

碩士學位論文

MPEG-2 Video의 실시간 처리를 위한 브라우저 설계 및 구현



제주대학교 중앙도서관
JEJU NATIONAL UNIVERSITY LIBRARY

濟州大學校 大學院

情報工學科

姜 權 益

1999年 12月

MPEG-2 Video의 실시간 처리를 위한 브라우저 설계 및 구현




指導教授 金 壯 亨

姜 權 益

이 論文을 工學 碩士學位 論文으로 提出함



姜權益의 工學 碩士學位 論文을 認准함

審査委員長	안 기 중	
委 員	김 권	
委 員	李 尚 俊	

濟州大學校 大學院

1999年 12月

The Design and Implementation of Real-Time Processing Browser for MPEG-2 Video

KwonIk Kang

(Supervised by professor JangHyeong Kim)



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING**

**DEPARTMENT OF INFORMATION ENGINEERING
GRADUATE SCHOOL
CHEJU NATIONAL UNIVERSITY**

1999. 12.

목 차

SUMMARY

I. 서론	1
II. MPEG-2 Video	3
1. MPEG-2 Video	3
2. MPEG-2 비디오의 구조	5
3. MPEG-2 Video를 위한 Algorithm	7
4. MPEG-2 부호화기 특징 ..	13
III. MPEG-2 Video에서 저 비트율, 고품질 개선 방안	16
1. MPEG-2 Video의 화질 개선 방안	16
2. GoP 목표 비트량의 결정	21
3. 슬라이스 목표 비트량과 양자화 계수 결정	22
IV. MPEG-2 비디오 브라우저 설계	24
1. 실시간 MPEG-2 비디오 압축을 위한 알고리즘	24
2. 실시간 MPEG-2 Video 브라우저의 인터페이스	27
3. 시스템 설계 ..	28
V. 결과 및 고찰	30
1. 구현 환경(H/W, S/W) ..	30
2. MPEG에서의 컷 검출 고찰	30
3. DC 계수와 흐름정보의 추정 고찰	34
4. 결과 내용 ..	36
VI. 결 론	40

【 참고 문 헌 】

SUMMARY

This paper suggests an indexing technique of video materials in the compressed format itself in order to reduce the cost and load which will be caused at the time of indexing the compressed video information.

The existent studies have been generally made on the analogue video sources. The present study is about how to store and classify the video data in the digital format, and further the study is to suggest a new technique improving the average image as well as maintaining uniformity of image among macro-blocks which are included within screens.

It aims to improve MPEG-2 video with low bit rates and high screen-quality.

The study underlies the process of encoding/decoding of video, and materializes and implements search range schema and motion estimation algorithm, but not uses search range block-matching technique. The study also evaluate the performance.

As a result, in the application using MPEG-2, software can implement a high-level application without employing hardware at the time of requiring a real-time processing.

Also in this study, considering the rapid increase of cost/effectiveness rate in the PC processors, MPEG-2 was implemented by real time software and was verified.

I. 서론

21세기는 컴퓨터의 처리 능력의 향상 및 통신의 고속화, 고 신뢰도 및 고도의 서비스 등 디지털의 발전을 통한 멀티미디어 시스템이 현실적으로 가까워졌다. 특히 멀티미디어 압축 처리부를 중점적으로 확대해야 할 필요성으로 인하여, 데이터의 비트율이 점점 더 낮아져 가는 권고안을 토대로 연구가 진행되었으며, 이러한 발전을 토대로 현재 정보 자원으로써 비디오의 실제 이용은 쉽고 효과적으로 사용할 수 있게 되었다. 현재 상태의 기술은 Logging 처리를 사용하여 데이터베이스에 입력된 구조화된 텍스트 방법으로 비디오 소스를 취급하고 있다. 따라서 이러한 전반적인 내용을 토대로 살펴볼 때, 효과적인 Tool이 없는 상태이다. 이러한 어려움을 극복하기 위해서 적어도 세 가지 타스크를 제공하는 툴이 필요하다^[5,8].

1. 비디오를 클립이라 불리는 적당한 인덱싱 단위로 분할한다.
2. 클립을 식별하기 위해서 저차원 영상 특성과 의미론적인 특성을 알아야 한다.
3. 질의로부터 결과를 비디오 자원에서부터 선택하는 검색과 브라우징이 필요하다.



이러한 내용으로 볼 때 지금까지의 연구는 압축되지 않은 비디오를 대상으로 했으며 일반적으로 아날로그 소스를 디지털화하여 왔다. 그러나 이 논문에서는 비디오 데이터를 압축된 디지털 형태로 저장하고 분류하는 기초 연구이며, 더 나아가 MPEG-2 Video에서 저 비트율, 고화질의 개선을 위하여 화면 안에 존재하는 매크로 블록들 사이 화질의 균일성을 유지하면서도 평균 화질을 향상시킨 새로운 방법을 제시하며 압축된 비디오 데이터로 비디오 파싱, 검색, 브라우징을 지원하는 툴을 개발한 연구 내용을 서술한다.

또한, 디지털 비디오 정보는 대부분이 고용량이기 때문에 압축을 피할 수 없고, 이에 보다 효과적이고 빠른 인덱싱 기법이 요구된다. 본 논문에서는 압축된 비디오 정보에서 인덱싱 시에 발생하는 많은 비용과 부하를 줄이기 위해 압축된 영역 자체에서의 비디오 인덱싱 기법을 제안한다.

먼저, 2장에서는 고품질, 저용량 매체인 MPEG-2 Video의 계층적 기본 구조와 알고리즘 및 부호화기의 구조를 소개한다. 3장에서는 MPEG-2 Video에서 저비트율, 고품질의 개선방안을 제시하였고, 4장에서는 비디오 브라우저를 설계하는데 있어서 가장 기본적인 핵심적인 MPEG-2 Video의 실시간 알고리즘에 대해 기술하며, 5장에서는 카메라를 통해 압축된 비디오의 내용을 기반으로 한 브라우징과 실험 결과에 대해 고찰한다.



II. MPEG-2 Video

1. MPEG-2 Video

MPEG-1 비디오가 CD등 디지털 축적 매체에 1.15Mbps의 저 비트율로 동화상을 저장하는데 반해, MPEG-2 비디오는 보다 고비트율의 방송, 통신, 축적 미디어에서 고화질의 동화상을 전송하거나 저장하는데 사용된다. 응용분야가 다양해진 만큼 충족시켜야 할 요구조건도 많아졌다.

세계 유수의 기업들이 이러한 조건을 만족시키기 위한 MPEG-2 비디오 알고리즘을 제안하여 91년 9월 최종적으로 30개의 방식이 일본의 JVC연구소에서 평가를 받았다. 그중 20개는 MPEG-1과 마찬가지로 DCT에 기초한 방식이었고, 5개는 서브밴드 부호화 방식이었으며, 나머지 5개는 웨이블릿 변환을 이용한 방식이었다.

평가 결과 여전히 DCT에 기반을 둔 방식들이 다소 우세하였고 MPEG-1 비디오와의 호환성도 고려하여, 결국 MPEG-2 비디오의 표준화의 방향도 DCT를 기반으로 하는 방식으로 결정되었다.

MPEG-2 비디오는 일종의 범용 압축 알고리즘으로, MPEG-1 비디오를 크게 확장 발전시키면서 많은 도구들을 마련하여 응용분야에 따라 이들을 적절히 선택 사용하도록 하고 있다.

압축효율의 향상을 위해 MPEG-2 비디오 필드단위의 처리, 움직임 추정과 보상방식, 양자화, DCT 계수와 주사방식 그리고 가변장 부호화 등 많은 부분들이 개선되었다.

MPEG-2 비디오의 범위는 매우 넓지만 응용분야마다 그 중 특정 해상도에 특정 기능까지만을 사용한다. 따라서 부호기(Decoder)와 복호기(Encoder)를 제작할 때의 편의를 위해 MPEG-2 비디오를 해상도와 기능에 따라 몇 가지로 분류하고 있다. 우선 화면의 해상도는 4개의 레벨로 분류된다. MPEG-1 비디오가 대상으로 하는 것과 같은 작은 화면인 Simple, 현행 TV 화면크기인 메인(Main), 유럽 고선명 TV HDTV의 화면크기인 High 1440, 미국 고선명 TV를 위한 규격인 High가 그것이다.

또 기능에 따라서는 5개의 프로파일로 나누어진다. 양방향 예측을 이용하는 B프레임을 제외

하여 구현을 용이하게 한 Simple, 많은 기능을 포함하여 대부분의 응용분야에 채택되고 있는 Main, 계층 구조를 가지면서 보다 기능이 확장된 SNR Scalable, Spatial Scalable, High 등이 그것이다.

구체적 응용 예로서는, 무궁화 위성을 이용한 디지털 DBS 방송이나 최근 도시바와 소니 진영이 규격 합의를 본 DVD 등은 Main Profile/Main Level, 미국과 우리나라의 HDTV는 Main Profile/High Level, 유럽의 HDTV는 Spatial Scalable Profile/High 1440 Level, 그리고 미국의 디지털 케이블 방송은 Simple Profile/Main Level이다.

MPEG-2 비디오는 현행TV나 HDTV를 효율적으로 압축하는 것이 주목적이다. 현행TV의 화질은 3~9Mbps에서, 그리고 HDTV 화질은 17~30Mbps에서 각각 얻어진다. 비트율은 주어진 채널의 용량과 요구 화질을 고려하여 선택된다. 예를 들어 무궁화 위성 DBS에 있어서는 약 7Mbps, 미국의 그랜드 얼 라이언스 HDTV방식에서는 17Mbps, 전화선을 이용한(ADSL-3 방식) 주문형 비디오(VOD)나 DVD에 있어서는 5~6Mbps를 비디오에 할당하고 있다. MPEG-2 비디오에서는 컴퓨터에서 채택되고 있는 순차주사만을 대상으로 하는 MPEG-1 비디오와는 달리, TV에서 사용되고 있는 비월주사 방식의 동화에 대해서도 많은 고려를 하고 있다. 즉 한 화면(프레임)을 두 필드(짝수 번째 주사선으로 이루어진 필드와 홀수 번째 주사선으로 이루어진 필드)로 나누어 필드구조로 부호화 할 수도 있고, 프레임 구조로 부호화 할 수도 있다. 움직임이 많은 장면은 한 프레임의 두 필드간에도 큰 차이가 나므로 필드구조로 부호화하는 것이 효과적이고, 정지화에 가까울수록 두 필드간에 상관도가 높아 프레임 구조로 부호화하는 것이 유리하다. 또 프레임 구조의 부호화에 있어서도 각 매크로블록(16×16화소 단위)별로 필드단위의 처리가 가능하도록 하여 화면내의 부분적 움직임을 용이하게 처리할 수 있게 하고 있다. 이와 함께 많은 요소가 결합되어 MPEG-2 비디오는 MPEG-1 비디오보다도 월등한 동화 압축 능력을 가지게 된다.

2. MPEG-2 비디오의 구조

많은 표준화 작업을 통해 MPEG은 DCT 부호화와 예측부호화를 이용하여 상당한 압축된 정보를 얻을 수 있다 그 데이터 구조를 살펴보면 Fig 2.1과 같이 시퀀스 층, GOP층, 픽처 층, 슬라이스 층, 매크로블록 층, DCT 계수를 포함하는 블록 층으로 분류되어진다

1) 시퀀스 층

비디오의 수평크기와 수직크기, 비디오내의 매크로블록 수, 초당 매크로블록 수, 화면을, 순방향과 역방향 f코드, 버퍼크기: VBV(Video Buffering Verifier), 비트율 등이 지정된다.

2) GOP 층

몇 개의 프레임들을 일련의 집합으로 표현한 것으로 비디오 정보의 랜덤액세스에 이용된다. GOP내의 프레임 수(N)나 I 또는 P 프레임이 나타나는 주기(M)를 지정한다. 비디오 스트림상에서 GOP의 최초는 I 프레임이고, GOP의 마지막은 I 또는 P 프레임이 된다. 통상적으로 이용되는 프레임 수(N)는 15이고 주기(M)는 3이다. 즉 I와 P 프레임 사이에 두 개의 B 프레임이 존재한다.

3) 픽처 층

한 프레임에 대한 픽처 타입, 움직임 벡터에 대한 플래그, 확장 비트 등의 공통적인 속성을 지정하는 층이다.

4) 슬라이스 층

한 프레임을 임의의 길이(MB들의 집합)로 분할한 것을 슬라이스라 하며, 이에 대한 공통적인 정보를 지정하는 층이다. 슬라이스는 임의의 길이의 매크로블록(MB)의 띠이며 여러 픽처에 걸쳐 있을 수는 없다. 처음과 마지막 MB는 서로 연계할 수 없고, 한 개의 MB로 구성된 슬라이스의 경우 그 MB는 건너 뛴 수 없다.

