

최적화된 허프만 엔트로피 코딩을 적용한 JPEG 영상 압축 기법의 구현에 관한 연구

이종수 · 이남용
컴퓨터공학과

<요 약>

국제표준인 JPEG영상압축기법에서는 4가지 유형의 프로세스로 베이스라인, 확장 DCT 기반, 무손실, 계층적 프로세스들을 제시하고 있는데, 이 중에서 베이스라인 프로세스들이 널리 사용되고 있다. 이 베이스라인 프로세스들을 JPEG 전문가 그룹(Independent JPEG Group)이 1992년에 구현하는데 그 구현기법은 압축 부호화과정에서 허프만 엔트로피 코딩을 최적화 하고자 전체적인 영상 데이터를 3번 스캔(Scan)하고 있다. 본 논문에서는 전체적인 영상 스캔을 2번하여 압축 부호화할 수 있는 구현기법을 제시하며, 이 기법에 의해 영상압축에 소요되는 시간을 약 2.8% 감소시킬 수 있음도 보여준다.

A Study on Implementation Technique of the JPEG Image Compression Algorithm taking the Optimized Huffman Entropy Coding

Jong-Soo Lee · Nam-Yong Lee
Department of Computer Engineering

<Abstract>

The International Standard IS10918-1 called JPEG offers 4 types of processes,

* 본 연구는 96년도 울산대학교 학술연구비 지원에 의해 수행되었음

baseline, extended DCT-based, lossless, and hierarchical. Among these processes, the baseline processes which is used widely for still image compression are implemented by the Independent JPEG Group of image compression coding experts in 1992. The implementation, to optimize the Huffman entropy coding while encoding an image, scans the entire image data three times. In this paper, we propose to use an implementation technique scan the entire image two times and we show that we can save about 2.8% of time required to encode image.

I. 서 론

영상정보의 처리, 저장, 그리고 전송을 효율적으로 하기 위해서는 방대한 양의 영상 데이터를 압축 부호화 하여야 한다. 영상 압축 부호화기법의 국제 표준인 JPEG(Joint Photographic Expert Group Standard)은 정지영상(Continuous Tone Still Image) 압축기법에 대한 국제표준으로 국제 표준화 기구인 ITU(CCITT)와 ISO에 의해 만들어 졌다.[1][2] 이 표준은 이진 영상을 제외한 흑백 영상이나 컬러 영상 등의 한 프레임 정지영상을 국제 표준방식으로 압축하는데 요구되는 사항들을 명시하고 있다.

JPEG기법은 크게 손실이 없는 압축 방법(Lossless Compression, Predictive Coding)과 손실이 있는 압축방법(Lossy Compression, Transform Coding)으로 나누어지며 손실이 있는 압축방법은 3가지 프로세스들로 구분하고 있다. 가장 기본이 되는 손실 압축 방법은 베이스라인 프로세스(Baseline Process)라 하며 이는 영상정보를 부호화 및 복호화하는데 필요한 최소한의 요구사항들로 이루어진다. 이에 비해 확장 DCT(Discrete Cosine Transform)기반 프로세스나 계층적 프로세스는 베이스라인 프로세스에서 규정하고 있는 기본적인 구성과 기능, 정밀도, 오차 등에 대한 요구 사항들 등에 보다 융통성을 부여함으로써 영상정보의 압축 효율과 영상품질을 향상시키고 있다.

현재 인터넷에서 영상파일 전송에 널리 사용되고 있으며, 많은 PC 응용 프로그램들에서 영상을 압축 저장하는데 사용되고 있는 프로세스는 베이스라인 프로세스로서 영상압축 전문가 그룹(Independent JPEG Group)에 의해 1992년에 구현되었다. 이 베이스라인 프로세스의 구현은 허프만 엔트로피 코딩을 최적화 시키기 위해 영상전체를 3번 스캔하고 있는데 이를 2번 스캔하도록 구현하여 영상압축에 소요되는 시간을 단축시킬 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 먼저 1992년에 구현된 기법(JPEG 92 구현 기법)에 대해 상세히 설명하고, 3장에서는 최적화된 허프만 엔트로피 코딩에 의한 영상압축의 구현이 2번의 영상스캔으로 가능함을 보여준다. 또한 2번의 영상스캔으로 영상압축에 소요되는 시간을 단축시킬 수 있음도 보여준다.

