

3차원 입체 변환을 위한 MPEG 압축 데이터에서의 영상 처리 기법

김만배

강원대학교 정보통신공학과

Video Processing of MPEG Compressed Data For 3D Stereoscopic Conversion

Man-Bae Kim

Department of Information and Telecommunications
Kangwon National University, Chuncheon, Korea

e-mail) manbae@cc.kangwon.ac.kr

ABSTRACT

The conversion of monoscopic video to 3D stereoscopic video has been studied by some pioneering researchers. In spite of the commercial potential of the technology, two problems have bothered the progress of this research area: vertical motion parallax and high computational complexity. The former causes the low 3D perception, while the hardware complexity is required by the latter. The previous research has dealt with NTSC video, thus requiring complex processing steps, one of which is motion estimation. This paper proposes 3D stereoscopic conversion method of MPEG encoded data. Our proposed method has the advantage that motion estimation can be avoided by processing MPEG compressed data for the extraction of motion data as well as that camera and object motion in random directions can be handled. 1)

1. 서론

2D/3D 동영상 변환으로 불리는 2차원 동영상의 3차원 입체 동영상 변환은 연속적으로 입력되는 TV등의 2차원 동영상을 입체 영상으로 변환하는 기술이다. 입체 영상은 3차원 시청 기기 또는 3D TV를 이용하면 2차원 영상에서 느낄 수 없는 3차원 입체감을 얻을 수 있다.

2D/3D 동영상 변환 기술의 관련 연구는 꾸준히 연구되어 왔다 [1,2,3,4,5]. Garcia는 인간 시각 특성이 가지고 있는 spatial-temporal interpolation 기능을 변환 방법에 적용하였다[1]. Okino 그룹은 세계최초로 MTF(Modified Time Difference) 방법을 이용하여 상업용 2D/3D 동영상 변환 TV를 제작하였다[2]. Matsumoto는 영상의 깊이 정보를 이용하여 단안 영상에서 입체 영상을 생성하였다[3]. 움직이는 물체가 존재할 때 정확한 깊이 추출이 어려운 단점이 있다. Kim은 수직시차의 수평시차 변환을 이용하여 2D/3D 변환 방법을 제안하였다[4,5].

본 연구는 정보통신부의 정보통신 우수 시범학교 지원 사업에 의하여 수행된 것임.

기존 논문에서는 NTSC 신호의 2D/3D 동영상 변환을 다루었으나[1-5], 본 논문에서는 MPEG(Moving Picture Expert Group) 압축 데이터를 이용, 직접 3차원 입체 영상으로 변환하는 새로운 방법을 제안한다. NTSC 신호 처리의 단점은 2D/3D 동영상 변환의 핵심 모듈인 운동 예측의 수행 시간 소요 및 이에 따른 하드웨어의 복잡성이다. 따라서 MPEG의 비트 스트림에는 블록의 운동 벡터가 있으므로 이것을 효율적으로 이용하면 기존 방법보다 간단한 구조로 입체 영상을 생성할 수 있다. MPEG 엔코더(encoder)에서 예측된 운동 벡터 값을 데코더(decoder)에서 사용하여 알고리즘 연산량을 줄일 수 있다. 압축 영역에서의 데이터 처리는 시간이 흐름에 따라 점점 더 많은 동영상들이 MPEG과 같은 부호화 과정을 거쳐 압축된 형태로 저장되거나 전송이 이루어 질 전망이다. 일반적으로 이런 압축된 동영상 처리는 일련의 복원 과정에서 시작된다. 그러나 완전 복원된 동영상을 그대로 처리할 경우, 그 방대한 데이터 량 때문에 완전 복원하지 않고 일부만을 복호화한 후 압축된 데이터에 영상처리 기술을 이용하는 방법이 선호되고 있다. 특히 2D/3D 동영상 변환에서는 운동 추출 및 해석이 매우 중요한 사항