5) 매크로블록 층

슬라이스 층을 더욱 분할하여 매크로블록에 대한 공통적인 정보를 지정한다. 부호 발생량이 적을 때 삽입하는 매크로블록 스타핑, 매크로블록 Escape, MBA(Macroblock Address), MB 타입 등이 지정된다.

6) 블록 층

매크로블록 층을 8×8블록으로 더욱 세분화한 층이며, 해당되는 DCT 계수를 포함하고 EOB(End Of Block)로 종료된다. 인트라(Intra) DC는 독자적인 VLC(Variable Length Coding, 가변장 부호화)를 사용한다.

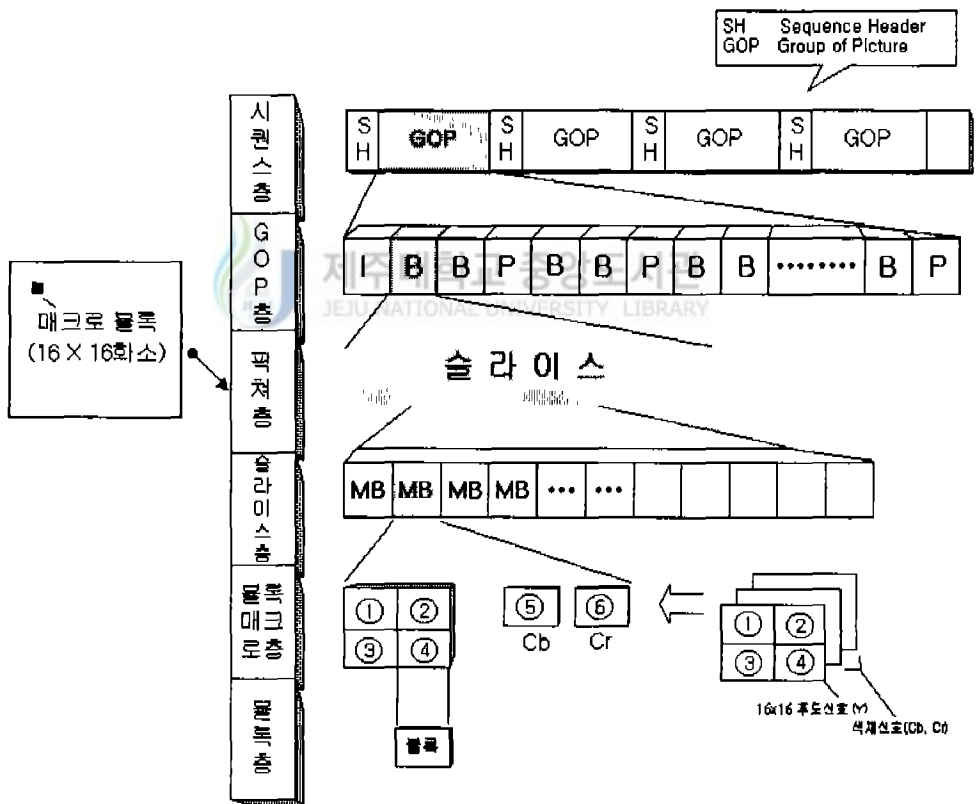


Fig. 2.1 The Structure of MPEG-2 Video

3. MPEG-2 Video를 위한 Algorithm

1) JPEG and MPEG 의 개요

본 논문의 알고리즘을 소개하기 전에 본 논문의 배경이 된 JPEG과 MPEG의 원리를 요약한다. Fig. 2.2는 정지 영상 압축 표준 JPEG의 기본 과정을 설명한다. 주요 개념은 공간적 여분을 줄이는 것이다. 압축은 이미지를 8×8 픽셀 블록들로 나누는 데서 시작된다^[2,6,4]. 각 블록의 세기 정도는 이산여현변환(DCT)에 의해 64 계수로 변환되고, 양자화와 run-length, 허프만 엔트로피로 부호화 된다.

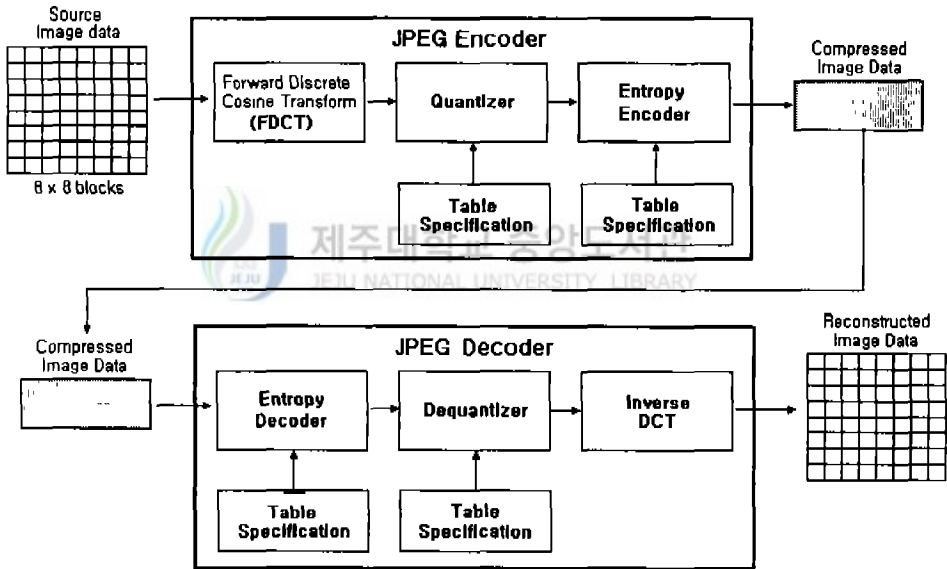


Fig. 2.2 Block diagram of sequential JPEG encoder and decoder

MPEG, 동영상 압축 표준은 Moving Picture Experts Group에 의해 제안되었다. MPEG은 JPEG과 같이 공간적 여분을 줄이는 것이지만 시간적 여분을 줄이기 위해 움직임 보상을 사용한다. 그렇지만 이 방법은 계수 코딩과 빠른 랜덤 접근사이의 교환을 수반한다.

랜덤 접근을 용이하게 하기 위해서 어떤 프레임은 움직임 보상 부호화 없이 still JPEG 영상처럼 부호화 된다. 이들 프레임을 intra-pictures(I)라 한다. 프레임의 두 가지 타입은 움직임 보상으로 부호화 된다: 이들 프레임을 순방향 코드(P) 프레임과 양방향 코드 (B) 프레임이라 한다. Fig. 2.3에 이 세 가지 타입을 설명하였다^[2, 4]. 움직임 벡터의 두 집합(매 8×8 블록중 하나)은 앞뒤이고, 각각의 B프레임과 각 P프레임의 단일 집합과 관련이 있다. P와 B프레임에서 각 블록의 움직임 보상과 관련이 있는 나머지 차이점은 JPEG 프레임에서 블록같이 같은 방법으로 DCT로 부호화 된다. 본 논문에서는 비디오 파싱을 위해서 DCT와 움직임 보상 데이터를 이용하는 방법을 연구한다.

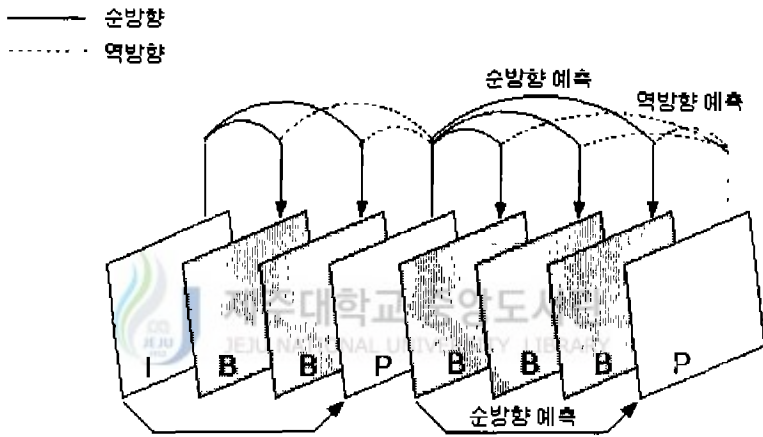


Fig. 2.3 Types of Frame in the MPEG-2 Video Standard

각 픽처의 순서는 우선 I, P 픽처를 먼저 처리하고 그 다음에 I, P의 사이에 있는 B픽처를 처리한다.

Table. 2.1 A Kind of screen picture

픽처(화면)의 종류	내 용
Intra-Picture 프레임 내 부호화 영상	<ul style="list-style-type: none"> · I 픽처는 GOP(Group of Picture)의 독립성을 가짐 · 화면을 모두 화면 내 부호화하는 화면 · 원 영상과 같은 순서로 부호화된다
Predictive-Picture 프레임간 순방향 예측 부호화 영상	<ul style="list-style-type: none"> · P 픽처는 화면 내의 소블록 단위의 부분에는 화면내 부호화를 포함하는 경우도 있다. · 원화면과 같은 순서로 부호화된다.
Bidirectionally Predictive-Picture 쌍방향 예측부호화 영상	<ul style="list-style-type: none"> · I, P 픽처를 먼저 처리한 후에 그 사이의 삽입된 B픽처를 부호화한다. · B 픽처로 화면 내의 소블록 단위의 부분에서는 화면내 부호화를 포함하는 경우도 있다.

2) DCT를 기반으로 한 알고리즘

공식화된 알고리즘은 Arman등에 의해 개발되었다. JPEG으로 압축된 비디오의 연속된 프레임의 DCT 계수사이의 관계에 기반을 두었다. 또한 보통의 영상인 경우에는 통계적으로 고주파보다는 저주파가 많기 때문에 화상을 주파수별로 분해하면 값이 저주파에 몰리게되고 고주파에는 거의 존재하지 않아서 작은 값을 갖는 고주파를 무시해도 화상에는 별로 영향을 미치지 않는다. 이렇게 압축이 수행되어 지는데, 이때 부호화 하는 것이 DCT이고, 복호화 하는 것이 역 DCT (Inverse DCT, IDCT)이다. N×N의 2차원 DCT의 수식을 보면 다음과 같다.

■ DCT 수식

$$F(u, v) = \frac{2}{N} C(u) C(v) \sum_{x=0}^{N-1} f(x, y) \cos \frac{(2x+1)u\pi}{2N} \cos \frac{(2y+1)v\pi}{2N} \quad (2-1)$$

- $u, v, x, y = 0, 1, 2, \dots, N-1$
- $x, y =$ 샘플 영역에서의 공간좌표

• $u, v =$ 변환 영역에서의 공간좌표

$$C(u), C(v) = \begin{pmatrix} \frac{1}{2} & u, v=0 \text{ 일때} \\ 1 & \text{그렇지않을때} \end{pmatrix} \quad (2-2)$$

■ IDCT 수식

$$F(u, v) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u, v) \cos \frac{(2x+1)u\pi}{2N} \cos \frac{(2y+1)v\pi}{2N} \quad (2-3)$$

또한, 각 프레임의 벡터 표시는 프레임 안에 블록의 부분집합에 의해 구성된다.

$$V_f = \{c_1, c_2, c_3, \dots, c_k\} \quad (2-4)$$

프레임간의 차이 값은 정규화 된 내적에 의해 다음과 같이 정의된다:

$$\psi = 1 - \frac{|V_f \cdot V_{f+\varphi}|}{|V_f| |V_{f+\varphi}|} \quad (2-5)$$

비교되는 두 프레임사이의 프레임의 수가 φ 이다.

