

## 초 저속 비트율을 갖는 영상 부호화를 위한 2차원 삼각형 그물 기반 움직임 보상 방법

正會員 김 학 수\*, 이 규 원\*, 박 규 태\*

### A 2-D Triangular Mesh Based Motion Compensation For Very Low Bit Rate Video Coding

Hak Soo Kim\*, Kyu Won Lee\*, Kyu Tae Park\* *Regular Members*

#### 요 약

본 논문은 2차원 삼각형 그물 기반 움직임 보상과 두 단계의 격자 점 움직임 추정을 근간으로 한 새로운 영상 부호화기를 제안한다. 제안한 방법으로 압축 효율과 재구성된 영상의 화질을 향상시킬 수 있다. 각각의 삼각형 조각에 대한 움직임 보상은 삼각형 격자 점들에서의 움직임 벡터를 이용한 영상 왜핑(warping) 기법에 의해 행해진다. 예측 오류 부호화와 발생 비트율 제어는 MPEG-4 VM 3.0 규정을 준수했다. 모의 실험 결과, 제안한 부호화기가 복잡도가 낮은 반면, 복호화된 영상의 화질이 기존의 블록 정합을 기반으로 한 부호화기에 비해 향상됨을 보여 줄 수 있었다.

#### ABSTRACT

This paper presents a new video codec which is based on 2-D triangular mesh-based motion compensation and two step grid point motion estimation. With this approach the efficiency of compression and the quality of reconstructed images are improved. The compensation of motion for each triangular patch is performed by image warping using motion vectors at the grid points. The prediction error coding and the rate control meet MPEG-4 VM 3.0 specification. The experimental results show that the codec system proposed is simple in complexity and moreover, the quality of decoded images is improved.

\*연세대학교 전자공학과  
論文番號:97220-0630  
接受日年:1997年 6月 30日

## I. 서 론

전통적인 영상 부호화기에서, 블럭 정합 알고리즘(block matching algorithm)은 움직임 추정과 보상을 위해 가장 널리 사용되고 있으며, H.261, H.263, MPEG-1, MPEG-2와 같은 기존의 부호화기 표준들에서 채택되고 있다. 블럭 정합 알고리즘은 움직임을 추정하고자 하는 프레임을 일정 크기의 사각형 블럭으로 분할한 뒤 각 블럭 당 오차가 가장 적은 블럭을 기준 프레임으로부터 찾아내는 것으로써, 전송 시, 찾아진 블럭과 현재 블럭과의 벡터를 부호화하여 전송한다. 이 방법은 구조가 간단하여 구현하기가 용이하다는 큰 장점을 갖고 있는 반면에 물체의 움직임들을 블럭들의 평행이동 만으로 묘사하는 움직임 모델에 기반을 두고 있기 때문에 블럭 현상등과 같은 많은 문제점들을 야기시켰다. 이러한 결점들을 극복하기 위해 보다 복잡한 움직임 모델에 기반을 둔 많은 부호화 방법들이 제안되어졌다. 제안한 많은 부호화 방법들을 크게 다음과 같은 세가지 부류로 분류할 수 있는데, 첫째가 3차원 모델 기반 부호화[1], 둘째가 객체 기반 부호화[2], [3] 그리고 셋째가 2차원 그물 기반 부호화이다[4], [5].

3차원 모델 기반 부호화에서는 미리 정의된 3차원 모델이 부호화기와 복호화기에 각각 같이 사용된다. 이 부호화 방법의 기본 구조는 첫번째 프레임을 전송한 후 차후 부호화될 객체의 3차원 구조를 추정하는 것이다. 두번째 프레임부터 부호화기는 각각의 프레임을 분석하여 파라미터를 구한 후, 그 구해진 파라미터만을 영상을 대신해서 전송하게 된다. 즉, 이 부호화기는 이미 정의된 3차원 모델을 기반으로 입력된 영상을 분석하여 분석된 파라미터를 전송하며, 복호화기는 전송받은 파라미터를 이용하여 3차원 모델의 형태를 변형하는 것에 의해 출력 영상을 합성한다. 3차원 모델은 실제 움직임을 대체적으로 잘 표현하나, 기존의 영상 분석 기술을 가지고 자연 영상들로부터 움직임 파라미터를 정확하게 추출하는 것은 아직도 많은 어려움을 가지고 있다. 게다가, 3차원 모델이 영상의 내용에 따라 적응적으로 변해야 하기 때문에 이 방법을 이용하여 범용적인 영상 부호화기를 구현하는 것은 상당히 힘들다.