II. JPEG92의 구현기법

JPEG92는 크게 초기화 및 방법선택(method selection), 영상 압축, 메모리 해제 및 종료 과정으로 나누어진다[그림 1]. 초기화 및 method selection은 사용자에 의해 입력되는 명령

어 라인(Command Line)의 스위치를 처리하여 사용자의 지시대로 영상압축이 수행되도록 하며, 입력영상 파일 포맷에 의한 파일 읽기 방법 및 출력파일의 저장방법을 결정한 후, 입력영상의 크기와 제공된 상대적 샘플링 인수에 의해 압축영상 성분들의 크기를 계산한다. 그리고 영상압축시에 적용될 방법(method)들을 선택한다.

영상 정보 압축은 앞의 과정에서 선택된 방법을 사용하여 영상데이터를 downsampling, 최소부호화 단위(MCU) 추출, DCT변환, 양자화, 엔트로피 최적화(entropy optimization), 엔트로피 인코딩(entropy encoding)을 거쳐 영상정보를 부호화하며 그 결과를 출력 파일에 저장한다.

메모리 해제 및 종료는 위의 두 과정에서 사용하기 위해 할당된 메모리를 해제하고 프로그램을 종료시킨다.

2. 1 초기화 및 방법선택

초기화 및 방법선택(method selection)은 switch처리, 입출력 파일 포맷과 입출력 방법 결정, 입력 영상의 차원과 상대적 샘플링, 그리고 방법선택과정이 있다.

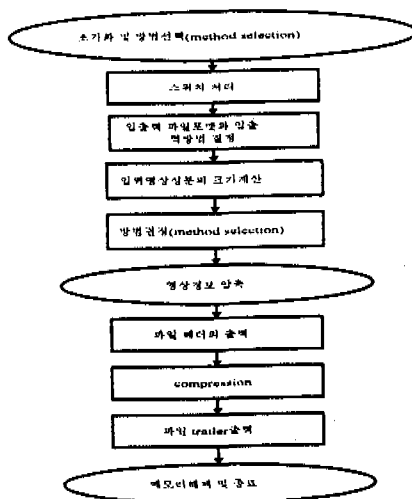


그림 1. JPEG92 전체 흐름도.

2. 1. 1 스위치 처리

JPEG92에서는 사용자가 명령어 라인(command line)을 다음과 같은 2가지 형식중 하나를 선택하여 입력시키도록 한다.

```

cjpeg [선택사항] 입력파일명 출력파일명
cjpeg [선택사항] 입력파일명
    
```

위와 같은 명령어 라인에서 선택사항(option)부분에 다음과 같은 스위치가 올 수 있다.

-arithmetic : 엔트로피 코딩에서 지원하는 허프만 코딩과 산술 코딩중 산술 코딩을 수행하도록 하는 명령어로 JPEG92에서는 산술 코딩이 구현되지 않아 실제적으로 사용될 수 없는 스위치이다.

-debug, -verbose : JPEG 버전과 저작권 등을 화면에 출력하도록 한다.

-grayscale, -greyscale : 출력파일의 color space를 흑백 모드로 전환시켜 준다.

-maxmemory size character : 영상압축 과정에서는 메모리 할당을 필요로 하는 데 이 스위치는 사용될 최대 메모리 용량을 제한한다. 메모리 용량의 크기는 size에 입력하며 size단위는 character에 M과 K를 입력하여 Mbyte와 Kbyte 단위임을 표시한다.

-noninterleave : 이 스위치는 영상데이터를 스캔하는 방식으로 끼워 넣기 스캔과 끼워 넣지 않기 스캔중 끼워 넣지 않기 스캔을 지원하도록 한다.

-optimize, -optimise : 허프만 엔트로피 최적화를 수행하도록 한다.

-quality N : 압축영상의 품질을 결정하는 값을 입력한다.

-qtables filename : JPEG92에서는 영상압축 과정을 수행하기 위하여 DCT변환에 이어 양자화를 수행하는 데 이때 양자화 테이블로 JPEG92의 디폴트 테이블이외에 다른 양자화 테이블을 사용하고자 할 때 테이블 입력방법으로 사용한다.

