과 같이, 휘도 데이터(Y)의 비트맵을 서브 그레이의 비트맵으로 적용하는 방법을 사용하였다.^[7] Y의 비트맵을 서브 그레이의 비트맵에 적용하는 방법은 연산이 적어 하드웨어 구성이 용이하고, 기존 논문에서 발표된 내용처럼, 압축손실이 낮고, 높은 화질 특성을 얻을 수 있다. Y의 비트맵을 서브 그레이 비트맵(이하 Y 비트맵 방법)으로 활용할 경우, 1/3 로 비트맵 데이터가 축소되는 결과를 얻을 수 있다.

$$Y = 0.299R + 0.587G + 0.114B \quad (7)$$

하지만, 8-level BTC의 경우에는 화소마다 3 비트의 비트맵이 필요하기 때문에 Y 비트맵 방법만으로는 본 논문에서 목표치인 6:1의 압축비를 얻을 수 없다. 그래서 본 논문에서 제안하는 Quasi 8-level BTC 압축에서는 비트맵을 압축하기 위해서, 그림 3과 같이, 블록내의 데이터 중 절반에 해당되는 12개 데이터들은 8-level BTC 압축 방법 대신, 인접 화소 값으로, 치환하는 방법을 사용하였다. 일반적으로 특정 위치의 화소 값은 특정 위치의 인접된 화소 값과 유사한 값을 가지게 됨으로써, Quasi 8-level BTC 압축에서는 비트맵 압축 방법으로 인접 화소 값을 치환하는 방법을 사용하였다. 치

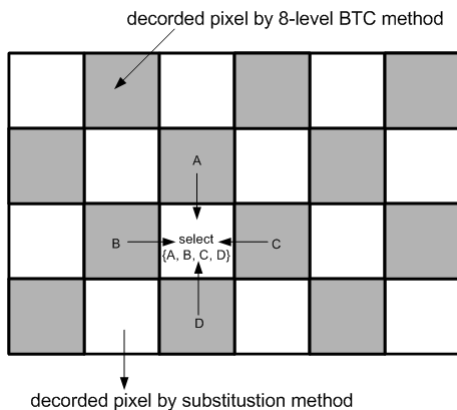


그림 3. Quasi 8-level 개념도
Fig. 3. Quasi 8-level Diagram.

환하는 방법은 복원된 상하좌우 4개의 화소 값과 비교하여 가장 오차가 적은 화소 값을 선택하는 방법을 사용하였다. 8-level BTC 으로 복원하기 위해서는 각 화소마다 3 비트가 비트맵이 필요하나, 인접 화소로 치환하는 방법은 4개의 화소 값으로 대체할 수 있으므로 2 비트의 선택 신호만으로 복호가 가능하다. 그림 3과 같이, 인접 화소로 대체되는 화소수가 전체 화소수의 절반이기 때문에 Quasi 8-level BTC의 비트맵은 화소마다 2.5 비트가 필요하게 된다.

Quasi 8-level BTC 방법을 순차적으로 설명하면, 입력 데이터를 Y로 변환하고 변환된 Y에서 8-level BTC를 수행하여, 화소마다 3 비트의 비트맵 및 Y의 평균과 동작범위를 구한다. 그리고 구해진 8-level Y 비트맵과 평균, 동작 범위로 그림 3에서 표현한 바와 같이 8-level BTC 압축 방법이 적용되는 화소부분을 복호시킨다. 그리고 치환방법이 적용되는 픽셀은 4개의 복호된 인접 데이터와 비교하여 가장 오차가 적은 데이터를 구하고 해당되는 데이터를 2 비트의 선택신호로 표현한다. 서브 그레이의 양자화 데이터는 8-level BTC 방법으로 양자화 데이터를 구한다. 그래서 복호기에서는 8-level 비트맵과 인접 화소 선택 신호 및 서브 그레이의 양자화 데이터를 사용하여 최종 출력 데이터가 복호된다. 즉, Quasi 8-level BTC는 비트맵을 Y 비트맵 방법과 인접 화소 데이터를 치환하는 방법을 혼용하여,