화소단위비교법의 접근^[4]에 따르면 연속되는 비디오 프레임에 대응되는 블록의 DCT계수를 비교하는 화소단위 블록 알고리즘이 개발되었다. 보다 엄밀히 말해서, $c_{l,k}(i)$ 를 프레임 i 에서 블록 l 의 DCT계수라고 하면, k 의 범위는 1에서 64까지이고 l 은 프레임의 크기와 상관 있다. φ 프레임 떨어져 있는 두 프레임에서 블록 l 의 내용 차이는 다음과 같은 식으로 측정할 수 있다.

$$Diff_i = \frac{1}{64} \sum_{k=1}^{64} \frac{|c_{i,k}(i) - c_{i,k}(i+\varphi)|}{\max[c_{i,k}(i), c_{i,k}(i+\varphi)]} \cdot 100\% \quad (2-6)$$

만일 이 차이가 주어진 경계값 t 를 초과하면:

$$Diff_i > t \quad (2-7)$$

두 프레임사이 에 변화가 있다는 것을 알 수 있다. 만일 $D(i, i+\varphi)$ 이 변경된 블록의 퍼센트로 정의된다면 분할 경계는 다음과 같이 선언된다.

$$D(i, i+\varphi) > T_b \quad (2-8)$$

여기서 T_b 는 카메라 브레이크에 대한 경계 값이다. 이 차이 값은 픽셀의 세기 대신에 DCT계수를 사용한 화소단위비교법과 유사한 방법이다. t 에 대한 적당한 값을 선택하고 T_b 는 [7]에서 언급된 것과 같은 기술을 쓴다. t 는 다른 비디오 소스를 통해서 변화하는 경향이 없고 실지로 쉽게 결정될 수 있다. 반면에 T_b 는 약 5개 표준편차의 평균을 초과하는 값으로 $D(i, i+\varphi)$ 의 값에 대한 통계 전체에 관하여 최적으로 계산된다. 실행을 목적으로 한 처리 시간은 계수와 블록의 부분집합을 이용하는 Arman의 기술을 적용함으로써 상당히 줄일 수 있다.

3) 동영상 부호화(temporal redundancy reduction) 압축

움직임 추정(Motion estimation) : 이전 화면(S_{n-1})내에서 탐색 영역을 정해 놓고 현 화면(S_n)의 매크로 블록과 가장 유사한 매크로 블록을 찾는 것으로, 움직임 벡터(Motion vector) 값이 얻어진다. 움직임 벡터란 현재의 MB(macroblock)와 가장 유사한 MB까지의 거리를 말한다.

Motion vector = V_x, V_y

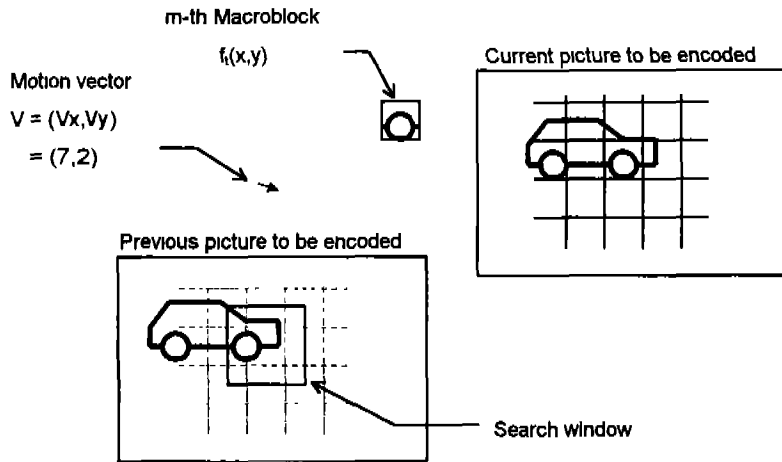


Fig. 2.4 Motion Estimation.

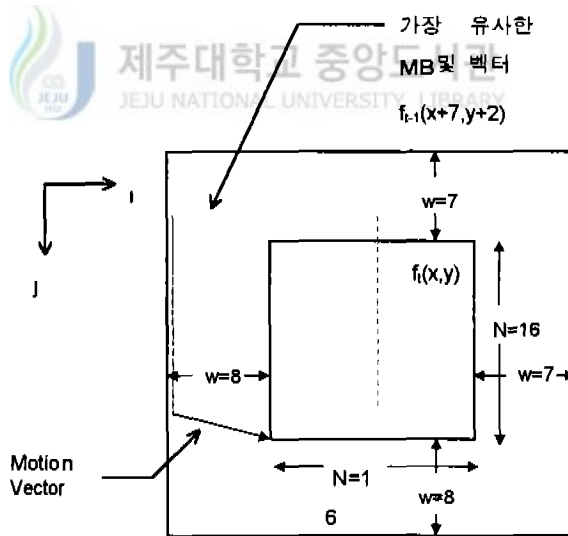
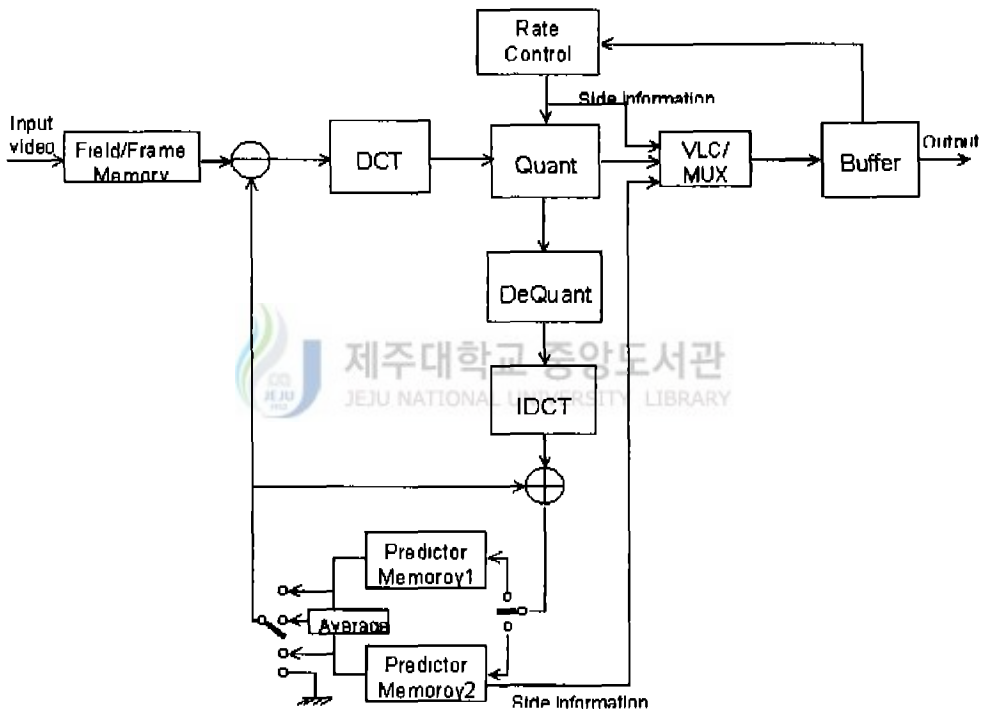


Fig. 2.5 The Search area in block-matching Techniques for motion Vector estimation.

4. MPEG-2 부호화기 특징

비디오 신호입력은 DCT, 양자화, 반화소 움직임보상, 예측부호화를 통해 압축되어지고 VLC를 통해 인코딩 된다. 이는 다시 역양자화, IDCT(Inverse DCT), 예측복호화를 거쳐 화면에 출력된다. 이를 도시한 블록도는 Fig. 2.6이다.

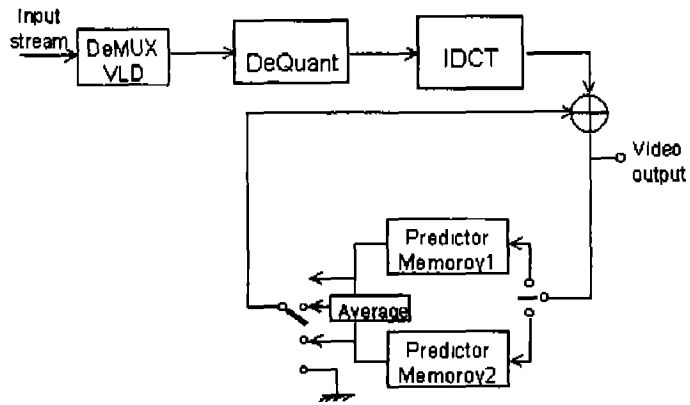
1) MPEG-2 ENCODER and DECODER



(a) MPEG Encoder Diagram

Fig. 2.6 Block diagram of MPEG Video Encoder and Decoder

<Image Continue>



(b) MPEG Decoder Diagram

Fig. 2.6 Block diagram of MPEG Video Encoder and Decoder

2) Bidirectional prediction

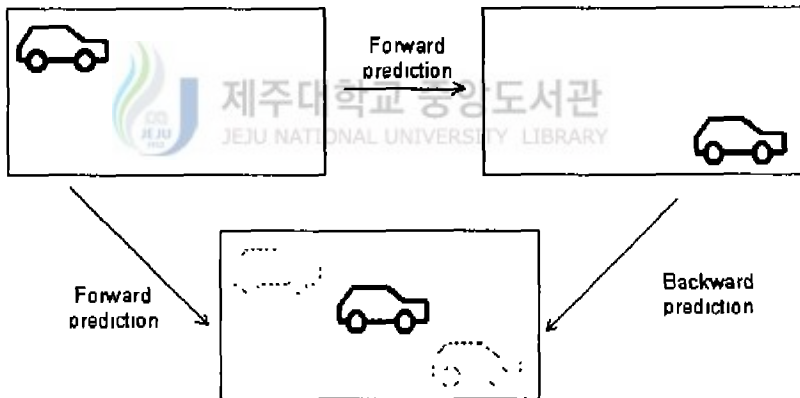


Fig. 2.7 Bidirectional prediction

3) 화면처리순서

At the encoder input:

1 2 3 4 5 6 7 8 9 10 11 12 13
I B B P B B P B B I B B P

At the encoder output:

1 4 2 3 7 5 6 10 8 9 13 11 12
I P B B P B B I B B P B B

At the decoder output:

1 2 3 4 5 6 7 8 9 10 11 12 13



III. MPEG-2 Video에서 저 비트율, 고화질 개선 방안

MPEG과 같은 압축방법으로 압축된 비디오에서 키 프레임의 정의를 하는데 어려움은 프레임들이 서로 다른 종류(I, P, B 프레임)라는 것과 다양한 패턴으로 나타난다는 것이다. I 프레임은 실제적인 픽셀정보의 DCT 계수를 포함하지만, 움직임벡터는 가지고 있지 않다. 반면에, P나 B 프레임은 다른 레퍼런스 프레임(들)으로부터 예측이나 보간 후에 발생하는 오차에러 데이터로부터 만들어진 DCT 계수를 가지고 있지만, 그 프레임들을 레퍼런스 프레임과 연결시키는 움직임벡터를 갖는다. 다양한 MPEG 클립들은 다양한 패턴의 I, P, B 프레임 순서를 갖는다. 동영상 부호화 시 I 프레임에는 그대로 적용할 수 있고, P, B 프레임 부호화 시에는 별도로 고려해야 할 점들이 생기게 되는데, 이 장에서는 MPEG-2 Video에서 저 비트율, 고화질의 해상도를 위하여 다음과 같은 방법을 제안한다.

1. MPEG-2 Video의 화질 개선 방안

이 절에서는 계산량을 크게 감소시킴으로서 실시간으로 화면변화에 효율적으로 대처하고, 화면과 화면사이 또는 하나의 화면 안에 존재하는 매크로 블록들 사이 화질의 균일성을 유지하면서도 평균 화질을 향상시킨 새로운 방법을 제시한다. 양자화 계수와 발생 비트량 사이의 관계를 나타내는 비트 테이블을 이용하여 화면간의 화질의 차를 최소화시키기 위해서 화면간 평균 양자화 계수의 크기를 고르게 하고, 화면 내부에서 상하좌우사이의 화질의 차를 감소시키기 위해서 화면을 몇 개의 작은 화면(Sub-picture)으로 나누어 매크로 블록들을 순차적인 방법으로 하는 것이 아니라 상하의 매크로 블록들을 혼합(Shuffle)하는 방법으로 부호화 하게 된다. 또한, GoS(Group of Slice)마다 부호화를 끝낸 후 인트라로 부호화 된 매크로블록의 비율을 계산하여 화면변화의 여부를 추정하여 목표 비트량과 비트 테이블을 조정하기 때문에 화면변화에 효과적으로 대처하여 화질의 저하를 극소화시켰다.