-restart value character : 영상압축과정이 독자적으로 수행되는 간격(interval)을 지정하여 value의 값은 0에서 65525이다. character는 b또는 B로 value의 값을 정수형에서 unsigned integer형으로 변환한다.

-sample value : 샘플링 인수를 입력한다.

-smooth value : value가 0에서 100사이의 값을 갖으며 압축 영상성분을 만드는 과정에 적용시킨다.

-targa : 입력파일의 포맷을 targa파일로 한다.

2. 1. 2 입출력 파일의 포맷과 입출력 방법 결정

입력 파일 포맷은 명령어 라인에 주어진 입력파일의 첫 번째 문자를 읽어 그 문자가 G이면 gif 파일, P이면 ppm파일, R이면 rle파일이며, 모두 0인 비트로 되어 있으면 targa 파일이다. 이 입력파일의 포맷에 의해 영상을 읽는 방법이 결정된다. 명령어 스위치의 지시대로 영상 압축이 수행되며 그 결과가 JPEG 파일 포맷으로 출력된다.

2. 1. 3 입력 영상 성분의 크기 계산

입력 영상의 크기와 명령어 라인이나 JPEG표준 디폴트 값에 의해 주어지는 샘플링 인수를 사용하여 JPEG압축이 적용되는 영상 성분들의 수평크기 x_i 와 수직크기 y_i 를 계산하는 과정으로 아래의 수식(1)에 의해 구해진다.[3]

$$x_i = \lceil X \times \frac{H_i}{H_{\max}} \rceil \quad (1)$$

$$y_i = \lceil Y \times \frac{V_i}{V_{\max}} \rceil$$

X, Y : 입력 영상의 수평 및 수직크기

H_i, V_i : 수평, 수직의 샘플링 인수

H_{max}, V_{max} : 최대 수평 및 수직 샘플링 인수

$\lceil \rceil$: 상한정수함수(ceiling function)

2. 1. 4 방법 선택

방법선택과정에는 칼라 표현 변환, 엔트로피 인코딩 방법선택, 그리고 파이프라인 제어 기 방법선택이 있다.

2. 1. 4. 1 칼라 표현 변환 방법선택

칼라 표현 변환 방법선택은 입력 영상의 색공간(color space)과 JPEG압축이 적용될 색공간을 상호 비교하여 입력영상 색공간을 JPEG 색공간으로 변환하는 방법을 선택한다. 아래의 수식(2)은 RGB 색공간을 YUV 색공간으로 변환하는 수식이며, 입력영상 파일 포맷이 GIF이고 이를 JPEG이 디폴트로 사용하는 YCbCr로 색변환을 하고자 할 때는 아래 수식(3)을 사용한다.[4]

$$\begin{aligned} Y &= 0.3R + 0.6G + 0.1B \\ V &= R - Y \\ U &= B - Y \end{aligned} \quad (2)$$

$$\begin{aligned} Y &= 0.3R + 0.6G + 0.1B \\ Cb &= (U/2) + 0.5 \\ Cr &= (V/1.6) + 0.5 \end{aligned} \quad (3)$$

2. 1. 4. 2 엔트로피 인코딩 방법선택(Entropy encoding method selection)

엔트로피 인코딩은 허프만 코딩(Huffman coding)과 산술 코딩이 있는데, 이 두가지 코딩 방법 중 하나를 선택하는 과정이다. 베이스라인 프로세스들은 허프만 코딩만을 사용하며, 확장 DCT기반 프로세스들은 허프만과 산술코딩 모두를 사용할 수 있다.

2. 1. 4. 3 파이프라인 제어기 방법선택(Pipeline controller method selection)

파이프라인 제어기 방법선택은 영상 압축과정에서 스캔을 한 번할 것인지 아니면 여러 번에 걸쳐 수행할것이지와 엔트로피 인코딩 최적화(entropy encoding optimization)수행여부에 따라 4가지 제어기로 구분되며 JPEG92에서는 single scan만을 지원하는 끼워 넣기 스캔방식만을 구현하고 있다.

2. 2 영상 압축

영상의 압축과정은 헤더의 출력부분, compression부분, 파일 트레일러(file trailer)의 출력부분으로 나누어 진다.