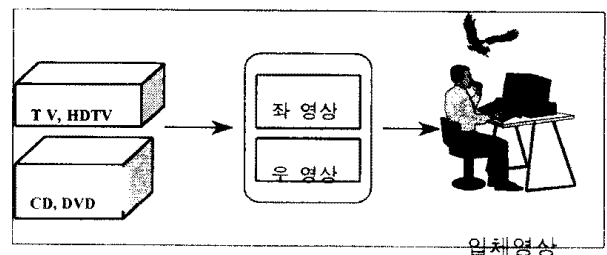


그림 1 2D/3D 동영상 변환의 원리

이다. 카메라 및 물체의 수직 운동 시 MPEG 압축 데이터에서 추출한 운동 값을 이용하여 운동 시차를 양안 시차로 변환하고자 한다. 이 기술은 MPEG-1 CD 및 MPEG-2 HDTV, DVD를 적절한 3차원 시청 기기 또는 3차원 TV를 이용하여 3차원 입체 영상을 시청하는 데 이용될 수 있다.

본 논문은 다음과 같이 구성되어 있다. 2절에서는 입체

영상의 원리 및 주요 문제점을 설명한다. 3절에서는 MPEG 압축 데이터 처리를 설명하고, 얻어진 결과를 이용하여 운동 벡터의 수평 시차 변환 방법은 4절에서 소개한다. 마지막으로 5절에서는 실험 결과를 분석한다.

2. 입체 영상의 원리

그림 1은 2D/3D 동영상 변환의 기본 구조를 보여준다. TV, HDTV, CD, DVD에서 얻어지는 영상은 2차원 영상이다. 각 한 장의 영상에서 좌영상과 우영상으로 구성되어 있는 입체 영상을 생성하는 것이 2D/3D 동영상 변환의 기본 개념이다.

그림 2와 3은 카메라 및 물체의 운동이 수직성분을 가지고 있을 때와 빠른 운동으로 인한 시차의 불일치 현상을 보여준다. 그림 2는 양눈이 각각 우영상 및 좌영상을 보고 있을 때에 영상 퓨전(image fusion) 현상을 보여준다. 왼쪽 그림은 두 매칭점(■로 표시됨)이 수직시차를 가지고 있지 않는 경우로 수평으로 매칭점이 놓여 있다. 두 직선간의 수렴점이 스크린 뒤에 발생하여 정확히 깊이감을 줄 수 있다. 오른쪽 그림에서는 두 매칭점이 수직시차

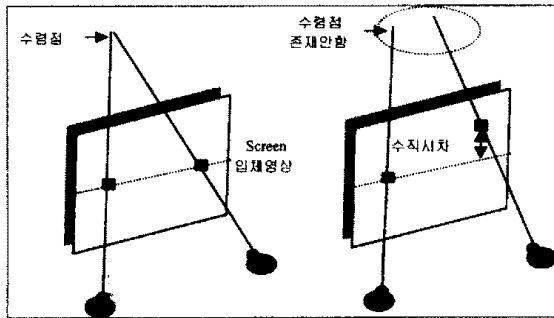


그림 2: 왼쪽 그림은 두 매칭점이 수직시차를 가지고 있지 않는 경우로 두 직선간의 수렴점이 발생하여 정확히 깊이감을 줄 수 있다. 오른쪽 그림에서는 두 매칭점의 수직시차때문에 수렴점이 존재하지 않으므로 입체감의 부재로 영상 퓨전(image fusion)이 일어나지 않는다.

(↔로 표시됨)를 가지는 경우를 보여준다. 두 매칭점에 수직시차가 존재하므로 수렴점이 존재하지 않는다. 따라서 영상 퓨전이 일어나지 않는다. 이 현상은 입체로 시청할 때에 깊이감의 상실로 눈의 피로감의 원인이 된다. 그림 3은 빠른 카메라 또는 물체 운동으로 매칭점의 시차가 양안 거리보다 크면 수렴점이 없다. 따라서 입체감의 상실로 인한 눈의 피로의 원인이 된다. 두 그림에서 관찰했듯이 2D/3D 동영상 변환에서는 위에 언급한 수직 시차 및 빠른 운동 처리가 중요한 문제이다. 다음절에서는 이 두 문제를 해결할 수 있는 방법을 MPEG 압축 데이터 처리와 함께 제안한다.