비트율 제어 방법은 먼저 출력 비트율(Transmission rate)과 초당 부호화 되어야 하는 화면의 개수 등 입력정보를 이용하여 GoP(Group of Picture)의 목표 비트량을 정하고, 정해진

GoP의 목표 비트량을 이용하여 GoP내 각각 화면의 종류와 순서에 따른 화질의 균일성을 유지하기 위해서 비트 테이블을 이용하여 각 화면들이 양자화 계수의 차가 최소화 되도록 화면의 목표 비트량을 할당한다. 화면의 목표 비트량이 할당되면 I-화면이나 B-화면의 경우는 화면을 다시 작은 화면으로 나누어 각각의 작은 화면에서 순차적으로 하나의 슬라이스를 뽑아 GoS을 만들고, 이 GoS을 단위로 화면 안의 화질의 균일성을 유지하도록 비트 테이블을 이용하여 양자화 계수를 정한다. 따라서, 하나의 GoS안에 모든 매크로블록은 같은 양자화 계수를 갖는다. P-화면의 경우는 I-화면이나 B-화면과 같은 방법으로 GoS의 양자화 계수를 계산하고, 각각의 GoS을 부호화 한 후 인트라 모드로 부호화 된 매크로블록의 비율을 계산하여 그 비율에 따라 슬라이스 인트라 문턱 값(δ_{GoS})과 비교하여 화면변화의 여부를 판단하고, 화면변화가 발생되었다 추정되면 목표 비트량을 I-화면의 목표 비트량을 할당하고, 기중치를 조정하게 된다. GoS을 부호화 한 후 비트 테이블은 부호화 결과 양자화 계수와 발생 비트량의 관계를 이용하여 선형 보간(Linear interpolation)의 방법으로 조정함으로써 비트 테이블이 영상의 특성을 충분히 반영하도록 한다. Fig. 3.1은 제안하는 방법의 전체 구조이다.

화면과 화면사이 화질의 균일성을 유지하도록 목표 비트량을 결정하고, 양자화 계수를 결정하는데 사용한다. 여기서 비트 테이블은 화면의 목표 비트량을 정하고, 매크로 블록의 양자화 계수를 정하는데 이용하게 되고, 화면의 종류에 따라서 같은 양자화 계수라도 발생 비트량이 다르기 때문에 비트 테이블도 달라지게 된다. 영상의 특성과 출력 비트율(Transmission rate)에 따라서 I-화면, P-화면, B-화면사이의 양자화 계수에 따른 발생 비트량은 다르게 나타난다 P-화면은 I-화면에 의해서 움직임 추정을 통하여 보상된 화면이기 때문에 I-화면에 비해서 발생 비트량이 작다.

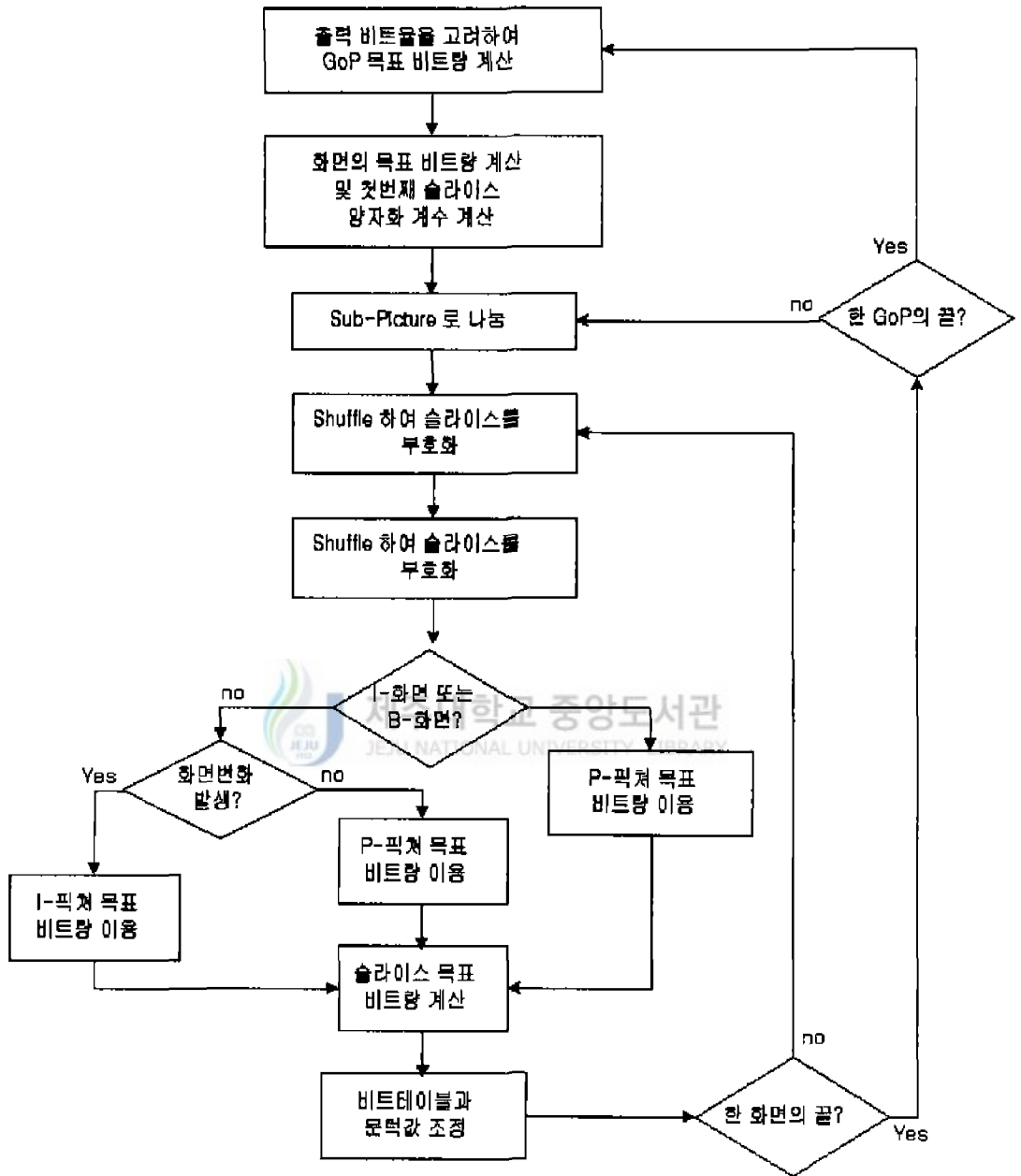


Fig. 3.1 The Block diagram of proposed for improving quality of image algorithm

B-화면은 양방향 움직임 추정을 통하여 보상된 화면이기 때문에 P-화면보다 발생 비트량이 작게 된다. 또한 양자화 계수가 작아지면 화면의 종류에 따른 발생 비트량의 차가 감소하게 되는데, 이것은 목표 비트량이 증가하게 되면 양자화 단계가 너무 작아져 이전 화면과의 차(Residue)를 부호화 하여 발생하는 비트량과 움직임 벡터를 나타내는 비트량의 합이 원 영상을 부호화 하여 발생하는 비트량과 비슷하기 때문이다. 그러나, 목표 비트량이 감소하여 양자화 단계값이 증가하면 양자의 발생 비트량이 차가 크게 나타나게 된다. 이와 같이 양자화 계수에 따른 실제 발생 비트량 사이의 관계를 Gaussian 확률변수의 비트율-왜곡의 모델을 적용하면 다음 수식 (3-1)과 같이 표현된다.

$$\begin{aligned}
 R_I &= \alpha_I + \beta_I \log \frac{1}{Q} \\
 R_P &= \alpha_P + \beta_P \log \frac{1}{Q} \\
 R_B &= \alpha_B + \beta_B \log \frac{1}{Q}
 \end{aligned}
 \tag{3-1}$$

여기서, R_I , R_P , R_B 는 발생 비트량을 의미하고, 상수 α_I , α_P , α_B 와 상수 β_I , β_P , β_B 는 영상의 특성과 출력 비트율에 따라서 그래프가 변하는 것을 나타내는 상수이다 Q 는 양자화 계수를 의미한다. 그러나 양자화 계수 - 실제 발생 비트량 사이의 관계는 비트율 - 왜곡의 관계에 비해서 훨씬 더 흔들림(Damping)현상이 심하다. 이것은 DCT계수가 엔트로피(Entropy) Coding이 되는 것이 아니라 Run-Length Coding이 되기 때문에 발생하는 현상으로 위의 수식 (3-1)과 약간 다른 다음 수식(3-2)으로 간단하게 표현될 수 있다

$$R_x = \alpha_x + \frac{\beta_x}{Q^\gamma} \quad , \quad (0 \leq \gamma \leq 2)
 \tag{3-2}$$

여기서 x 는 화면의 종류(I, P, B)를 의미한다.

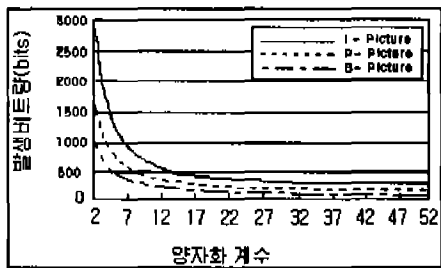


Fig. 3.2 The Generating bit rate that Quantization of Garden Image

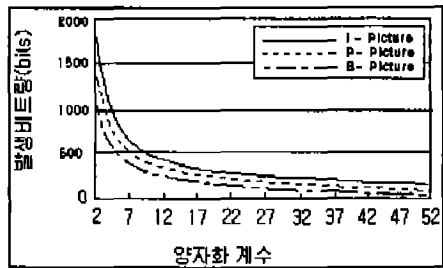


Fig. 3.2 The Generating bit rate that Quantization of Football Image

Foot영상과 모발영상의 양자화 계수와 실제 발생 비트량 사이의 관계는 영상의 특성에 따라 다르게 나타난다. 이것은 같은 양자화 단계값으로 양자화를 하더라도 영상의 특성에 따라 발생 비트량이 다르기 때문이다. 즉, 영상이 복잡하면 DCT계수의 AC성분이 크므로 양자화를 하더라도 발생하는 비트량이 증가하게 된다. 그러나, 비트 테이블은 GoS마다 계속해서 재조정되기 때문에 부호화를 진행함에 따라 부호화하고 있는 영상의 특성을 적응적으로 반영하여 영상에 적합한 비트 테이블이 된다. Table 3.1은 양자화 계수가 5의 배수일 때 실제 발생 비트량을 나타낸 초기 비트 테이블이다. 실제의 초기 비트 테이블은 52개의 양자화 계수에 대하여 표현하였다. 비트 테이블은 영상의 부호화 초기에 출력 비트율에 따라서 수식 (3-3)과 같이 조정된다.

$$B_Q^x = \frac{\gamma}{k} \times B_0 \quad x \quad (3-3)$$

Table. 3.1 Bit Table (The Generating bit rate per prame, Kbits)

종류 \ mq	2	5	10	15	20	25	30	35	40	45	50
I-화면	2800	1408	748	605	385	318	280	249	223	201	190
P-화면	1680	845	527	393	231	191	168	150	134	120	114
B-화면	1050	627	308	198	145	118	101	941	81	73	69

여기서 B_0^k 는 출력 비트율(γ)을 이용하여 조정된 화면의 종류($x \in I, P, B$)에 따른 비트 테이블의 발생 비트량이고, k 는 초기 비트 테이블을 만들 때 사용한 출력 비트율이다. B_0^k 는 k 의 출력 비트율로 부호화 하였을 때 발생한 비트량을 평균한 비트 테이블이 발생 비트량이다.

2. GoP 목표 비트량의 결정

GoP의 목표 비트량은 출력 비트율과 초당 부호화기에서 부호화 되는 화면수인 프레임율(frame rate)에 의해서 결정된다 즉, 목표 출력 비트율이 크게 되면 부호화의 결과 발생하는 비트량도 많아야 하기 때문에 GoP의 목표 비트량은 증가하고, 프레임율이 증가하면 일정한 출력 비트율이라면 프레임 당 발생하는 비트율이 감소하여 부호화의 결과 발생하는 비트량이 감소하게 된다. 즉, 화면 당 발생 비트량에 하나의 GoP안에 존재하는 화면의 개수를 곱한 결과 값에 이전 GoP에서 남은 비트량(목표 비트량과 실제 발생 비트량의 차)을 더하여 새로운 GoP의 목표 비트량의 수식은 다음과 같이 계산한다.



$$T_{GoP} = \frac{r_t}{r_f} \times N_{GoP} + R_{prev} \quad (3-4)$$

T_{GoP} 는 GoP의 목표 비트량을 의미하고, r_t 는 출력 비트율, r_f 는 초당 부호화 되는 화면 수를 의미한다. N_{GoP} 는 GoP안에 있는 화면 수, R_{prev} 는 직전 GoP의 목표 비트량과 실제 발생 비트량 사이의 차를 의미한다. 제안하는 구조에서는 화면 변화가 발생했을 경우 GoP의 크기가 변하는 가변 GoP 구조이기 때문에 GoP 목표 비트량은 GoP마다 큰 차를 가질 수 있다.

3. 슬라이스 목표 비트량과 양자화 계수 결정

가. GoS 목표 비트량 계산

화면의 목표 비트량이 정해지면 비트 테이블을 이용하여 GoS에 따라 양자화 계수를 계산한다. GoS의 양자화 계수를 계산하기 위해서는 먼저 GoS의 목표 비트량을 설정해야 한다.