2. 2. 1 파일 헤더 출력

JPEG파일 포맷의 파일 헤더를 만들기 위해 다음의 것들이 순서대로 출력된다.

- 1) SOI(Start Of Image marker)표시자를 출력한다.
- 2) APP0을 출력한다. 이 부분에 JPEG 압축 프로그램의 버전이 삽입된다.
- 3) JPEG표기자 세그먼트(Marker Segment에서 사용되는 양자화 테이블인 DQT를 출력한다.
- 4) SOF(Start Of Frame)표시자를 출력한다.

2. 2. 2 압축(Compression)

압축 부분은 2. 1에서 선택된 방법들을 사용하여 영상데이터를 실질적으로 압축 부호화하는 과정이다. 이 과정은 선택된 제어기에 의해 크게 좌우되는데 JPEG92에서 지원하는 제어기는 `single_controller` 제어기와 `single_eopt_controller` 제어기가 있다.

2. 2. 2. 1 Single_controller 제어기

Single_controller 제어기는 [그림 2]에서 보는 바와 같이 입력 영상으로부터 MCU줄 단위로 샘플들을 취하여 에지 확장 및 downsampling과정을 거치게 한다. MCU줄이란 8x8 영상블럭들을 포함하는 최소부호화단위로 구성되는 줄(row)이다. 다음에 이 MCU줄로부터 하나의 MCU가 추출되고 이 MCU내의 모든 블럭에서 DCT변환, 양자화, 그리고 엔트로피 인코딩이 수행된다.

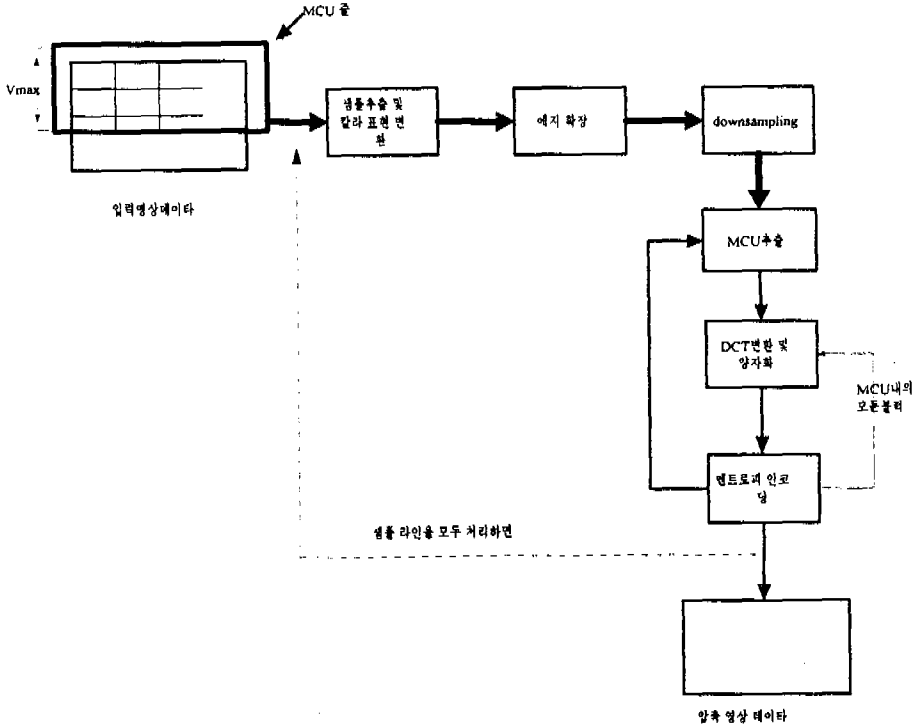


그림 2. single_controller 제어기의 흐름도.

2. 2. 2. 1. 1 샘플 추출 및 칼라 표현 변환

입력 영상으로부터 높이가 최대 샘플링 인수 V_{max} 이고, 폭이 입력영상과 같은 MCU줄 크기의 샘플들을 추출한다. 추출된 샘플들은 입력 영상 칼라 공간에서의 영상 정보를 나타내며, 이 샘플들은 JPEG압축기법이 적용될 칼라 공간 샘플들로 변환된다. 샘플들은 8비트 또는 12비트가 될 수 있으며, 베이스라인 프로세스에서는 8비트 샘플들만 압축한다. 그리고 변환된 샘플들은 변환전과 마찬가지로 MCU줄 크기를 갖는다.