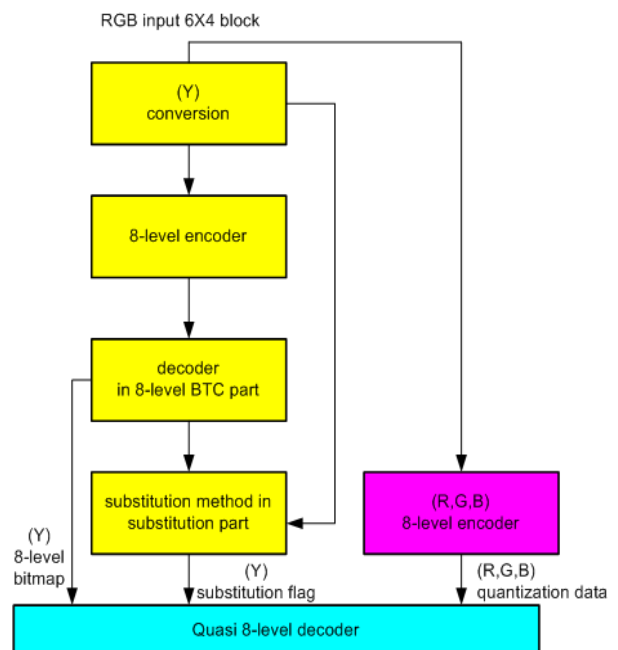


그림 4. Quasi 8-level 도식도
Fig. 4. Architecture Quasi 8-level.

화소마다 사용되는 비트맵을 2.5개로 줄이고 양자화 데이터에 8-level BTC 방법을 적용하여 6:1의 압축비를 얻을 수 있었다.

4.2. Advanced 2-level BTC algorithm

MLQ-BTC 알고리즘 중 Advanced 2-level BTC 방법은 압축 효율을 높이기 위해서 원본 6x4 블록을 2개의 3x4 블록으로 나누어 압축을 적용하였다. Y 비트맵 방법을 적용하여, 비트맵을 1/3 으로 압축하고, 2개의 서브블록에서 서브 그레이별로, 상위 평균과 하위 평균을 구하였다. 하지만 3x4 블록크기에서 Y 비트맵 압축 방법만으로는 6:1 압축비를 얻을 수 없기 때문에 추가적인 압축기법이 더 요구된다. 그래서 본 논문에서 제안하는 Advanced 2-level BTC 방법에서는 압축비를 높이기 위해서 복호에 필요한 상위 평균과 하위 평균들 (이하 양자화 데이터)을 압축하는 방법을 사용하였다. 그림 5와 같이 양자화 데이터 압축 방법은 2-level BTC 과정에서 서브 그레이별로 출력되는 4개의 양자화 데이터에 8-level AQC를 사용하여 압축하였다.

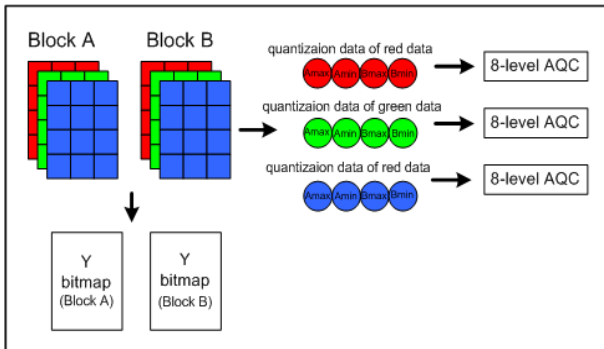


그림 5. Advanced 2-level 도식도
Fig. 5. Advanced 2-level Diagram.

4.3. Multi-Level Quantization BTC (MLQ-BTC) algorithm

본 논문에서 제안하는 MLQ-BTC는 앞 절에서 소개한 Quasi 8-level BTC 와 Advanced 2-level BTC를 각각 압축하고 복원하여 원본 데이터와 비교하여 가장 손실이 적은 방법을 선택하는 기법을 사용하였다. 두 가지 방법이 사용한 블록 크기가 같으므로 블록마다 1비트 선택신호를 복호기로 전송하면 해당 복원 방법으로 복호가 가능하다. MLQ-BTC의 할당된 비트를 살펴보면, 6x4 블록에서 6:1 압축의 경우, 그림 6에서 표현

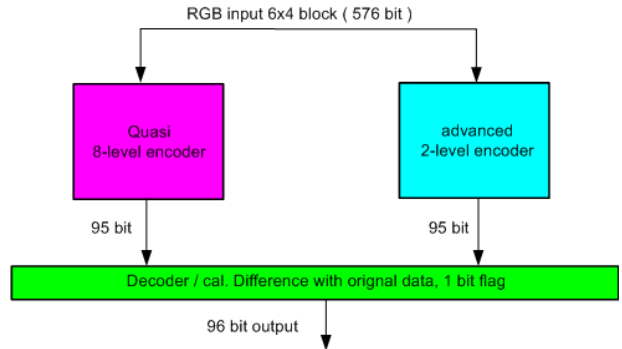


그림 6. MLQ-BTC 도식도
Fig. 6. MLQ-BTC Diagram.