3. 제안 방법

본 논문의 제안 방법은 다음 세 가지 주요 과정으로 구성되어 있다.

1) MPEG 데코더로 입력되는 비트 스트림에서 I, B, P 픽처들의 매크로블록(macroblock) 운동 벡터를 추출한다. 인트라 코딩되는 I 픽처는 운동 벡터가 없으므로 주변의 다른 픽처들의 운동 벡터를 이용하여 I 픽처의 운동 벡터를 구한다. 반대로, P와 B 픽처는 엔코더에서 주변 픽처(들)를 이용하여 운동 예측되므로 운동 벡터를 비트스트림에서 추출할 수 있다.

2) 추출한 각 매크로블록의 운동 벡터는 수평성분과 수

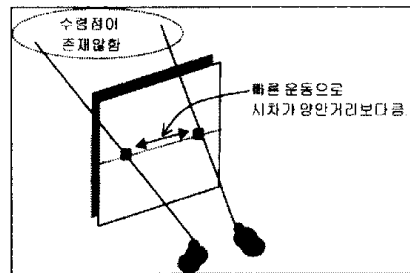


그림 3: 빠른 카메라 운동 또는 물체 운동으로 매칭점의 시차가 양안 거리보다 크면 수렴점이 없다.

직 성분으로 구성되어 있는데, 수직 성분은 입체 영상 시청시 깊이감 상실로 인한 눈의 피로를 가져온다. 수직 성분을 없애기 위해 운동벡터의 놈(norm) 값을 운동 시차(motion parallax) 값으로 사용한다.

3) 픽처마다 각 매크로블록을 2)에서 구해진 놈 값만큼 오른쪽으로 수평 이동하여 합성영상을 만든다. 이 과정에서 운동 시차는 양안 시차(binocular parallax)로 변환한다. 이 양안 시차는 깊이감을 결정하는 중요한 요소이다. 양안 시차와 깊이의 관계식은 [5]를 참조한다. 이 과정이 완료되면 각 픽처마다 대응하는 합성 영상이 만들어진다. 복원 영상은 기존의 MPEG 데코더를 이용하여 얻어진다. 입체 영상은 복원 영상과 합성 영상으로 구성된다.

그림 4는 MPEG 엔코더에서의 I, P, B 픽처의 운동 예측의 개요도를 보여준다. I 픽처는 P 및 B 픽처의 운동 예측에 이용된다. I 픽처는 운동 예측 없이 인트라 코딩되므로 자신의 운동 벡터는 없다. P 픽처는 이전 I 또는 P 픽처로부터 운동 보상 예측(motion compensated prediction)을 이용하여 코딩된다. 이 예측 방법은 순방향 예측(forward prediction)이라 불린다. B 픽처는 양방향(bi-directional)으로 운동 예측이 수행되는데 순방향 운동 벡터 및 역방향(backward) 운동 벡터가 얻어진다. 이전 또는 이후의 가장 가까운 I 및 P 픽처를 이용하여 운동 보상 예측된다. 엔코더에 입력되는 픽처와 데코더에 입력되는 픽처는 위에서 언급한 역방향 및 순방향 예측 단계 때문에 다른 순서로 들어온다.