2. 2. 2. 1. 2 에지 확장(Edge expansion)

JPEG의 영상압축은 MCU단위로 수행되므로 추출된 MCU줄이 MCU단위로 일치하지 않은 경우 오른쪽 에지샘플들이 확장되어 하나의 MCU단위를 구성하도록 한다. 마찬가지로 마지막 MCU줄의 높이가 MCU 높이와 일치하지 않을 경우 입력영상의 아래 에지가 확장된다.

2. 2. 2. 1. 3 Downsampling

Downsampling은 압축될 영상성분들에 대한 정해진 샘플링 인수들에 의해 영상성분들의 크기가 결정된다. [그림 3]은 샘플링 인수들이 수평 방향으로 $H_1=2, H_2=1, H_3=1$ 이고 수직방향으로 $V_1=2, V_2=2, V_3=2$ 인 경우 YCbCr성분들이 4 : 2 : 2 샘플링 됨을 보여준다.

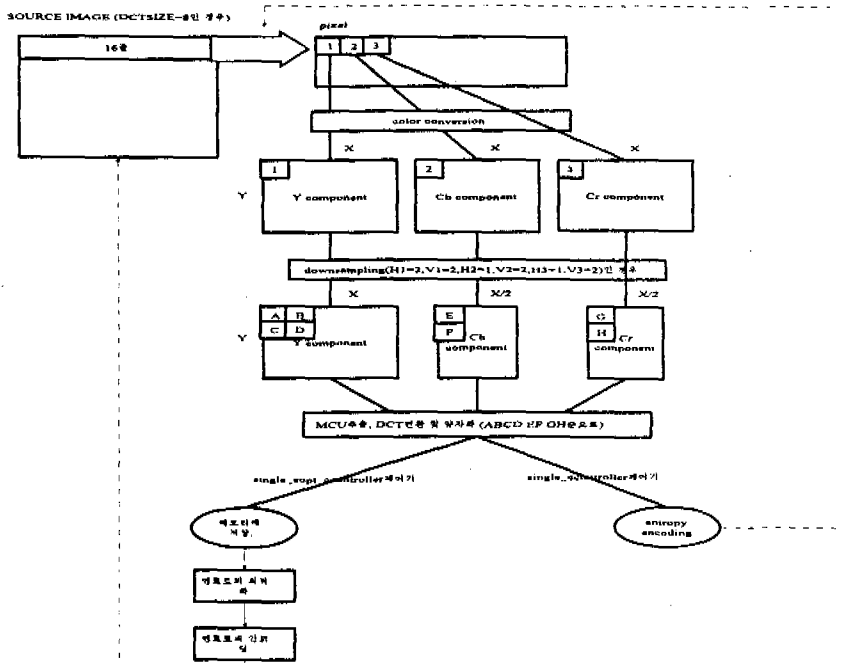


그림 3. downsampling 과정

2. 2. 2. 1. 4 MCU 추출

MCU추출은 2.1의 스캔방식에 의존한다. 끼워 넣기 스캔인 경우 각 성분의 블록들이 그 성분에 대한 샘플링 인수와 비례하는 개수만큼의 블록들로 MCU가 구성된다. 끼워 넣지 않기 스캔인 경우 각 성분에 대해 MCU가 존재하므로 하나의 MCU는 오직 하나의 성분 블록을 갖는다.

2. 2. 2. 1. 5 DCT 변환 및 양자화

DCT(Discrete cosine tranform)과정은 추출된 MCU내의 블록 하나 하나에 대해 먼저 DCT변환되고 변환 계수들은 양자화 테이블에 의해 양자화된다.