한 방식과 같이 필요한 비트수는 총 96 비트이다. Quasi 8-level BTC의 비트맵과 선택신호는 화소마다 2.5 비트가 구성됨으로 총 60 비트가 할당된다. 동작범위와 평균은 모두 35 비트로 구성하여, 총 95 비트로 압축하였다. Advanced 2-level BTC의 경우, 비트맵은 총 24 비트로 표현하였고, 양자화 데이터의 압축에 필요한 min. 값들과 양자화 구간 값들은 71 비트를 적용하여, 총 95 비트를 사용하여, 전체적으로 압축비가 6:1이 되도록 구성하였다.

V. Simulation 결과

표 1은 528x528 크기의 다양한 정지 영상에 대해서 본 논문에서 제안하는 MLQ-BTC의 PSNR 결과이다. MLQ-BTC의 압축 효율을 비교 분석하기 위해서 AMBTC(4x4)와 AQC(6x4), CE-BTC^[8](12x3) 그리고 FBTC^[5](8x4)의 결과와 비교하였다. 압축비는 AMBTC

표 1. PSNR 결과
Table 1. Result of PSNR.

	PSNR (dB)				
	AMBTC (4x4)	AQC (6x4)	CE-BTC (12x3)	FBTC (8x4)	MLQ-BTC (6x4)
압축률	4:1	6:1	6:1	6:1	6:1
lena	33.51	34.56	34.54	34.63	34.82
porait	36.02	37.00	36.66	36.65	37.73
peppers	35.26	36.43	36.53	36.44	37.23
fruits	34.00	33.86	34.95	35.08	35.82
house	36.80	39.89	38.12	40.42	40.60