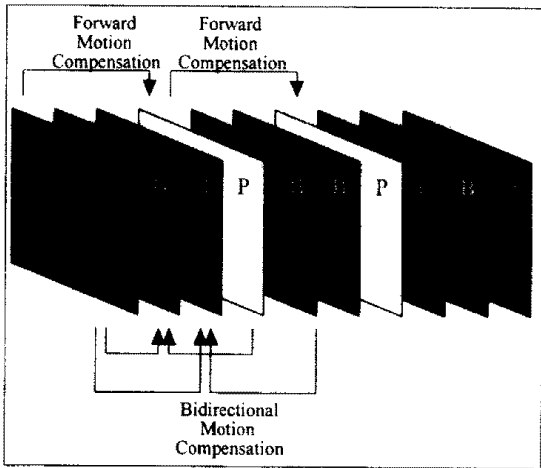


그림 4 MPEG에서의 I, P, B 픽처의 운동 예측 개요도

표 1은 엔코더 및 데코더에 입력되는 픽처 및 해당 픽처 번호를 표시하다. ①, ②는 엔코더로 입력되는 픽처 및 해당 픽처 번호이다. ③, ④는 데코더로 입력되는 픽처 및 ②의 해당 픽처 번호이다. 그림 5는 MPEG 데코더로 입력되는 픽처의 배열을 보여준다. MPEG에서는 I 픽처의 경우 인트라 코딩되므로 운동 벡터의 미 존재로 P 또는 B 픽처와는 달리 문제가 발생한다. GOP(Group Of Picture) 크기 N 이 12인 경우, N 개의 픽처마다 발생하는

①	B	B	P	B	B	I	B	B	P	B	B	P	B	B	P	B	B	I
②	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
③	P	B	B	I	B	B	P	B	B	P	B	B	P	B	B	I	B	B
④	2	0	1	5	3	4	8	6	7	11	9	10	14	12	13	17	15	16

표 1: ①, ②는 엔코더로 입력되는 픽처 및 해당 픽처 번호이다. ③, ④는 데코더로 입력되는 픽처 및 ②의 해당 픽처 번호이다.

I 픽처의 입체영상 구현 없이 2차원 영상을 보여주면 3차원 입체감은 급격히 저하된다. I 픽처의 운동 벡터는 가장 가까운 두 B 픽처의 운동 벡터의 평균값으로 구하였다. 이 과정을 표 1을 이용하여 설명한다. 표 1에의 ①, ②에서 I(5)의 이웃 픽처는 B(4)와 B(6)이다. (I(5)의 5는 ②의 대응하는 번호이다.) 데코더에서 I(5) 픽처의 운동 벡터를 얻기 위해 ③, ④의 B(4)와 B(6)의 운동 벡터가 필요하다. 이 두 B 픽처는 I 픽처 다음에 데코더로 입력된다. 따라서 실제 입체 영상 합성은 픽처 재배열(picture reorder) 후에 수행된다. I 픽처의 운동 벡터는 두 B 픽처의 평균을 이용하는데, B(4)는 양방향 운동 벡터중에서 엔코더에서 I(5)를 이용하여 역방향 예측되었고, B(6)는 반대로 I(5)를 이용하여 순방향 예측되었다. B(4)의 순방향 운동벡터와 B(6)의 역방향 운동벡터가 I(5)의 운동 벡터 계산에 이용된다. I 픽처의 매크로블록 운동벡터는 다음과 같이 계산된다.

$$MV_I(n) = \frac{MV_B^r(n-1) + MV_B^f(n+1)}{2} \quad (1)$$

여기서 $MV_I(n)$ 은 픽처 시퀀스에서 n 번째 I 픽처의 매크로블록의 운동 벡터이다. $MV_B^r(n-1)$ 은 $(n-1)$ 번째 B 픽처의 역방향 운동 벡터이다. $MV_B^f(n+1)$ 은 $(n+1)$ 번째 B 픽처의 순방향 운동 벡터이다.