2. 2. 2. 1. 6 엔트로피 부호화

베이스라인 프로세스의 엔트로피 인코딩은 허프만 부호화를 사용하여 양자화된 계수를 허프만 테이블에 적용, 부호화된 영상 정보를 구하는 과정이다. 이 과정을 거친 영상 정보는 2.1에서 선택된 출력 파일에 저장되어 진다.[5][6]

2. 2. 2. 2 Single_eopt_controller제어기

Single_eopt_controller제어기는 single_controller제어기와 수행과정이 동일하나 단지 엔트로피 최적화를 한다는 것이다. 엔트로피 최적화과정은 허프만 엔트로피 코딩을 수행하기 위한 전 처리과정으로 입력영상에 가장 적합한 허프만 DC, AC테이블들을 작성한다. 다음에 이 테이블을 사용하여 엔트로피 코딩이 수행되며, 엔트로피 최적화과정을 수행하지 않은 경우에 비해 압축율이 향상된다는 이점이 있다. 그러나 수행시간은 지연된다는 단점이 있다. 아래 [그림 4]는 single_eopt_controller제어기에 대한 흐름도를 나타낸다. 이 흐름도에서 보는 바와 같이 샘플추출 및 칼라표현변환, 예지확장, downsampling과 MCU 추출 및 DCT변환, 양자화과정을 수행함은 single_controller제어기와 동일하다. 그러나 생성된 양자화계수는 single_controller제어기의 엔트로피 부호화과정을 거치지 않고 메모리에 저장된다. 이 과정은 MCU줄 단위로 반복된다. 다음은 양자화계수 전체로부터 최적화된 허프만 AC 및 DC테이블을 만들어 낸다. 마지막으로 최적화된 허프만 AC 및 DC테이블들을 사용하여 양자화계수들에 대한 압축부호화된 영상 데이터를 얻는다.[7][8]

2. 3 메모리 해제 및 종료

프로그램의 종료를 위해서는 영상데이터 압축과정에서 할당된 전체 메모리를 해제하여야 하며, 이를 위해 JPEG92에서는 메모리를 할당시 이전에 할당된 메모리 블록들은 상호연관성을 갖도록 linked list를 형성한다. 아래 [그림 5]에서와 같이 최근에 할당된 메모리 블록의 헤더가 바로 전에 할당된 메모리블록의 헤더를 지시하고 있다면 메모리블록 해제시 가장 최근에 할당된 메모리블록부터 반복적으로 해제되어 할당된 메모리 블록들이 모두 해제된다. [9]

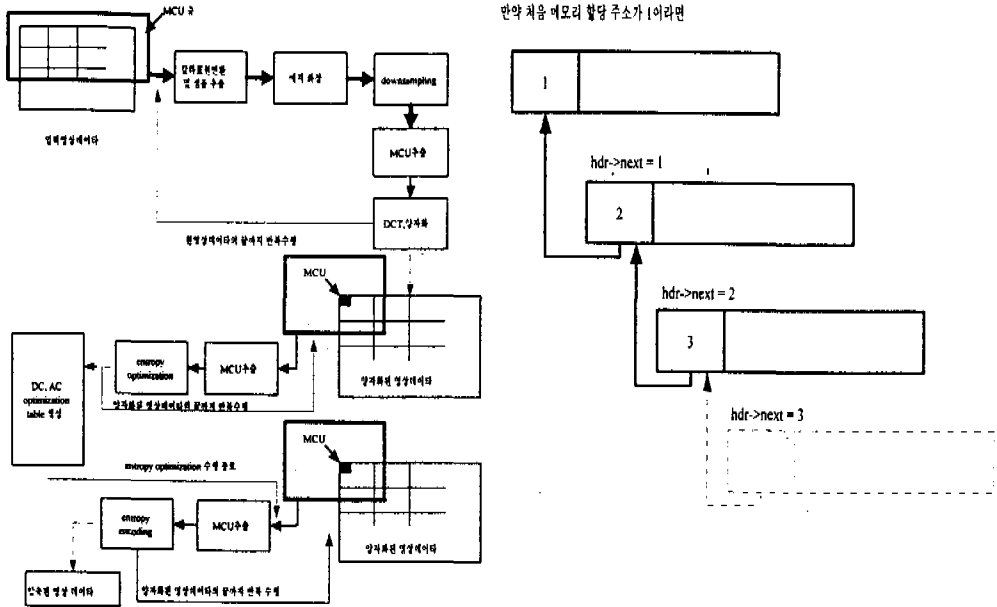


그림 4. single_eopt_controller 제어기의 흐름도그림 5. 메모리 블록 할당기법 (1,2,3: 메모리 주소)