화면을 n 개의 작은 화면으로 분할했다고 가정하면 GoS의 목표 비트량은 다음 수식(3-5)과 같이 구한다.

$$\begin{aligned} T_{GoS}^I &= \frac{T_I}{N_{slice}/N_{sp}} + \frac{R_{slice}}{N_{GoS}} \\ T_{GoS}^P &= \frac{T_P}{N_{slice}/N_{sp}} + \frac{R_{slice}}{N_{GoS}} \\ T_{GoS}^B &= \frac{T_B}{N_{slice}/N_{sp}} + \frac{R_{slice}}{N_{GoS}} \end{aligned} \quad (3-5)$$

위의 식(3-5)에서 T_{GoS}^I , T_{GoS}^P , T_{GoS}^B 는 화면의 종류에 따른 GoS의 목표 비트량이고, N_{slice} 는 하나의 화면 안에 있는 총 슬라이스의 개수이다. N_{sp} 는 하나의 화면 안에 존재하는 작은 화면의 개수이고, R_{slice} 는 직전 GoS에서 목표 비트량과 실제 발생 비트량의 차를 의미한다. N_{GoS} 는 현재 부호화 하는 화면에서 부호화 해야 할 남은 GoS의 수를 의미한다. 만일, 화면을 세($N_{sp}=3$)개의 작은 화면으로 분할하면 GoS의 목표 비트량은 3개 슬라이스의 목표 비트량을 더한 결과이다. 위의 수식(3-5)에서 직전 GoS까지의 목표 비트량과 실제 발생 비트량 사이의 차를 부호화 하는 화면의 남은 슬라이스 개수로 나누는 이유는 어떤 슬라이스의 특수한 성질에 따라 화질이 갑자기 변화하는 것을 억제하기 위한 것이다. 위의 식과 같이 구한 GoS의 목표 비트량을 이용하여 양자화 계수를 계산하게 되면 같은 GoS의 모든 매크로 블록은 같은 양자화 계수를 갖게 된다.

나. GoS 양자화 계수의 결정

매크로블록의 양자화 계수는 GoS의 목표 비트량을 참고로 GoS마다 결정되기 때문에 같은 GoS의 모든 매크로블록은 같은 양자화 계수를 갖게된다. GoS의 양자화 계수는 GoS의 목표 비트량을 가지고 GoS마다 조정되는 비트 테이블을 이용하여 결정한다. 즉, 부호화하고 있는 화면과 같은 종류의 비트 테이블에서 양자화 계수 축을 변화시켜 가며 비트량 값을 GoS의 목표 비트량과 비교한다. 그리고, 그 차가 최소가 될 때 양자화 계수 축의 변수를 GoS의 양자화 계수로 선택한다. 이렇게 함으로서 목표 비트량과 실제 발생 비트량의 차를 최소화 시키고, 화질의 균일성을 유지시킬 수 있다.

화면간의 양자화 계수의 변화율이 작으면 화질이 균일성을 유지할 수 있는 것과 마찬가지로 화면의 내부에서도 화면 상하 좌우사이 양자화 계수의 변화량이 작을수록 화질의 균일성을 유지할 수 있다. 그런데, 제안하는 방법은 화면의 목표 비트량을 결정하는 단계에서 양자화 계수의 변화를 최소화시키는 방향으로 목표 비트량을 할당했기 때문에 화면간 양자화 계수의 차는 매우 적다. 그리고, 화면 안에서는 GoS마다 비트 테이블의 재조정과 GoS의 목표 비트량이 R_{slice} 영향으로 다르기 때문에 GoS간에 양자화 계수의 차가 발생할 수 있으나 비트 테이블이 GoS마다 조정되어 영상의 특성을 충분히 반영하고, R_{slice} 값을 N_{GoS} 에 의해서 나누기 때문에 GoS간 양자화 계수의 차는 매우 작게된다. 그러므로, 화면내의 상하 좌우간 화질은 비슷하게 된다.

IV. MPEG-2 비디오 브라우저 설계

1. 실시간 MPEG-2 비디오 압축을 위한 알고리즘

MPEG-2 비디오를 부호화하고 복호화 과정은 시스템의 많은 부하를 초래하는 것을 피할 수 없다. 이에 본 연구에서 실시간 비디오 브라우징 시스템을 목적으로 MPEG-2 비디오를 이용하기 위해서 MPEG-2 비디오를 실시간적으로 부호화하고 복호화가 가능하도록 연구하였다.

MPEG-2 부호화 시스템에서 움직임 추정 과정은 가장 핵심적인 과정일 뿐만 아니라 연산이 복잡하여 지연이 가장 심한 과정이기도 하다. 그 원인으로는 속도를 고려하지 않은 전역 탐색 블록매칭 기법을 이용하였기 때문이다. 이러한 이유로 본 연구에서 MPEG-2 인코딩을 위한 실시간 움직임 추정 알고리즘을 개발하기 위해 블록매칭 기법과 그래디언트 기법을 혼합한다. 이를 간단하게 블록도로 표시하면 다음과 같다.

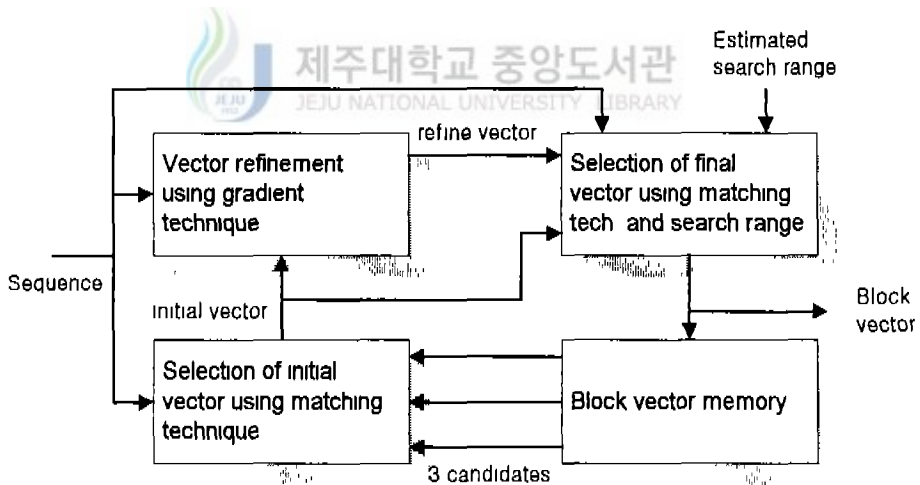


Fig. 4.1 The Block Diagram of Motion estimation Algorithm Using Hybrid matching Gradient method.

전역탐색 블록매칭에 기반한 움직임 추정에 대해, 벡터의 범위들은 주어진 탐색 범위에 의거하여 계산되므로 어떤 픽처라도 지연 없이 부호화 될 순 없다. 그러나, 그래디언트 기법을 이용한 움직임 추정에서는 탐색 영역이 불필요하게 된다. 이것은 모든 벡터들이 추정되어진 후에 벡터 범위가 계산되어진다는 문제점을 야기시킨다. 그러므로 움직임 추정에서의 그래디언트 기법이 한 픽처의 코딩 지연을 암시한다는 것을 이용하여 벡터의 범위들은 어떤 매크로블록이전에 전송되어야만 한다. 이러한 문제점을 피하기 위해, 본 연구에서 탐색 범위를 결정하는데 적용된 스키마를 개발하려 한다. 이 스키마는 움직임 벡터 필드의 후분석(post-analysis)에 기반한다. 그 개념은 연속되는 픽처들 사이에 움직임의 연속성을 이용하여 현재 픽처의 움직임 벡터 필드를 분석함으로써 코딩 순서에 있어서 다음 픽처에 대한 탐색영역을 결정하는 것이다. Fig. 4.2는 움직임 추정을 위한 탐색 범위를 결정하는데 적용된 스키마를 도시한다. 적용된 스키마는 다음과 같은 4가지 단계로 구성된다:

- ① 탐색 범위에 적용된 결정
- ② 결정된 탐색 범위의 정규화
- ③ 정규화된 탐색 범위의 스케일링
- ④ 스케일된 탐색 범위의 표준화

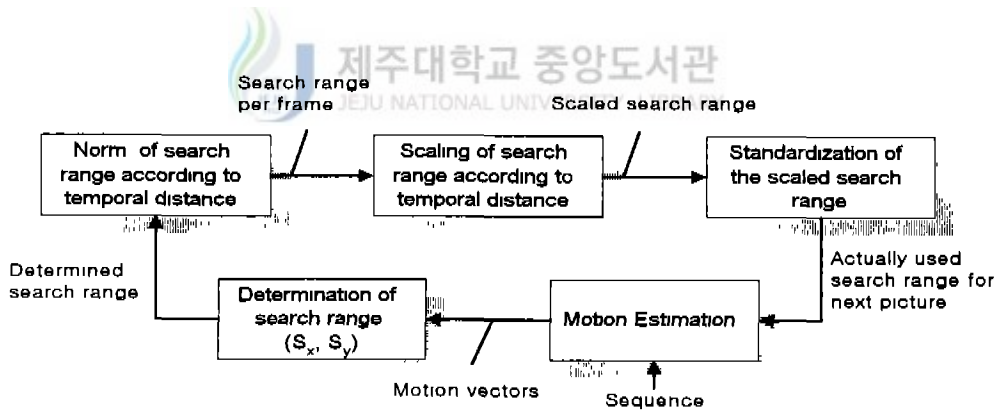


Fig. 4.2 Search Range decision method applied motion estimation

수평 탐색 범위 S_x 와 수직 탐색 영역 S_y 는 독립적으로 결정된다. 절대적 움직임 벡터들이 이 결정에 이용된다.

MB_{cnt} 를 픽처 내의 총 매크로블록 수로 놓고, $\{ (V_x^j, V_y^j) \mid j = 0, 1, 2, \dots, MB_{cnt}-1 \}$ 이 현재

픽처에서 적당치로 이용된 움직임 벡터라 할 수 있으므로, 탐색 범위는 다음과 같이 정의된다.

$$\begin{cases} S_x = E(|V_x|) + 2 \cdot D(|V_x|) \\ S_y = E(|V_y|) + 2 \cdot D(|V_y|) \end{cases}$$

$$\begin{cases} E(|V_x|) = \frac{1}{MB_{cnt}} \sum_{j=0}^{MB_{cnt}-1} |V_x^j| & D(|V_x|) = \frac{1}{MV_{cnt}} \sum_{j=0}^{MB_{cnt}-1} \|V_x^j - E(|V_x|)\| \\ E(|V_y|) = \frac{1}{MB_{cnt}} \sum_{j=0}^{MB_{cnt}-1} |V_y^j| & D(|V_y|) = \frac{1}{MV_{cnt}} \sum_{j=0}^{MB_{cnt}-1} \|V_y^j - E(|V_y|)\| \end{cases} \quad (4-1)$$

탐색 범위가 결정되어진 후에, 결정된 탐색 범위는 참조 프레임과 현재 프레임사이의 시간적인 거리를 이용하여 프레임 당 탐색 범위(search range per frame)로 정규화 된다. 그리고 나서, 다음 픽처와 그 참조 픽처(들)사이의 시간적인 거리에 따르는 정규화 된 탐색 범위를 스케일링함으로써 다음 픽처의 움직임 추정을 위한 탐색 범위가 얻어지게 된다.

마지막으로, 다음 픽처를 위한 스케일된 탐색 범위(순방향/역방향)가 표준화된다. S를 스케일된 탐색 범위(수평/수직/순방향/역방향)로 놓고, 탐색 범위는 Table. 4.1 에 보여진 것처럼 표준화된다.

Table. 4.1 Search Range

range of S	[1:8]	[9:16]	[17:32]	[33:64]	[65:128]	[129:256]	[256:512]	[512:1024]	[1025:2048]
std. S	8	16	32	64	128	256	512	1024	2048
f_code	1	2	3	4	5	6	7	8	9

추정된 탐색 범위들은 움직임 추정에서 탐색 범위 내로 벡터들을 조절하기 위해 이용된다. 이 방법은, 픽처를 움직임 추정하기 전에 벡터의 범위들이 추정된 탐색 범위에 따라 계산되어진다. 이 적용된 스키마를 이용함으로써 저지연(low-latency)이 보장될 뿐만 아니라 다양한 움직임을 갖는 시퀀스들의 높은 코딩 효율이 보장된다.

그 이유는 탐색 범위가 장면-적용(scene adaptive)된 것이기 때문이다.

이러한 기법들을 이용하면 MPEG-2 비디오 압축과정에서 거의 모든 부분을 차지하는 움직임 추정 과정이 단축되므로 실시간적으로 비디오를 부호화 할 수 있다.