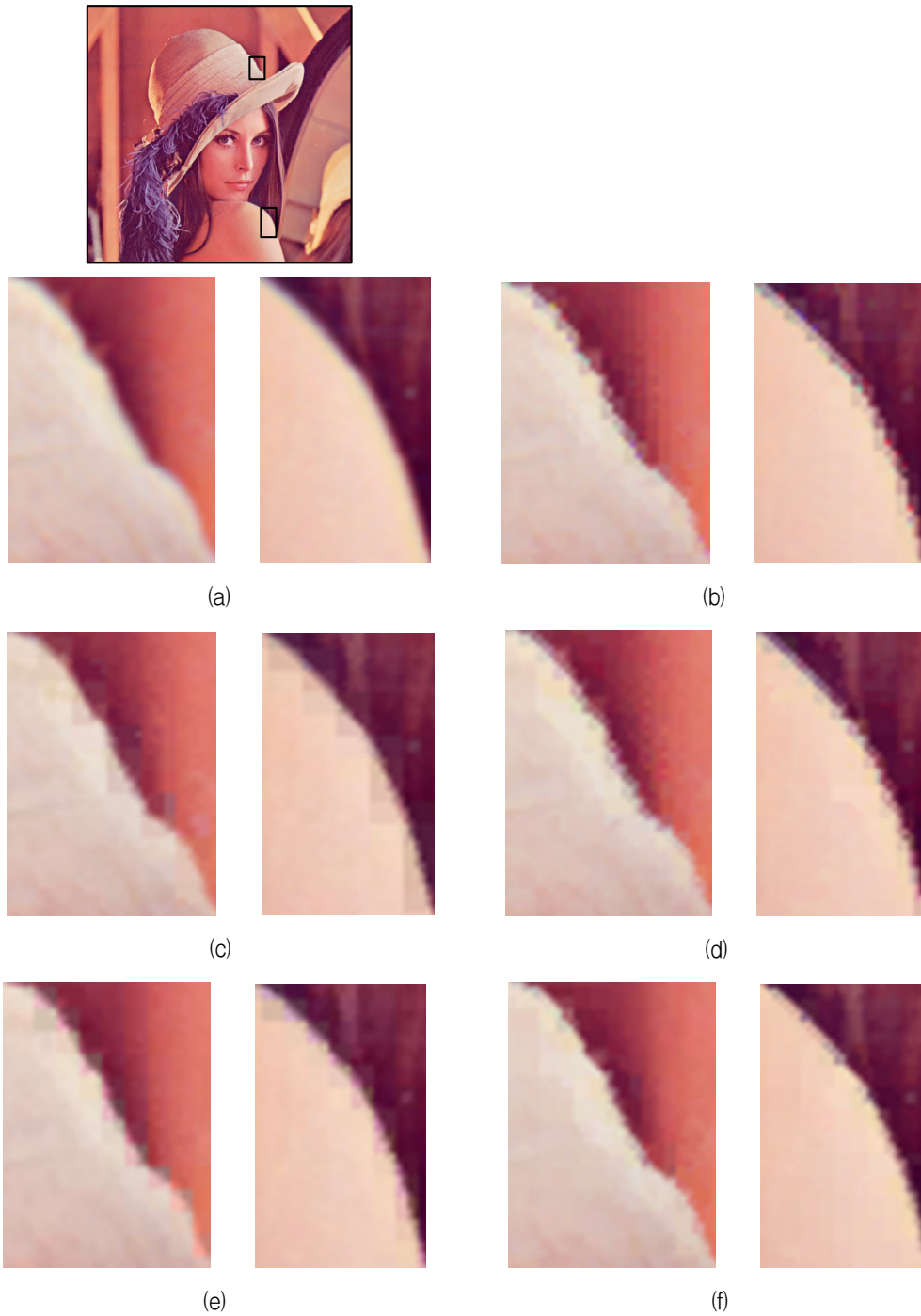


그림 7. 화질 비교

(a) 원본 영상, (b) AMBTC, (c) AQC, (d) CE-BTC, (e) FBTC, (f) MLQ-BTC

Fig. 7. Comparison of the subjective visual quality.

(a) original image, (b) AMBTC, (c) AQC, (d) CE-BTC, (e) FBTC, (f) MLQ-BTC

가 4:1이고, AQC, CE-BTC와 FBTC는 6:1이다. 본 장에서 사용된 AQC는 4-level 압축 방법을 적용하였고, 서브 그레이의 비트맵은 Y의 AQC-비트맵을 적용하였다. 그리고 min. 화소 값과 양자화 구간 등은 압축 없이

적용하여, 6x4 블록에서 압축비가 6:1이 되도록 구성하였다.

PSNR 결과를 비교해 보면 lena 영상에서 MLQ-BTC의 PSNR 결과가 AMBTC 대비 1.31 dB 높

은 PSNR을 확인 할 수 있었으며, AQC, CE-BTC, FBTC와 비교해도, 각 0.26 dB, 0.28 dB, 0.19 dB 높은 PSNR 결과를 확인 할 수 있었다. 5가지 영상의 평균 PSNR을 비교한 결과에서도, MLQ-BTC가 AMBTC 대비 2.12 dB 높은 PSNR을 확인 할 수 있었으며, AQC, CE-BTC 그리고 FBTC 와 비교해도, 각 0.89 dB, 1.08 dB 그리고 0.59 dB 높은 PSNR 결과를 확인 할 수 있었다. 그림 7은 lena 영상의 edge 부분에서 AMBTC, AQC, CE-BTC 그리고 FBTC 와 MLQ-BTC를 비교한 부분이다. MLQ-BTC의 edge 영상이 AMBTC, AQC, CE-BTC 그리고 FBTC의 edge 영상과 비교하여, MLQ-BTC 결과가 원본 영상과 가장 유사한 edge가 형성되어 있음을 확인 할 수 있다.