III. 개선된 single_eopt_controller 제어기

[그림 4]에서 보는 바와 같이 기존의 single_eopt_controller 제어기는 샘플추출 및 칼라 표현 변환에서 downsampling과정까지 입력 영상 데이터를 최대 샘플링 인수와 동일한 열 단위로 처리하여 MCU추출과정에서 이를 MCU단위로 나누어 DCT변환 및 양자화를 거쳐 메모리에 저장한다. 이러한 과정을 전체 영상 데이터의 끝까지 반복 수행하며 메모리에는 입력 영상 데이터를 양자화한 결과인 양자화된 영상 데이터가 저장된다. 이 양자화된 영상 데이터는 MCU단위로 추출되어 허프만 엔트로피 최적화과정을 수행하며 이 과정은 양자화된 영상데이터를 전체적으로 한 번 읽을 때까지 반복 수행되며 그 결과 AC 및 DC 테이블을 생성된다. 이렇게 생성된 AC 및 DC테이블을 사용하여 엔트로피 부호화과정을 수행하게 되는데 이때에도 메모리에 저장되어 있는 양자화된 영상데이터를 전체적으로 한번 읽어 압축된 영상데이터를 MCU단위로 출력파일에 저장하게 된다. 위와 같은 과정은 전체적인 영상 데이터를 3회 읽어야 하나 제어기의 흐름을 수정하여 전체적인 영상데이터를 2회 읽어 압축부호화 할 수 있다.

3. 1 제어기 흐름도의 수정

수정된 single_eopt_controller 제어기의 흐름도는 [그림 6]과 같다. 이 흐름도에서는 입력영상에 최적인 허프만 AC 및 DC테이블들이 전체적인 영상을 한 번 읽어 작성할 수 있다. 즉, DCT변환 및 양자화 후 생성되는 양자화 계수들을 메모리에 저장, 허프만 엔트로피 최적화과정을 거쳐 최적의 허프만 AC 및 DC테이블들을 작성하도록 한다. 다음에 양자화계수들에 대한 엔트로피 코딩을 최적화된 허프만 AC 및 DC테이블들을 사용하여 입력영상에 대한 압축부호화데이터를 최종적으로 출력한다. 즉, 입력영상을 전체적으로 한 번 읽고 다음은 양자화계수들을 전체적으로 한 번 읽어 영상을 압축할 수 있다.

본 논문에서는 앞에서 설명한 과정중 샘플추출 및 칼라표현변환과정에서 허프만 엔트로피 최적화까지의 과정을 하나로 묶어 입력영상데이터를 한 번 읽고 메모리에 양자화된 영상 데이터를 저장한 후 허프만 엔트로피 부호화를 수행하여 전체적인 부호화 과정에서 영상정보를 2회 읽는 형태로 기존의 single_eopt_controller제어기를 개선하였다. 개선된 single_eopt_controller제어기의 수행 절차는 아래의 [그림 6]과 같다.