2. 실시간 MPEG-2 Video 브라우저의 인터페이스

비디오 파싱 기술의 결과는 인덱싱과 브라우저를 하기 위해 이용될 수 있다. 경계 분할은 각각 기본적인 비디오 데이터의 인덱싱 단위에 대한 포인터로써 제공된다. 카메라 동작의 설명은 샷 내용 표시에서 대표 프레임 구성에 사용될 수 있고, 검색키로 사용할 수 있는 인덱싱의 또 다른 단계를 제공한다. 본 논문에서는 내용을 기반으로 한 비디오 검색과 빠른 브라우저에서 비디오 파싱으로 부터 얻을 수 있는 정보를 이용하는 방법을 검토한다. 비디오 정보 검색을 쉽게 하기 위한 중요한 기능은 질의에 대해 비디오 자료를 검색한 후에 빠르게 브라우저해 주는 도구이다. 이러한 브라우저는 시간적 요소의 두 단계(대충 보기와 자세히 보기)가 있으며, 또한 이러한 두 단계 사이의 동작을 용이하게 한다. 상세한 브라우저는 VCR과 같은 기능, 특히 빨리 감기와 되감기에 기반을 두었다. 이러한 기능은 계산 집약적이지만 내용 기반이 아닌 압축된 소스의 모든 프레임의 압축을 푸는 일이 필요하다. MPEG-2에서 I 프레임의 압축 풀기와 디스플레이 하기는 빠르게 브라우저를 하기 위한 가장 간단한 방법중 하나이다. 그러나, I 프레임은 시간상으로 균일하게 분포하므로, 내용을 기반으로 할 필요가 없다. 내용 기반에 더 가까운 방법은 파싱된 비디오 데이터의 각 샷에 대한 대표 프레임이나 키 프레임의 압축을 푸는 것이다

기존의 정지영상 즉, JPEG을 이용한 시스템이 대부분인 반면에 I Picture만을 이용하게 됨으로써 용량이 늘어나게 된다. 따라서 초당 최대 25 Frame(320×240 해상도) 저장 성능을 가지게 되며, 움직임 검출 기능을 개발하여 움직임이 발생할 시에만 저장이 될 수 있도록 함으로서 저장 용량을 최소한으로 줄일 수 있다. 또한 동영상 녹화/재생/관리 기능으로 카메라로부터 입력된 영상들은 디렉토리에 MPEG 파일로 저장되어 재생과 관리에 편리하며, 디지털 기술을 활용하여 모든 화상 처리와 화면 전환을 컴퓨터로 처리함으로써 여러 가지 다양한 모

드의 기능을 제공하고 뛰어난 화면 Capture기능으로 선명한 화질을 제공한다.

3. 시스템 설계

대부분의 화상회의 시스템에서 사용하는 방식은 카메라에서 들어온 영상을 통해 사용된다. 비디오 테이프 방식은 화질과 대용량인 면에서 본다면 많은 양의 데이터와 저 화질 등을 개선하기 위해 적은 양의 데이터 저장과 화질의 질을 높일 수 있는 측면을 고려하여 본 시스템을 제안하고 구현하였다.

1) 시스템 설계

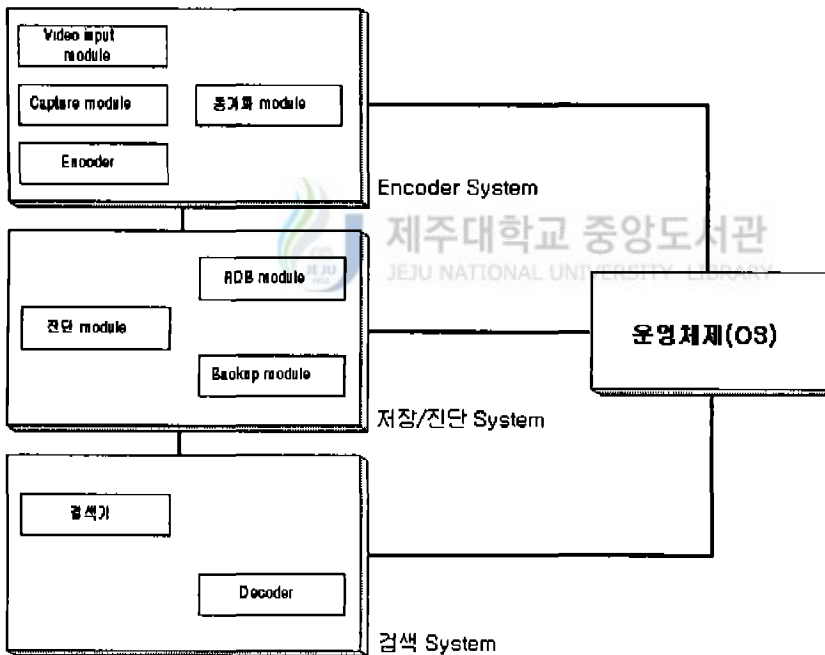


Fig. 4.3 System Implementation

(1) 부호화 시스템

비디오들의 입력을 부호화 시스템을 통해 MPEG DIB(Device Independent Bitmap)으로 변환하여 저장한다

- 가) Video Input Module : 카메라를 제어하고 영상을 화면에 출력한다.
- 나) Capture Module : 카메라의 아날로그 신호를 YUV 4:2:2 형식으로 변환한다.
- 다) 부호화 Module : YUV 4:2:2 형식의 신호를 MPEG DIB으로 변환한다.
- 라) 동기화 Module : 각각의 모듈간의 독립적 수행을 위한 동기화 기능을 수행한다.

(2) 저장/진단 시스템

부호화 시스템을 통해 생성되는 MPEG 파일을 저장/운용하고, 또한 Backup 기능을 수행하며, 세부사항은 다음과 같다.

- 가) RDB(Relational DataBase) Module : 카메라 ID와 그 움직임 그리고 생성시간에 따른 인덱스를 구축하고 각 테이블을 구성하여 관리
- 나) 진단 Module : 각 카메라의 상태를 항상 감시하며 문제 발생 시에 이를 통보하는 기능
- 다) Backup Module : DB에 저장된 MPEG파일을 저장 또는 외부 장치로의 이동

(3) 검색 시스템

RDB에 의해 생성된 각각의 인덱스에 의해 자료를 검색하고, 검색된 MPEG 정보를 복원하는 기능

- 가) 검색기 : DB에 저장된 데이터를 인덱스(카메라 ID, 움직임, 생성시간)를 통해 자료를 검색하는 기능
- 나) 디코더 : 인덱스에 의해 검색된 MPEG 정보를 복원하는 기능

V. 결과 및 고찰

1. 구현 환경(H/W, S/W)

전체시스템은 IV장 2, 3절에서 설계한 모든 기능들을 지원하고 Fig. 5.6과 같이 구현되었다. 동작하기 위한 환경은 다음과 같다.

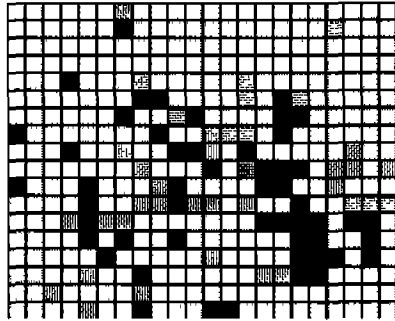
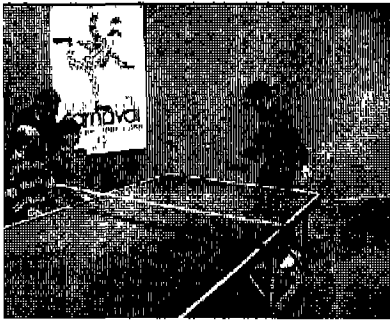
- ① 시스템 : 펜티엄II-333
- ② 개발 Tool : Visual C++ 6.0
- ③ 운영체제 : Windows 95/98, NT, RAM 64M
- ④ VGA Card : 1024*768, 16bit color
- ⑤ HDD . 9.1 GByte
- ⑥ 캡처 보드 : Invader 32
- ⑦ 스위치 카드, 카메라

2. MPEG에서의 컷 검출 고찰

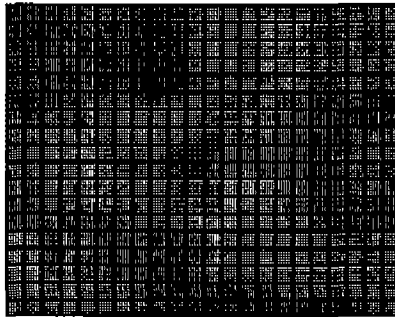
MPEG 스트림에서 컷은 대부분 다음과 같은 세 가지 경우에 발생한다

- ① 어떤 P 프레임이 인트라 부호화 된 MB들이 대다수일 때
- ② B 프레임에서 대다수의 MB들이 이전 I나 P 프레임들로부터 순방향으로 예측되었을 때
- ③ B 프레임에서 대다수의 MB들이 역방향으로 예측되었을 때

본 논문에서는 이런 MPEG 스트림에서 특성을 이용하여 컷을 검출한다. Fig. 5.1은 MPEG 스트림에서 컷이 발생할 때의 각 프레임의 MB들이 어떤 형태로 예측되었는지를 보여주고 있다.

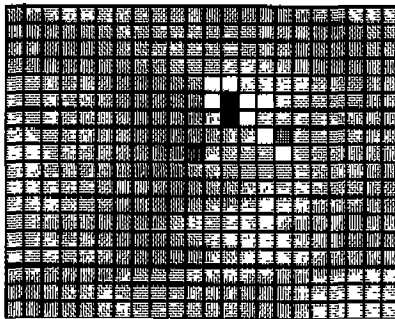
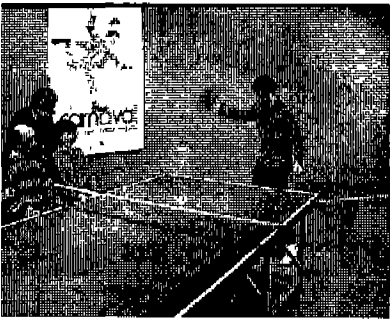


(a) B₂ Frame



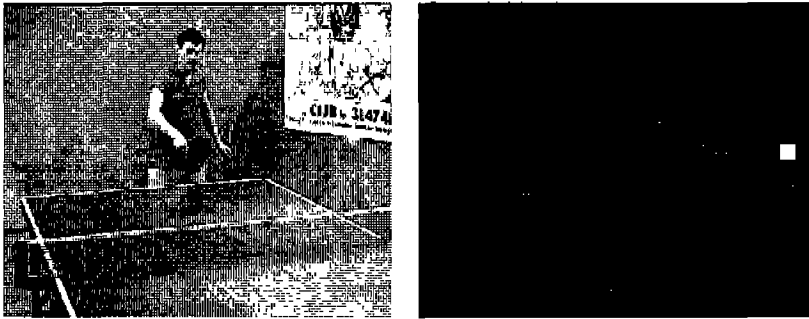
제주대학교 중앙도서관
JEJU NATIONAL UNIVERSITY LIBRARY

(b) I Frame



(c) B₁ Frame

<Image Continue>



(d) B₂ Frame

- Intra-coded MBs
- Backward-predicted MBs
- Forward-predicted MBs
- Bidirectionally MBs
- Skipped MBs

Fig. 5.1 Shot Change in MPEG Stream

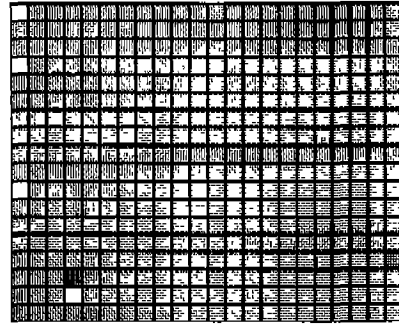
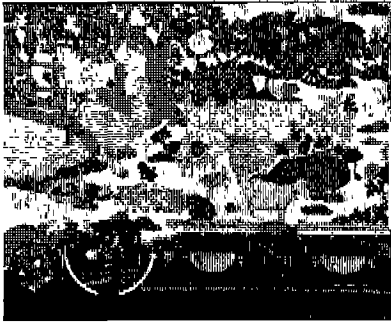
Fig. 5.1은 네 개의 프레임에서 컷이 발생한 예를 보여주고 있는데, 여기서는 조건 3과 조건 4에 의해 B₂ 프레임에서 컷이 검출되었다. 새로운 시퀀스는 다음 P 프레임과 거의 모든 부분이 비슷한 B₂ 프레임에서 시작한다. Table 5.1은 Fig. 5.1에서 나타난 각 프레임에 따른 조건 값들을 보여준다.