객체 기반 부호화는 미리 정의된 움직임 모델을 이

용하지 않는다. 이러한 장점 때문에, 이 부호화기는 보다 일반적인 부류의 영상들에 적용할 수 있다. 부호화기는 연속적인 영상들의 각각을 임의의 형태를 갖는 움직임 객체들로 분할하고 각각의 객체를 움직임, 모양, 색으로 정의한 파라미터들의 세가지 집합들로 묘사한다. 여기서 색 파라미터들은 객체 표면의 휘도와 색차 값을 말한다. 움직임들이 영상으로부터 추출한 원천 모델에 의해 정확하게 묘사되어지는 객체에 대해서는 단지 형태와 움직임 파라미터들만을 전송하고, 원천 모델에 의해 정확하게 묘사되어질 수 없는 객체들-모델 실패(model failure) 객체들을 위해서는 움직임 파라미터대신 색 파라미터들을 전송한다. 이 부호화 방법은 부분적으로는 효율적이나, 많은 실제 영상들에 적용함에 있어서, 부호화 효율이 H.263과 같은 기존의 영상 부호화 방법에 비해 우수하지가 않다.

2차원 그물 기반 부호화의 핵심 아이디어는 영상을 삼각형 조각들로 분할한 뒤, 객체의 움직임을 삼각형 조각들의 변형에 의해 묘사하는 것이다. 삼각형 조각의 변형은 2차원 아핀 변형(affine transform)으로 표현할 수 있으며, 직교 투영(orthographic projection)을 가정한다면, 3차원 공간에 있는 2차원 강체 평면(rigid plane)상에서의 화소의 움직임으로 표현할 수 있다. 그러므로 영상 내에 있는 객체들의 3차원 구조를 강체 평면들으로써 근사화 한다면 2차원 그물 기반으로서의 접근은 객체 움직임을 훌륭한 근사화 결과를 얻을 수 있다. 이 방법의 또 다른 장점은 예측 영상 내에서 불연속성이 없기 때문에 사람 눈에 덜 거슬리는 영상을 만들어낼 수가 있다는 것이다. 나카야[4], [5]는 그물 구조에 근간한 움직임 추정과 보상을 위한 HMA(hexagonal matching algorithm)를 제안하였다. 이 방법에서, 격자 점들에서의 움직임 추정은 두 단계로 구성되는데, 첫째는 움직임 벡터들을 대충 찾기 위한 블럭 정합 알고리즘이며, 둘째는 움직임 벡터를 세밀하게 찾기 위한 예측 오차의 반복적인 부분 최소화 방법이다. 이 방법은 모델 기반 부호화와 객체 기반 부호화에 비해 단순하지만 블럭 정합 알고리즘보다 복잡하다.

지금까지 언급한 영상 부호화 방법들의 단점들을 극복하기 위해 본 논문은 삼각형 그물 기반 움직임 보상과 간단한 2단계의 움직임 추정 방법을 갖는 영

상 부호화기를 제안한다. 구현된 부호화기의 전체 구조는 H.263과 동일하나, H.263의 움직임 추정부와 움직임 보상부가 제안한 방법으로 대체되었다. 제안한 부호화기의 복잡도는 기존의 H.263과 거의 같으며, 화질은 같은 조건의 H.263보다 우월하다.