P 픽처는 자신의 운동 벡터가 있으므로, ①의 P(8)는 ③에서 P(8)의 운동 벡터를 사용한다. 여기서 고려해야 할 사항은 엔코더에서 P 픽처의 운동 예측시 P 픽처와 운동 예측되는 I/P 픽처간의 픽처 간격이다. 입체영상 합성 단계에서 일정 운동량의 유지를 위해 P 픽처의 경우 I/P 및 P와의 간격, M 값을 고려해야한다. 따라서 개선된 P 픽처의 매크로블록 운동 값은 M 을 고려하여 다음과 같이 계산된다.

$$MV_P^s(n) = \frac{MV_P(n)}{M} \quad (2)$$

s 는 스케일(scale)을 의미한다.

B 픽처는 자신의 운동 벡터를 가지고 있으므로 해당되는 운동 벡터를 사용한다. B 픽처는 N 과 M 이 필요하다. $M=3$ 을 가정하면, 표 1에서 ①의 B(3)의 경우 순방향 예측되는 P(2)와는 한 픽처 간격이고, 역방향 예측되는 I(5)와는 두 픽처 간격이다. 반대로 B(4)의 경우 순방향 예측되는 P(2)와는 두 픽처 간격이고, 역방향 예측되는 I(5)와는

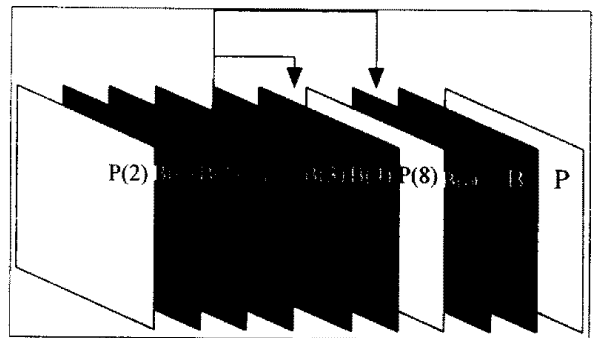


그림 5 : 데코더로 입력되는 픽처 순서임. I 픽처는 후에 나오는 2개의 B 픽처 (화살표로 지정된 픽처)의 운동 벡터가 필요하다.

한 픽처 간격이다. 픽처 간격은 M 과 관계가 있다. 따라서 B(3)의 경우는 운동 값이 다음 식으로 계산된다.

$$MV_B^s(n) = \frac{MV_B^f(n)}{M-2} + \frac{MV_B^r(n)}{M-1} \quad (3)$$

유사하게 B(4)의 운동 값은 다음 식을 이용된다.

$$MV_B^s(n) = \frac{MV_B^f(n)}{M-1} + \frac{MV_B^r(n)}{M-2} \quad (4)$$

다음 단계는 각 픽처의 운동 값이 얻어지면 이를 이용하여 각 픽처에 대응하는 입체영상 합성 단계이다[4]. 매크로블록의 운동 벡터가 얻어지면 수직 운동 성분의 영향

을 없애기 위해 운동 벡터의 놈(norm) d_{MB} 를 구한다.

$$d_{MB} = \sqrt{MV_x^2 + MV_y^2} \quad (5)$$

여기서 (MV_x, MV_y) 는 매크로블록 운동 벡터의 수평 및 수직 성분이다.

식 (5)의 의미는 운동 방향에 관계없이 운동의 크기가 시차를 결정하는 것이다. 만일 카메라 운동만 존재할 때, 카메라의 운동 방향에 관계없이 운동의 크기는 카메라에서 장면까지의 거리에 반비례한다. 즉 가까운 장면은 운동의 크기가 크고, 멀리 있는 장면은 운동의 크기가 작다. 또는 카메라가 정지 상태이고, 물체의 운동시 물체 속도의 크기는 깊이와 비례한다.