Table 5.1 Result of condition values

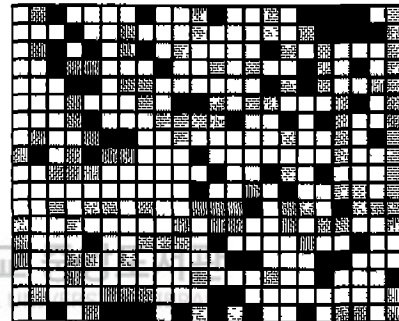
	SV _i /N	SV _i /N	N _{f(i-1)} -N _{b(i-1)}	N _{fb(i-1)}	N _{b(i)} -N _{f(i)}	N _{fb(i)}
B ₂ Frame	Passed	Passed	27	317	16	307
I Frame	Passed	1.77 (>t ₃)	Passed		Passed	
B ₁ Frame	Passed	Passed	Passed		-374	8
B ₂ Frame	Passed	Passed	374	8	395	1

Fig. 5.2은 두 B 프레임들 사이에서 샷 변화가 일어나지 않는 경우를 보여주고 있다. 그러나 MB들의 형태에 의존하면 샷 변화가 발생한 것으로 정의되어질 것이다. 이런 경우에 DCT

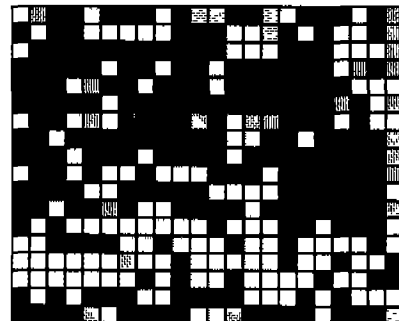
비교 과정이 요구된다.



(a) P Frame

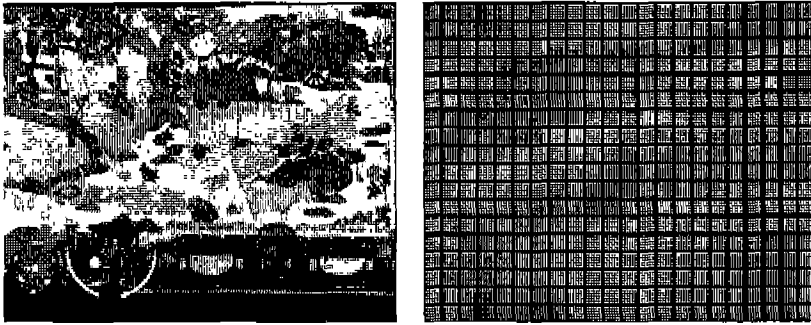


(b) B₁ Frame



(c) B₂ Frame

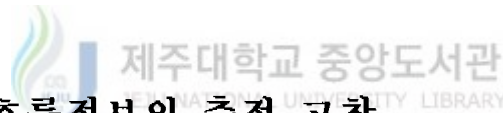
<Image Continue>



(d) I Frame

Fig. 5.2 Example without a shot change

비디오가 샷으로 분해된 후, 그 샷을 대표할만한 키 프레임의 선택이 선택되어야 한다. 이미 샷들은 컷 검출에서 비슷한 프레임들끼리 분할된 것이므로, 본 논문에서는 키 프레임 선택을 각 샷의 첫 번째 프레임으로 설정한다. 이는 영화 촬영가들이 트래킹이나 클로즈 업을 위한 주밍을 시작하기 전에, 처음 몇 개의 프레임들로 한 샷을 “특징화”하려고 하는 관점에 기반한 것이다.



3. DC 계수와 흐름정보의 추정 고찰

MPEG 시퀀스가 컷 검출에 의해 여러 개의 샷으로 분해되어지고 그 샷들을 대표하는 키 프레임을 선택한다 이 절에서는 각 키 프레임들의 특징인 DC 계수들과 흐름 정보를 추정한 결과를 보여준다. 만약, 해당 키 프레임이 I 프레임이라면 DC 계수들을 곧바로 얻을 수 있기 때문에 DC 추정은 생략되어지지만 P 프레임일 경우에는 추정되어야만 한다. 또한, P 프레임에서 추정된 DC 계수에 의해 복원된 영상과 압축을 해제한 후에 복원된 영상을 보여주고 있고 Fig. 5.4은 본 DC 계수의 추정법이 원래의 DC 계수들과 큰 차이를 보이지 않는다는 것을 보여주고 있다.

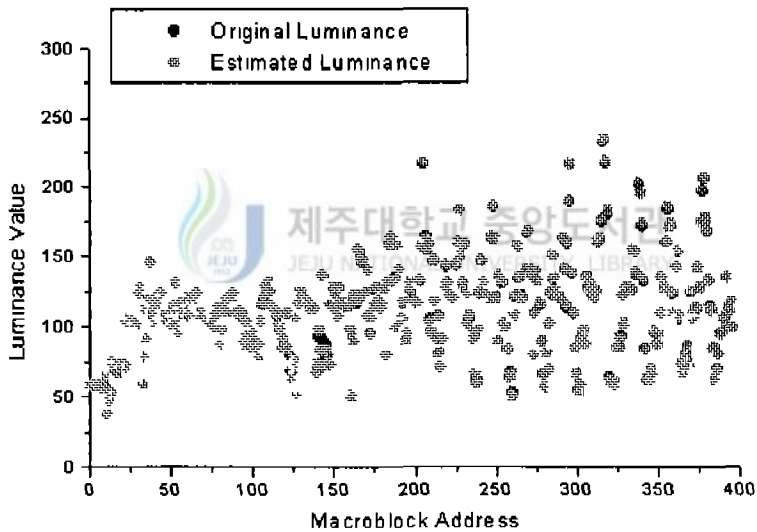


(a) Frame by original DC



(b) Frame by estimated DC

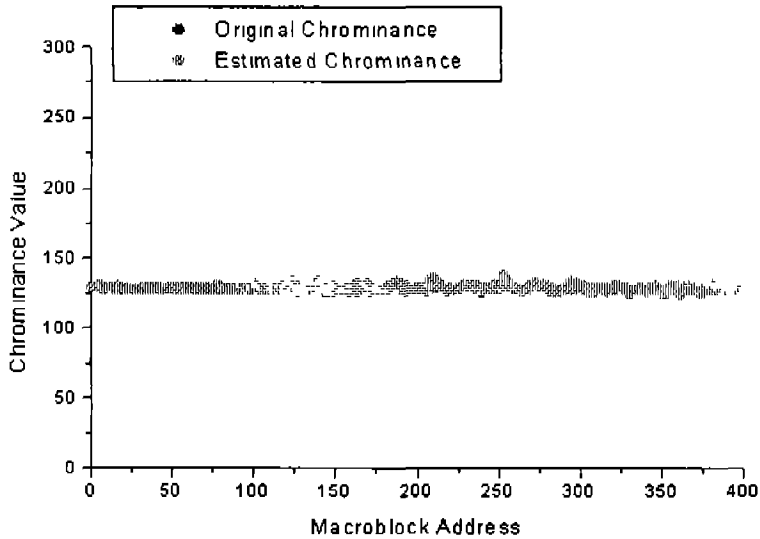
Fig. 5.3 Frame by estimated DC



a) Luminance of comparison

Fig. 5.4 Comparison of estimated DC and original DC

<Image Continue>



(b) Chrominance of comparison

Fig. 5.4 Comparison of estimated DC and original DC

4. 결과 내용



MPEG-2는 디지털 TV(또는 HDTV)와 같은 방송용을 목표로 제안되었다. 따라서 매우 고화질이며, 초저비트율이다. 그러나 처리량이 많아 고가의 전용 ASIC칩으로 구현되어 있으며, 그나마 아직 보편화되지 않았다. 그러나 본 개발에서는 PC 프로세서의 가격대 성능비가 급격히 상승하고 있는 점에 주목하여 실시간 S/W로 MPEG-2를 구현하였다. MPEG-2는 S/W 구현이 불가능한 것으로 알려져 있으나, 본 논문을 통하여 그 가능성을 입증하였다.

주요한 기술은 MPEG에서 가장 많은 연산을 요구하는 '움직임 추정과 보상'의 알고리즘을 개선하였고, DCT도 실시간 처리가 되도록 하였다. 아래의 화면은 실제 구현된 시스템의 브라우저이다.

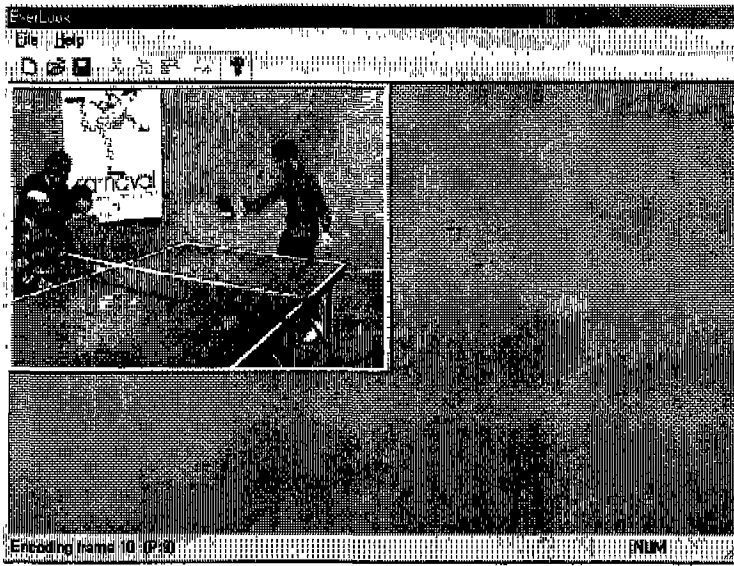


Fig. 5.5 Software MPEG Browser

NTSC 신호의 Camera로부터 입력된 실영상을 캡처하여 메모리에서 실시간 S/W로 압축한다. CIF 크기의 압축 비디오는 ATM망으로 실시간 전송된다. 현재는 초당 6프레임이 전송된다. 기존의 영상회의 시스템은 모두 전용 인코더/디코더가 필요했으나, 본 논문 결과 순전히 S/W에 의하여 구현된 시스템이 구축되었다.

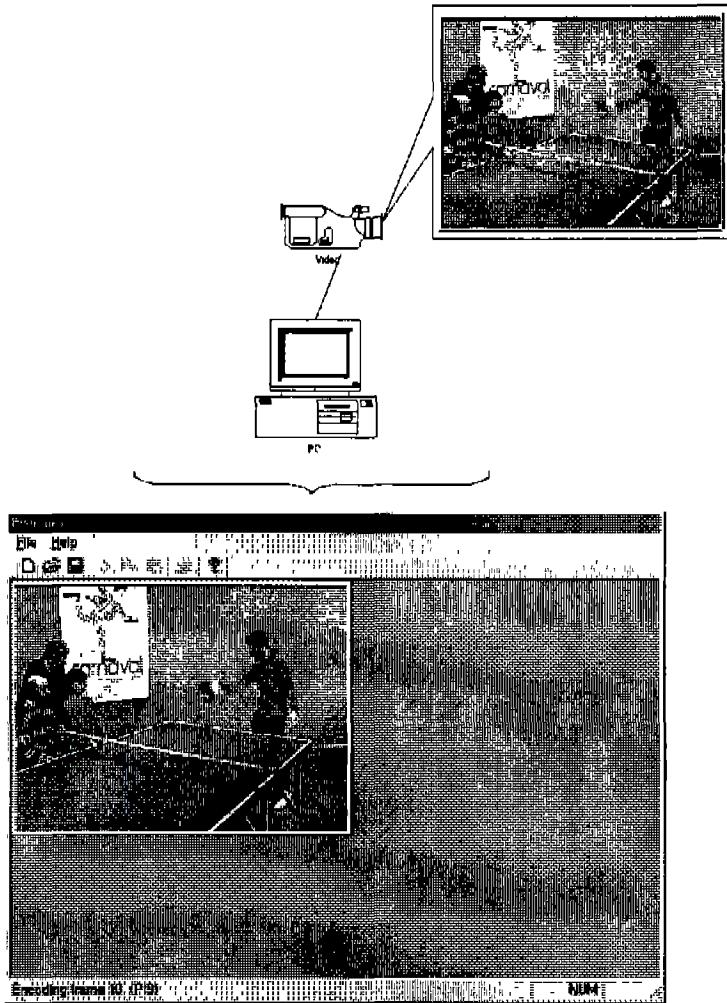


Fig. 5.6 Software MPEG Browser Interface

상호간의 전송되어온 MPEG-2의 데이터는 다음과 같은 Software를 통해 재생된다.