## II. 2차원 삼각형 그물 기반 움직임 추정 방법들

### 1. 변화도 기반(Gradient-based) 움직임 추정 방법

Brusewitz[6]는 삼각형 기반 움직임 보상을 위한 변화도 기반 움직임 추정 방법을 제안하였다. 이 방법에서, 격자 점에서의 움직임 추정은 2단계로 구성되어 있다. 1단계는 블럭 정합에 의한 대략적인 움직임 추정이며, 2단계는 변화도 기반 방법에 의한 세밀한 움직임 추정이다. 2단계 방법은 식 (1)의 광류 방정식(optical flow equation)을 이용하여 행해진다.

$$\frac{\partial I(x, y, t)}{\partial x} \cdot u(x, y) + \frac{\partial I(x, y, t)}{\partial y} \cdot v(x, y) + \frac{\partial I(x, y, t)}{\partial t} = 0 \quad (1)$$

여기서  $I(x, y, t)$ 는 위치  $(x, y)$ 의 시간  $t$ 에서의 휘도 값이며,  $u(x, y)$ 와  $v(x, y)$ 는 각각 위치  $(x, y)$ 에서  $x$ 축과  $y$ 축으로의 움직임 벡터이다.

격자 점들의 세밀한 움직임 벡터들은 예측된 영상과 현재 영상간의 평균자승오차(mean square error)를 최소화 하는 것에 의해 계산되어진다. 이 방법은 삼각형 조각 내에 있는 각 점의  $(u, v)$ 가 삼각형의 3개 꼭지점의 움직임 벡터들을 선형 보간하는 것에 의해 얻어질 수 있기 때문에 가능하다. 삼각형의 각 꼭지점을  $A, B, C$ 로 놓고 각각의 움직임을  $\mathbf{d}_a = (u_a, v_a)$ ,  $\mathbf{d}_b = (u_b, v_b)$ ,  $\mathbf{d}_c = (u_c, v_c)$ 라 하면, 삼각형 내부의 위치  $\mathbf{r} = (x, y)$ 에서의 움직임 벡터  $\mathbf{d}_r = (u_r, v_r)$ 은 식 (2)와 같이 표현할 수 있다.

$$\mathbf{d}_r = (1 - p_r - q_r) \mathbf{d}_a + p_r \mathbf{d}_b + q_r \mathbf{d}_c \quad (2)$$

식 (2)의  $p_r, q_r$ 은 식 (3)에서와 같이 삼각형 내부의 위치  $\mathbf{r}$ 을 표현할 때 사용되는 방향 벡터의 크기이다.

$$\mathbf{r} = \mathbf{a} + p_r \overrightarrow{AB} + q_r \overrightarrow{AC} \quad (3)$$

여기서  $\mathbf{a}$ 는 꼭지점  $A$ 의 위치 벡터이다.

식 (1)을 이용,  $PI_k(x, y)$ (광류 방정식을 가정할 때, 프레임  $k$ 에서의 예측된 영상)를 식 (4)와 같이 정의한다.

$$PI_k(x, y) = I_{k-1}(x, y) - \frac{\partial I_{k-1}(x, y)}{\partial x} \cdot u(x, y) - \frac{\partial I_{k-1}(x, y)}{\partial y} \cdot v(x, y) \quad (4)$$

여기서  $I_k(x, y)$ 는 프레임  $k$ 에서의 화소 위치  $(x, y)$ 의 휘도 값을 나타낸다.

식 (4)는 식 (2)를 이용하여 식 (5)와 같이 표현할 수 있으며, 약어  $I_x(x, y) = \frac{\partial I_{k-1}(x, y)}{\partial x}$  를  $I_x(x, y) = \frac{\partial I_{k-1}(x, y)}{\partial x}$  를 각각 나타낸다.

$$PI_k(x_r, y_r) = I_{k-1}(x_r, y_r) + \mathbf{A}_r \cdot (u_a \ v_a \ u_b \ v_b \ u_c \ v_c)^T \quad (5)$$

여기서

$$\mathbf{A}_r = -((1 - p_r - q_r) I_x(x_r, y_r) \ (1 - p_r - q_r) I_y(x_r, y_r) \ p_r I_x(x_r, y_r) \ p_r I_y(x_r, y_r) \ q_r I_x(x_r, y_r) \ q_r I_y(x_r, y_r))$$

이다.