VI. 결 론

본 논문에서는 6:1의 높은 압축비로 낮은 압축손실을 얻기 위해서, Advanced 2-level 과 Quasi 8-level를 사용한 MLQ-BTC를 제안하였다. MLQ-BTC는 Advanced 2-level BTC 결과와 Quasi 8-level BTC 결과를 복호하여 원본 데이터와 비교하여 원본 데이터와 가장 오차가 적은 방법을 선택하는 기법으로, Quasi 8-level BTC 압축 방법은 블록내의 절반의 영역은 서브 그레이들의 비트맵 대신 Y 비트맵을 적용하는 방법을 사용하였고, 나머지 절반의 영역은 Y 비트맵을 적용하여 복호된 4개의 인접 픽셀 데이터와 비교하여 가장 오차가 적은 데이터를 선택하는 방법을 사용하여, 6:1의 압축영상을 얻을 수 있었다. Advanced 2-level는 Y 비트맵을 적용하여 비트맵의 총량을 1/3 으로 줄이고, 서브 그레이의 양자화 데이터는 8-level AQC 방법을 적용하여 양자화 데이터를 압축하였다. MLQ-BTC 알고리즘의 성능 분석을 위해서, 다양한 영상에서, AMBTC, AQC, CE-BTC 그리고 FBTC의 PSNR 결과와 MLQ-BTC의 PSNR 결과를 비교 분석하였다. MLQ-BTC의 PSNR 결과를 AMBTC 결과와 비교해보면 MLQ-BTC가 2.12 dB 높은 PSNR 결과를 얻었으며, AQC, CE-BTC 그리고 FBTC의 PSNR 결과와 비교해도 0.89 dB, 1.08 dB 그리고 0.59 dB 높은 결과가 확인되어, 본 논문에서 제안하는 MLQ-BTC의 효율성을 확인할 수 있었다.

REFERENCES

- [1] J. W. Han, M. C. Hwang, and S.J. Ko, "Vector quantizer based block truncation coding for color image compression in LCD overdrive," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 4, pp. 1839-1845, Nov. 2008.
- [2] J. Someya, "A new LCD-Controller for Improvement of Response Time by Compression FFD," *SID Symposium Digest of Technical Papers*, Vol. 34, pp. 1346-1349, May 2003.
- [3] E. J. Delp and O. R. Mitchell, "Image compression using block truncation coding," *IEEE Transactions on Communication*, vol. 27, no. 9, pp. 1335-1342, Sep. 1979.
- [4] M. D. Lema and O. R. Mitchell, "Absolute moment block truncation coding and its application to color images," *IEEE Trans. Commun.*, vol. COM-32, pp. 1148-1157, Oct. 1984.
- [5] J. Someya, A. Nagase, N. Okuda, K. Nakanishi, and H. Sugiura, "Development of single chip overdrive LSI with embedded frame memory," *SID Symposium Digest of Technical Papers*, vol. 39, no.1, pp.464-467, May 2008.
- [6] J. Wang and J. W. Chong, "High performance overdrive using improved motion adaptive codec in LCD," *IEEE Transactions Consumer Electronics*, vol. 55, no. 1, pp. 20-26, Feb. 2009.
- [7] J. Wang, K. Y. Min, Y. C. Jeung, and J. W. Chong, "Improved BTC using luminance bitmap for color image compression," in *Proceedings of the 2009 2nd International Congress on Image and Signal Processing, CISP'09*, 2009.
- [8] J. Wang, K. Y. Min, and J. W. Chong, "Cost Effective Block Truncation Coding for Color Image Compression," *AISS*, 2(3):91-98, 2010.

저 자 소 개



조 문 기(정회원)

1998년 인하대학교 전자재료
공학과 학사졸업

2002년 인하대학교 전자재료
공학과 석사졸업

2002년~2004년 하이닉스
반도체 주임 연구원

2004년~현재 매그나칩 반도체 책임 연구원

2009년~현재 인하대학교 전자공학과 박사과정

<주관심분야 : LCD/AMOLED T-CON 개발, 디
지털 아날로그 회로설계, FPGA 설계, ODC 용 압
축 알고리즘 개발>



윤 영 섭(정회원)

1975년 서울대학교

금속공학과 학사졸업

1977년 한국과학원

재료공학과 석사졸업

1988년 Univ. Southern California
전자공학과 박사졸업

1987년~1988년 Oklahoma State University
대우교수

1988년~1989년 UCLA Device Research Lab.
연구원

1989년~1992년 삼성전자 기흥 반도체 연구소
수석연구원

1992년~현재 인하대학교 전자공학과 교수

<주관심분야 : ULSI DRAM을 위한 신물질 개
발 강유전성박막 Pyroelectric 센서 SAW device,
회로설계>