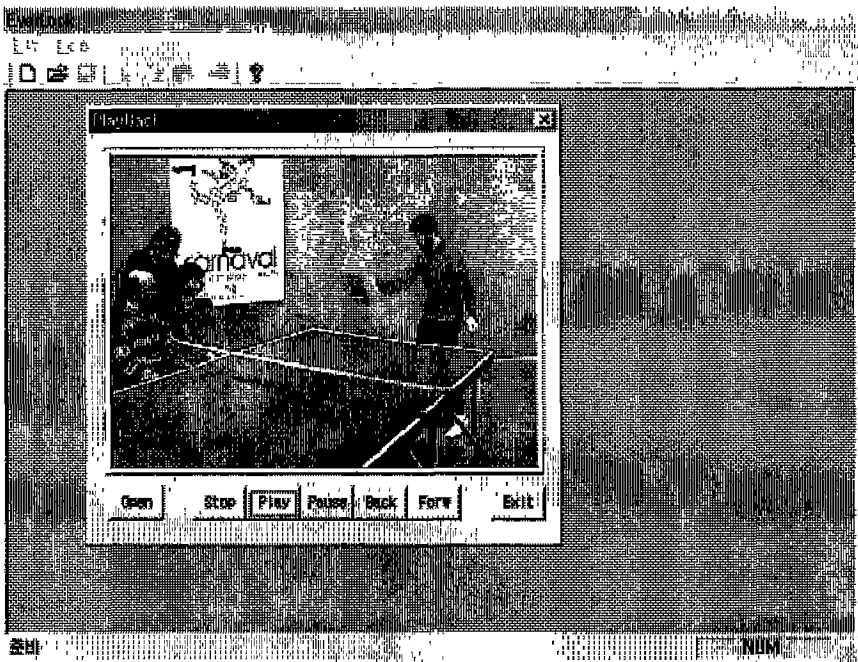


Fig. 5.7 Real-Time Process

VI. 결 론

본 논문에서는 압축된 MPEG-2 Video 스트림의 내용을 하드웨어적인 요소에서 벗어나 실시간 소프트웨어 MPEG-2 Video 브라우저 기법을 제안하였다. MPEG-2 비디오는 샷들로 분해되고 그 샷들을 대표하는 각각의 키 프레임들을 선택하여 공간적인 정보와 시간적인 정보로 그 프레임을 인덱스 한다. 또한 계산량을 크게 감소시킴으로서 실시간으로 화면 변화에 효율적으로 대처하고, 화면과 화면사이 또는 하나의 화면 안에 존재하는 매크로 블록들 사이 화질의 균일성을 유지하면서도 평균 화질을 향상시킨 새로운 방법을 제시하였다.

본 논문에서는 PC 프로세서의 가격 대 성능비가 급격히 상승하고 있는 점에 주목하여 실시간 S/W로 MPEG-2를 구현하였다. MPEG-2는 S/W 구현이 불가능한 것으로 알려져 있으나, 본 논문을 통하여 그 가능성을 확인하였다. 주요한 기술은 MPEG에서 가장 많은 연산을 요구하는 '움직임 추정과 보상'의 알고리즘을 개선하였고, DCT도 실시간 처리가 가능하도록 하였다. 차후 실제 구현된 브라우저를 통하여 실시간 처리 화상회의 시스템과 기타 감시 작동 시스템에서 용이하게 사용되어질 수 있다.

본 논문에서 비디오 부호화/복호화 과정을 기본으로 하였으며, 본 연구에서 전역탐색 블록 매칭 기법을 이용하지 않고 사전 탐색영역 결정 스키마와 움직임추정 알고리즘을 보다 구체화시키고 이를 직접적으로 구현하여 그 성능을 평가하였다. 결과적으로 MPEG-2를 이용한 어플리케이션에서 실시간적인 처리가 요구될 때 하드웨어를 사용하지 않고 소프트웨어만으로도 수준 높은 응용 소프트웨어가 구현될 수 있다.

참 고 문 헌

- F. Arman, A. Hsu, and M.Y. Chiu, "Image Processing on Compressed Data for B Large Video Databases", Proc. ACM Multimedia 93, Anaheim, CA, pp. 267-272, 1993.
- D. Le Gall, "MPEG: A Video Compression Standard for Multimedia Applications", Communications of the ACM, 34(4), pp. 46-58, 1991.
- R. M. Haralick and L. G. Shapiro, "Computer and Robot Vision", Vol. 2, Addison-Wisely, Reading, MA, 1993.
- M. Mills, J. Cohen, and Y. Y. Wong, "A Magnifier Tool for Video Data", Proc. CHI '92, Monterey, CA, pp. 93-98, 1992.
- A. Nagasaka and Y. Tanaka, "Automatic Video Indexing and Full-Video Search for Object Appearances", Visual Database Systems II, E. Knuth and L. M. Wegner, editors, North-Holland, pp. 119-133, 1991.
- B. C. O'Connor, "Selecting Key Frames of Moving Image Documents: A Digital Environment for Analysis and Navigation", Microcomputers for Information Management 8(2), pp. 119-133, 1991.
- R. Steinmetz, "Data Compression in Multimedia Computing-Standards Systems", Multimedia Systems 1(4), pp. 187-204, 1994.
- D. Swanberg, C. F. Shu, and R. Jain, "Knowledge Guided Parsing in Video Databases", Proc. IS&T/SPIE Conf. on Storage and Retrieval for Image Video Databases, San Jose, CA, 1993.
- L. Teodosio and W. Bender, "Salient Video Stills : Content and Context Preserved", Proc. ACM Multimedia 93, Anaheim, CA, pp. 39-46, 1993.
- Y. T. Tse and R. L. Baker, "Camera Zoom/Pan Estimation and Compensation for Video Compression", Proc. SPIE Conf. on Image Processing Algorithms and

Techniques II, Boston, MA, pp. 468-479, 1991.

- H. Ueda, T. Miyatake, and S. Yoshizawa, "IMPACT : An Interactive Natural-Motion-Picture Multimedia Authoring System", Proc. CHI'91, New Orleans, LA, pp. 343-350, 1991.
- G. K. Wallace, "The JPEG Still Picture Compression Standard", Communications of the ACM, 34(4), pp. 30-44, 1991.
- H. J. Zhang, A. Kankanhalli, and S. W. Smoliar, "Automatic Partitioning of Full-motion Video", Multimedia Systems, 1(1), pp. 10-28, 1993.
- H. J. Zhang, C. Y. Low, Y. Gong, and S. W. Smoliar, "Video Parsing Using Compressed Data", Proc. IS&T/SPIE Conf. on Image and Video Processing II, San Jose, CA, pp. 142-149, 1994.
- H. J. Zhang, S. W. Smoliar, "Developing Power Tools for Video Indexing and Retrieval", Proc. IS&T/SPIE Conf. on Storage and Retrieval for Image and Video Databases II, San Jose, CA, pp. 140-149, 1994.
- W. Verbiest, L. Pinno, and B. Voeten, "The Impact of the ATM Concept on Video Coding," IEEE J. SAC., 6-9, pp. 1623-1632, 1988
- M. Wada "Slective Recovery of Video Packet Loss using Error Concealment," IEEE J. SAC., Vol. 7-5. pp. 807-813, 1989
- M. Ghanbari and C. J. Hughes, "Packing Coded Video Signals into ATM Cells," IEEE/ACM Trans. Networking, 1-5, pp. 505-509, 1993
- S. Iai and N. Kitawaki, "Video Quality Degradation by Cell Loss in ATM Networks," NTT Review, 4-4, pp. 52-57, 1992
- P. Pancha, M. E. Zarki, "MPEG Coding for variable bit rate video transmission," IEEE. Comm. Mag, pp. 54-66, 1994

奇昌舒 “소프트웨어 MPEG-2 복호기 구현에 관한 연구” 강원대학교 석사학위논문, 1998

김성윤 “범용 DSP를 이용한 MPEG-2 부호화기의 실시간 구현”

연세대학교 석사학위논문, 1996

대우전자 영상연구소 지음 “MPEG 비디오” 연암출판사, 1995

柳明分 “디지털 컬러 영상의 검색과 MPEG-2 압축된 비디오의 브라우징”

동국대학교 석사학위논문, 1997

閔載泓 “MPEG-2의 scalability와 disparity 정보를 이용한 입체 영상 압축에 관한 연구”

한국항공대학교 석사학위논문, 1998

멀티미디어통신연구회 편 ; 정제창 역 “(그림으로보는)최신 MPEG” 교보문고, 1995

변진수 “MPEG-2 동영상을 위한 DCT 영역에서의 Transcoding”

한국과학기술원 석사학위논문, 1998

Gyungnam Boo “KAIST image computing system(KICS)에서 MPEG-2를 위한

이산여현변환의 실시간 구현에 관한 연구” 한국과학기술원 석사학위논문, 1997

Joong-Seob Song “실시간 MPEG-2를 위한 움직임 추정 및 보정의 구현”

한국과학기술원 석사학위논문, 1998

유시룡[등]저 “MPEG 시스템” 대영사, 1997

李相錫 "MPEG-2 시스템에서 CLT를 이용한 움직임 벡터 추정"

경북대학교 석사학위논문, 1995

李承濤 "TMS320C80을 이용한 실시간 MPEG-2 video encoding system을 위한

소프트웨어 개발에 관한 연구" 홍익대학교 석사학위논문, 1998

일본멀티미디어통신연구회 편 ; 정제창 역 "(그림으로보는)응용 MPEG" 교보문고, 1997

Seokwoo Chung "KICS를 이용한 실시간 MPEG-2의 움직임 복원에 관한 연구 및 구현

한국과학기술원 석사학위논문, 1997

최용일 "MPEG-2의 실시간 영상 압축을 위한 분산 연산 방식 DCT구조의 설계에 관한

연구" 전남대학교 석사학위논문, 1996

한성철 "FPGA를 이용한 MPEG-2 오디오 복호화기의 구현"

연세대학교 석사학위논문, 1996

許鎭龍 "MPEG-2 압축 동영상 정보관리 시스템에 대한 연구"

경상대학교 석사학위논문, 1998

감사의 글

시작 할 때는 막막하고 두렵기도 하였지만 2년이란 세월은 너무도 빨리 지나 지금은 아쉬움만 남고있습니다. 힘들었을 땀 그토록 빨리 지나가 버리길 바랬었던 시간들이었지만 이제 그 끝에 서서 돌아보니 언제나 그렇듯 참으로 많은 후회가 됩니다.

그 동안 해왔던 공부를 마무리하면서 이제 인쇄소에 제본을 맡기려하니 “좀 더 학문에 정진할 걸”하고 후회도 되고 아직도 부족한 부분들이 이 구석 저 구석에서 눈에 보여 부끄럽고 아쉬운 마음에 자꾸만 망설여집니다. 항상 바쁘다고 소홀히 하는 저를 채근하시면서 그나마 이렇게 부족한 제가 논문을 쓸 수 있게 도와주신 주위의 많은 분들께 진심으로 감사드립니다.

논문의 시작부터 완성될 때까지 항상 자상하게 지도해주신 김장형 교수님께 진심으로 감사드리고, 부족하지만 끝까지 논문을 읽어 주시고 조언을 아끼지 않으시며 심사해주신 안기중 교수님과 이상준 교수님, 논문이 완성되기까지 모든 고생을 같이 해준 강진석 선생께 진심으로 감사를 드립니다.

바쁜 중에도 직장일과 학업을 병행할 수 있도록 배려해주시고 성원을 보내주신 관장님과 과장님, 그리고 도서관 여러 선생님들께도 이 지면을 빌어서 진심으로 감사료를 드립니다.

일생을 자식을 위하여 헌신하신 어머니, 병중에 제신 몸으로도 제가 학위를 준비하는 동안 항상 관심과 애정을 쏟으시면서 오직 자식만을 걱정하신 어머니께 어찌 이 작은 지면에 글 몇 줄로 고마움을 전부 표현할 수 있겠습니까?

어머님 고맙습니다.

내 곁에 존재하는 것만으로도 나에게 힘은 되는 매형, 누나, 경용, 경희, 정훈, 기훈, 제수씨, 그리고 아버님의 빈자리쯤 채워 주시면서 힘이 되어 주신 장인, 장모님께도 감사료를 드립니다.

끝으로 직장일과 학업을 동시에 하느라 힘들어 할 때마다 용기와 격려를 아끼지 않으셨던 성원을 보내준 사랑하는 나의 아내 희숙과 늘 바쁘다고 가족들과 시간을 함께 하지 못해도 불평 없이 참아준 사랑하는 태한, 주변에게도 고마움을 전하며 편찮으신 어머니의 쾌유를 빌고 일찍 떠나신 아버님 영전에 이 논문을 바치고 싶습니다.

1999년 12월 강권익