영상내에  $n$ 개의 격자점이 있다고 가정하면, 식 (5)를 다음과 같이 쓸 수 있다.

$$PI_k(x_r, y_r) = I_{k-1}(x_r, y_r) + \mathbf{A}(x_r, y_r) \mathbf{D} \quad (6)$$

여기서

$$\mathbf{D} = (u_1 \ v_1 \ u_2 \ v_2 \ \dots \ u_n \ v_n)^T$$

이며,  $(u_s, v_s)$ 는  $s$ 번째 격자점의 움직임 벡터를 나타낸다.

세밀한 움직임 벡터  $\hat{\mathbf{D}}$ 는 자승오차의 합을 최소화 하는 것에 의해 추정되어진다.

$$\sum_{x, y} (I_k(x, y) - PI_k(x, y))^2 \quad (7)$$

식 (7)에 최소자승법(least square method)을 적용하면,  $\hat{\mathbf{D}}$ 는 (8)식으로 부터 구해진다.

$$\sum_{x, y} \mathbf{A}(x, y)^T \mathbf{A}(x, y) \hat{\mathbf{D}} = \sum_{x, y} (I_k(x, y) - I_{k-1}(x, y)) \mathbf{A}(x, y)^T \quad (8)$$

2단계의 세밀한 움직임 추정은 예측 오차를 줄이기 위해 반복되어질 수 있다. 그러나 영상 데이터가 항상 식 (1)과 같은 광류 방정식을 만족하는 것이 아니기 때문에 반복이 항상 예측 오차의 감소를 가져오지는 않는다. 특히 이 알고리즘을 영상 부호화기의 움직임 추정부에 적용할 때는 현재 영상과 비교 대상이 되는 재구성된 전의 영상은 이미 많이 훼손된 영상이기 때문에 식 (1)의 조건이 성립되지 않는다. 따라서, 이 방법은 영상 부호화기의 움직임 추정부에 적용하기에는 적절치 않다. 또한 이 방법은 계산량이 엄청나게 많기 때문에 실시간 영상 부호화기에 적용하기에는 상당히 많은 문제점이 있다.

2. HMA (Hexagonal Matching Algorithm)

나카야[4], [5]는 BMA의 국소 최소화 방법을 AFMC (affine motion compensation)에 적용한 HMA라는 반복 움직임 추정 방법을 제안하였다. 이 방법에서, 격자점의 움직임 추정은 두 단계로 이루어져 있다. 첫 번째 단계는 대략적인 움직임을 찾기 위한 BMA 방법이며, 두 번째 단계는 움직임 벡터를 세밀하게 찾기 위한 예측 오차의 반복적인 국소 최소화 방법이다. 두 번째 세밀화 단계를 자세하게 설명하면 다음과 같다.

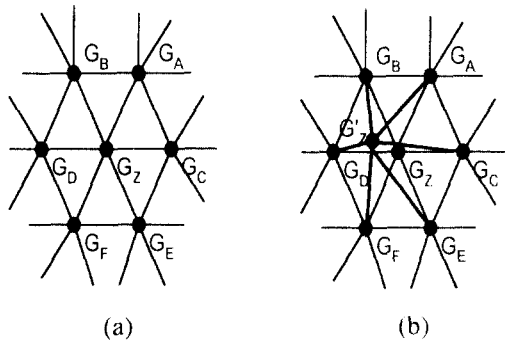


그림 1. Hexagonal matching 방법  
Fig. 1. Hexagonal matching method

(1) 그림 1(a)에서 보는 바와 같이 격자점  $G_Z$ 의 위치를 수정하고자 한다.  $G_Z$  주변에 있는 6개의 격자점,  $G_A \sim G_F$ 들은 고정시킨다. 차후의 동작들은 이러한 6개의 격자점들로 구성된 육각형 내에 있는 영역으로 한정된다.