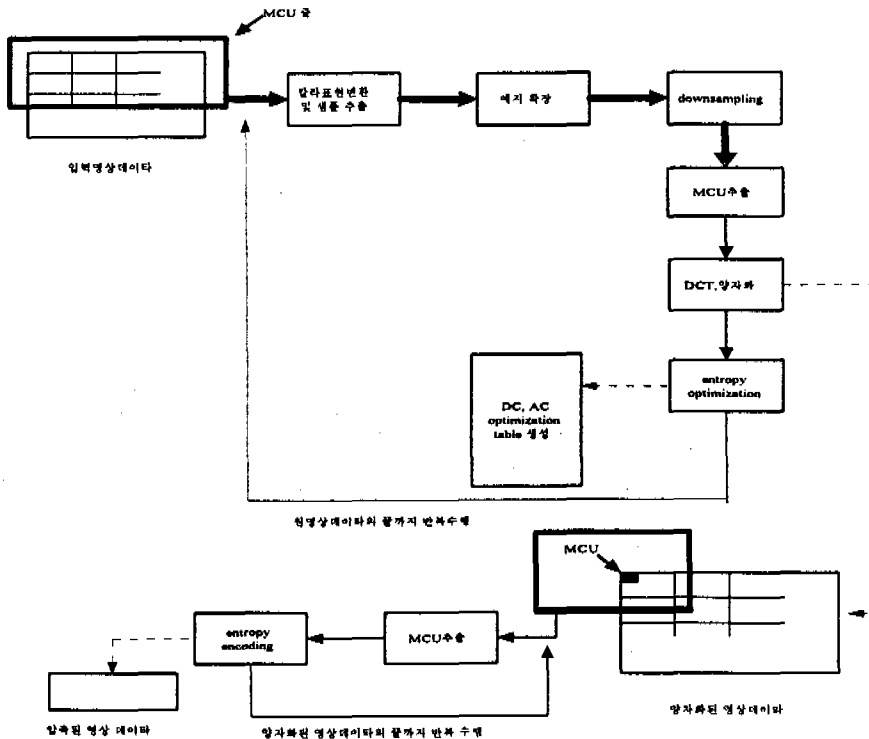


그림 6. 개선된 single_eopt_controller제어기 흐름도

3. 2 개선된 제어기의 성능

본 논문에서는 입력영상을 엔트로피 최적화하지 않고 영상 압축을 수행하는 single_controller제어기와 엔트로피 최적화를 사용하는 single_eopt_controller제어기, 그리고 개선된 single_eopt_controller제어기를 사용하여 영상 압축을 수행한 결과는 [표 1]과 같다. 영상압축에 사용한 영상데이터의 크기가 720x576인 JPEG표준시험데이터 boats, blackboard, 그리고 flower들이다. 개선된 single_eopt_controller제어기에 대해 수행시간이 [표 1]에서 보는 바와 같이 약 2.8%빨라짐을 알 수 있다. 이 실험에서 사용된 영상은 700x576의 크기를 가지는 ISO표준영상데이터이다. 사용된 컴퓨터는 50MHz동작되는 IBM PC 486이다.

image	원영상의 크기 (bytes)	압축된 영상부호의 크기 (bytes)		압축시 걸리는 시간(1/100sec)		
		single_controller제어기	single_eopt_controller제어기	single_controller제어기	기존 single_eopt_controller제어기	개선된 single_eopt_controller제어기
boats	414,720	63,501	61,781	2642	3037	2944
barbara	"	78,730	77,209	2729	3159	3070
blackboard	"	52,239	49,819	2543	2955	2867
flower	"	86,623	84,801	2604	3043	2933

표 1. 각 제어기의 영상압축 수행시간 및 영상압축부호화 크기

IV. 결 론

본 논문에서는 베이스라인 프로세스들을 구현한 JPEG92 기법을 분석하였고, 구현기법의 개선으로single_controller제어기의 흐름도 수정을 제안하였다. 또한 개선기법의 성능 분석 실험도 실시하여 이 기법에 의해 영상압축시간을 약 2.8% 단축시킬 수 있음을 보여주었다. 영상압축시간을 단축시킬 수 있는 이유는 전체적인 영상데이터를 처리하는 횟수를 3회에서 2회로 줄일 수 있기 때문이다. 앞으로 연구는 점진적으로 영상이 개선되는 Progressive JPEG영상압축기법 구현에 개선된 기법을 적용하여 영상압축시간을 단축하는 것이다.

참 고 문 헌

[1] William B. Pennebaker, Joan L. Mitchell
 "JPEG Still image data compression standard" VNR ISBN 0-442-01272-1, 1993
 [2] A. K. jain. "Fundamentals of Digital Image Processing. Englewood Cliffs, NJ:

Prentice Hall,1989

- [3] George Wolberg, "Digital Image Warping", Pub. by IEEE Computer Society Press, Los Alanitos, CA. ISBN 0-8186-8944-7, 1990.
- [4] R. W. G. Hunt. "Measuring Color." New York: Halsted Press, 1987
- [5] K. R. Rao and P. Yip. "Discrete Cosine Transform." New York: Academic Press, 1990
- [6] W. Chen, C. H. Smith, and S. C. Fralick. "A Fast Computational Algorithm for the Discrete Cosine Transform." IEEE Trans. on Commun. COM-25(9):1004-9, Sept. 1977
- [7] G. G. Langdon. "An Introduction to Arithmetic Coding." IBM J. Res. Develop. 28(2):135-49, Mar. 1984
- [8] S. W. Golomb. "Run-Length Encodings." IEEE Trans. Info. Theory. IT-12:399-401, Jul. 1966
- [9] IBM Corporation. "PC Intern System Programming." 1992