4. 입체 영상 합성

3 절에서 각 I, P, B 픽처의 매크로블록의 운동벡터를 구하는 방법과 수평 시차 d_{MB} 를 구하는 방법을 소개하였다. 원영상은 MPEG 데코더에 의해 복원영상으로 만들어진다. 입체 합성영상은 d_{MB} 를 이용하는데 양의 시차를 얻기 위해 픽처에 있는 모든 매크로블록을 수평 이동하여 오른쪽 영상을 합성한다. 양의 시차에선 입체 영상 시청시 스크린에서 후면으로 3D 깊이감이 발생한다. 양의 시차를 얻기 위해서는 실제 수평 시차는 다음식을 이용하여 구해진다[7].

$$d_{MB}^+ = \text{MAX}(d_{MB}) - d_{MB} \quad (6)$$

여기서 $\text{MAX}(d_{MB})$ 는 미리 정해진 값 또는 각 픽처의 매크로블록의 운동 놈의 최대치로 정할 수 있다.

매크로블록 수평 이동시 매크로블록의 겹침 및 매크로블록간에 빈 픽셀 등이 발생하는데, 이를 극복하기 위해 다음 방법을 사용한다. 연속적인 매크로블록을 MB_1 및 MB_2 로 하고, 각각 d_{MB1}^+ 과 d_{MB2}^+ 의 시차를 가지고 있다고 가정한다.

1) d_{MB1}^+ 이 d_{MB2}^+ 와 같을 때, MB_1 과 MB_2 는 같은 시차 값으로 오른쪽으로 수평 이동한다. 따라서 빈 공간이나 매크로블록 겹침이 발생하지 않는다.

2) d_{MB1}^+ 이 d_{MB2}^+ 보다 작을 때, MB_1 의 모든 픽셀들은 d_{MB1}^+ 만큼 이동한다. 이 때 빈 픽셀들이 발생하는데 인터폴레이션(interpolation) 방법을 이용하여 보간한다.

3) d_{MB1}^+ 이 d_{MB2}^+ 보다 클 때, 먼저 MB_2 를 이동하고, 다음에 MB_1 을 이동한다. 이 때 MB_2 의 왼쪽에 있는 $(d_{MB2}^+ - d_{MB1}^+)$ 개의 픽셀들은 MB_1 에 의해 덮여진다.

이 과정이 완료되면 합성영상의 합성된다. 적절한 3D 시청기기를 이용하여 복원 영상은 왼쪽 눈에, 합성 영상은 오른쪽 눈에 디스플레이하면 3차원 입체영상을 시청할 수 있다.

그림 6은 본 논문에서 제안한 MPEG 입체 영상 변환 시스템의 구조도이다. 구조도의 왼쪽은 기존 MPEG 데코

더이고 오른쪽은 입체영상 합성에 사용된다. MV1 Store와 MV2 Store는 I 픽처의 운동벡터를 구하기 위해 필요한 두 B 픽처의 운동벡터를 저장한다. 제안한 입체 변환 알고리즘은 입체 변환에서 수행된다.

5. 실험 결과

제안한 알고리즘을 실제 영상에 적용하였다. 영상 크기가 352X240인 실험 영상 50장을 4:2:2 YUV 영상으로 만든 후 MPEG-2 엔코더 프로그램을 이용하여 압축 데이터를 생성하였다. GOP 크기 N 은 12이고 M 값은 3으로 하였다. 다시 MPEG-2 데코더 프로그램을 이용하여 입체영상을 생성하였다. MPEG-2 데코더를 이용하여 복원 영상이 만들어지고, 압축 데이터에서 추출한 P, B 픽처의 매크로블록의 운동 벡터를 이용하여 I 픽처의 입체영상을 만들었다. P, B 픽처도 제안한 알고리즘을 이용하여 입체영상을 생성하였다.