- (2)  $G_Z$  점을 근처의 위치  $G'_Z$  로 이동하면서, 육각형 내에 있는 삼각형들을 변형시킨다. 전의 프레임의 변형된 삼각형 조각들을 현재 프레임의 변형 안된 삼각형 조각들로 와핑(warping)하는 것에 의해 육각형 내에 있는 예측된 영상을 합성한다.
- (3) 육각형 내에 있는 현재 프레임과 예측된 프레임과의 평균절대차를 구한다. 이러한 오차가 최소가 되는 최적의 위치를 찾는다. 최적의 위치  $G'_Z$  를  $G_Z$ 의 새로운 위치로 등록한 후 다음 격자점의 최적 위치를 찾기 위해 (1)로 다시 돌아간다.

각각의 세밀화 작업 이후, 현재 프레임과 예측된 프레임 사이의 차이는 감소되거나, 변하지 않고 일정하게 유지된다. 따라서, 세밀화 과정은 모든 격자점들이 국부적 혹은 전체적 최소값에 수렴할 때 까지 반복된다. 이상의 설명에서 알 수 있듯이 HMA는 변화도 기반 움직임 추정 방법의 근본적인 문제점인 광류 방정식의 영상 부호화기 부적합성 문제를 해결하였으나, 계산량은 반복에 의한 세밀한 움직임 추정 방법으로 인해 상당히 많아 실시간 부호화기 구현에 부적합하다.

III. 제안한 움직임 추정 및 보상 방법

영상 부호화기의 움직임 추정부에서, 기존의 방법인 변화도 기반 움직임 추정 방법은 광류 방정식의 부적절성 및 많은 계산량으로 인해 실시간 구현이 어려운 문제점들이 있었으며, HMA 방법도 세밀한 움직임 추정에 있어서 반복 계산으로 인한 많은 계산량을 초래해 실시간 구현에 많은 문제점을 남겼다. 따라서 본 논문에서 제안한 방법은 움직임을 빠르게 하기 위해 세밀한 움직임을 찾는 단계에서 HMA 처럼 반복 기법을 사용한 것이 아니라 첫 번째 단계에서 찾아진 움직임 벡터를 근거로 하여 그 주변 반화소(half pixel) 단위로 국소의 범위에서 BMA 기법을 이용하여 움직임을 추정하였으며, 움직임 보상 시에는 BMA 방법의 움직임 보상에서 주로 문제시 됐던 불력간의 끊어짐(discontinuity)으로 인한 눈에 거슬리는 현상을 없애기 위해 각각의 삼각형 단위로 영상 와핑에 의한 영상 합성을 행하였다.

3.1. 움직임 추정 방법

3.1.1. 이차원 삼각형 그물 구조의 초기화

두 프레임간의 움직임을 추정하기 위해, 휘도 성분의 프레임을 16x16 매크로 블럭으로 분할한 후 각각의 매크로 블럭을 그림 2에서 보는 바와 같이 두개의 삼각형으로 나눈다. 프레임의 가장자리에 위치한 격자점들은 고정되어, 다른 격자점들은 움직임이 발생하면 이동하여 결과적으로 변형된 삼각형들을 만들게 된다. 변형된 삼각형들로부터 아핀 변환과 보간(interpolation)을 이용, 이전 프레임으로부터 움직임 추정된 프레임을 합성한다.

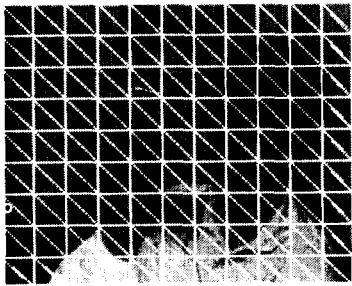


그림 2. 2차원 그물 구조의 초기화  
Fig. 2. Initialization of 2-D triangular meshes

3.2. 움직임 벡터의 추정

현재의 원래 프레임과 이전 재구성 프레임 사이의 움직임 추정은 삼각형의 각 격자점을 중심으로 2단계 계층구조 블럭 정합을 이용하여 행해진다.