그림 7, 8, 9는 각각 I 픽처의 복원 영상, 입체 합성 영상, 및 두 영상의 비월영상(interlaced)을 보여준다. 두 영상의 시차 차이는 비월 영상에서 쉽게 관찰할 수 있다. 움직이는 물고기에서 운동으로 인한 두 영상의 시차가 관찰되고 정지상태의 배경은 시차가 없다. 앞에 있는 물고기들은 시차가 크므로 식 (6)에 의해 시차 값이 반전되므로 실제 구현시 시차가 작아진다. 따라서 스크린에 가깝게 보이고 뒤의 배경은 시차가 0에 가깝다. 마찬가지로 식 (6)에 의해 시차가 커지므로 실제 입체 시청시에는 스크린에서 멀리 보인다. 문자영역에서 시차 변화의 artifact가 관찰되는데 매크로블록 단위로 수평이동되기 때문에 발생한다.

그림 10, 11, 12는 각각 P픽처의 복원 영상, 입체 합성 영상, 및 두 영상의 비월영상을 보여준다. 시차차이는 비월 영상에서 보여진다. B 픽처 입체 변환 결과는 그림 13, 14, 15에서 보여진다. 실험결과에서 보는 바와 같이 매크로블록별 수평 시차 값에 의해 수평이동되므로 블록 경계가 부드럽지 못하다. 이는 좀더 세밀한 수평 이동 방법이 필요하다고 할 수 있다.

6. 결론

본 논문에서는 기존의 NTSC 영상의 2D/3D 영상변환과 달리 MPEG 영상의 변환 방법을 제안하였다. 이를 위해 MPEG 데코더에서 I, B, P 픽처의 운동 벡터 추출 방법을 제안하였다. 얻어진 운동벡터를 이용하여 각 픽처의 입체영상 합성방법을 제안하였다. 제안한 방법은 다음의 두가지 장점이 있다. MPEG 압축데이터에서 추출한 운동 벡터를 이용하여 입체영상을 합성함으로써 수행 시간을 단축하였고, 또한 수평시차 변환을 이용하여 수직 및 빠른 운동의 문제를 해결하였다.

참고 문헌

[1] B. J. Garcia, "Approaches to stereoscopic video based on spatial-temporal interpolation", SPIE Vol. 2 635, Photonic West, 1990.
 [2] T. Okino and et. al, "New television with 2D/3D image conversion techniques", SPIE Vol.2653, Photonic West, 1995.
 [3] Y. Matsumoto and et al., "Conversion system of monocular image conversion technologies", SPIE Vol. 3012, 1997
 [4] M. B. Kim, et al., "Stereo conversion of monoscopic video by the transformation of vertical-to-horizontal disparity, SPIE Photonic West, Vol. 3295, Jan. 1998.
 [5] 김만배, 최광철, 김도균, 송문섭, "2차원 동영상의 3차원 입체 동영상 변환", 제11회 신호처리합동학술대회, 서울대학교, 1998년 10월.
 [6] K. R. Rao and J. J. Hwang, "Techniques & Standard for image, video and audio coding", Prentice Hall, 1996.
 [7] D. F. McAllister, "Stereo computer graphics and other true 3D Technologies", Princeton University Press, Princeton, NJ, 1993. (edited)

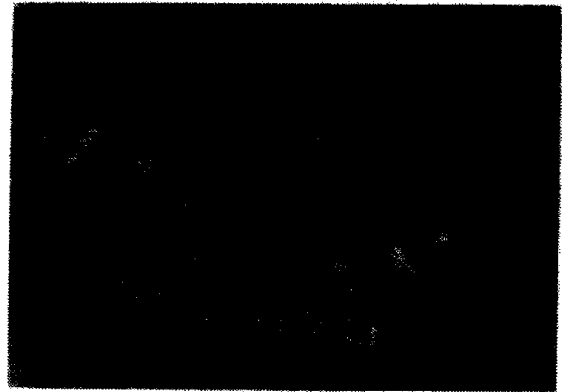


그림 8: I 픽처의 입체 합성 영상

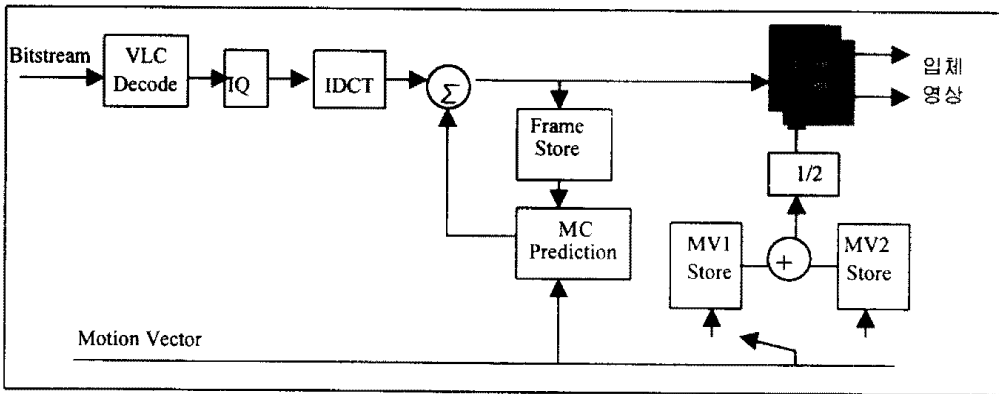


그림 6: MPEG 입체 영상 변환 시스템 구조도

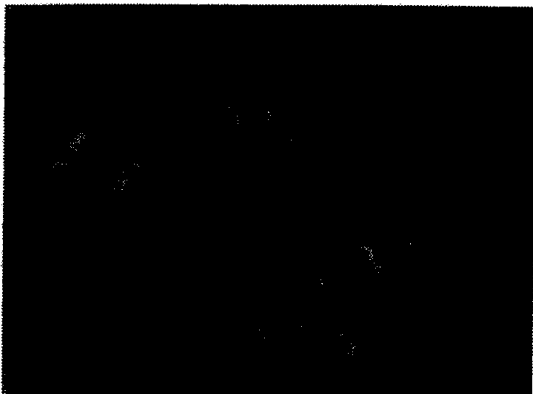


그림 7: 복원된 I 픽처



그림 9: 그림 7과 그림 8의 이월영상

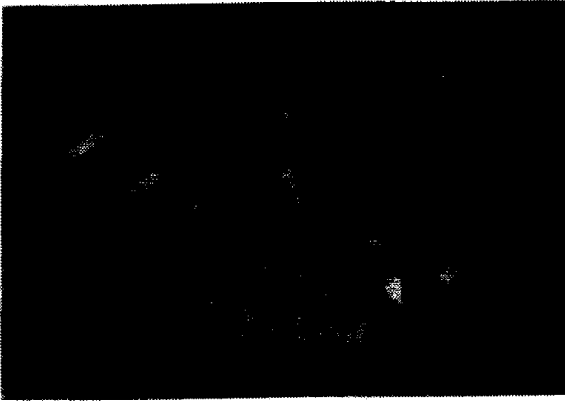


그림 10: 복원된 P 픽처



그림 13: 복원된 B 픽처



그림 11: P 픽처의 입체 합성영상



그림 14: B 픽처의 입체 합성 영상

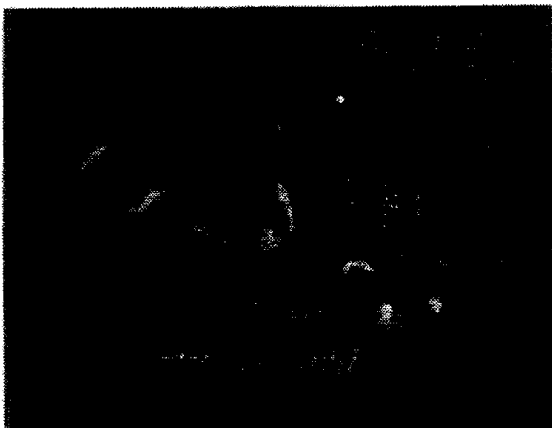


그림 12: 그림 10과 그림 11의 이월영상



그림 15: 그림 13과 그림 14의 이월 영상