3.2.1. 1단계 움직임 추정

움직임 추정은 영상의 휘도 성분에 대해 정수 화소 위치 단위로 행해진다. 움직임 벡터를 찾기 위한 비교는 입력되는 현재 프레임의 블럭과 이전의 재구성된 프레임에 있는 천이 된 블럭 사이에서 이루어진다. 즉, 현재 프레임 내에 있는 삼각형 격자블 중심으로 16x16 매크로 블럭을 설정한 후, 그 위치를 중심으로 찾는 영역이 [-15, +15]내에서 전 영역 찾기(full search)를 한다. 매크로 블럭에서의 예측 에러는 SAD (sum of absolute difference)를 이용하여 평가되며, 움직임 벡터 (u, v)를 구하기 위한 SAD는 다음과 같이 정의된다.

$$SAD_N(u, v) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |prev(i+u, j+v) - orig(i, j)| \quad (9)$$

u, v = "up to [-15, 15]", N = 16

3.2.2. 2단계 움직임 추정

1단계 움직임 추정 이후 2단계로써 반화소(half pixel) 움직임 추정이 행해진다. 이 추정 또한 영상의 휘도 성분상에서 이루어지며, 찾는 영역은 1단계 추정에서 찾아진 움직임 벡터를 근간으로 이동 위치를 계산한 후, 그 위치를 중심으로 반화소로 설정한다. 반화소 값들은 그림 3에서 보는 바와 같이 이중 선형 보간법(bilinear interpolation)에 의해서 구한다.

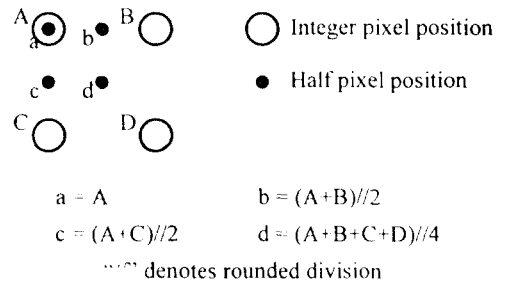


그림 3. 반화소 추정을 위한 보간 방법  
Fig. 3. Interpolation scheme for half sample search

3.3. 움직임 보상 방법

각각의 삼각형 조각들을 위한 움직임 보상은 각 삼각형의 꼭지점들에서 발생한 움직임 벡터들을 이용한 영상 왜핑(warping)에 의해 행해진다. 이 과정은 두 단계를 거치는데, 첫번째 단계가 삼각형 조각들의 변형이며, 두번째 단계가 영상 합성이다.

3.3.1. 삼각형 조각의 변형

현재 프레임에 있는 삼각형의 꼭지점을 각각 A, B, C라하고 이전 프레임에 있는 삼각형과 대응되는 꼭지점을 각각 A', B', C'라 하자. 삼각형의 각 꼭지점에서 움직임이 발생하면, 삼각형들의 모양이 그림 4에서 보는 바와 같이 변형된다. 만약 전 프레임에 있는 X'가 현재 프레임에 있는 X로 이동한다고 가정하면 X에서 X'로의 움직임 벡터 (u, v)를 다음과 같이 유도할 수 있다.

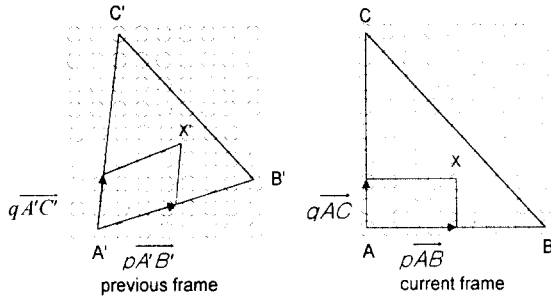


그림 4. 삼각형의 변형. 이전 프레임에 있는 X'가 현재 프레임에 있는 X로 이동함.

Fig. 4. Deformation of a triangle. X' in the previous frame moves to X in the current frame

그림 4에서, 삼각형들의 각 꼭지점인 A, B, C, A', B', C'들과 각 삼각형 내부의 점인 X, X'들의 위치 벡터가 각각 다음과 같다고 하자.

$$\begin{aligned} A' &= (x'_a, y'_a), B' = (x'_b, y'_b), C' = (x'_c, y'_c) \\ A &= (x_a, y_a), B = (x_b, y_b), C = (x_c, y_c) \\ X' &= (x', y'), X = (x, y) \end{aligned}$$

위와 같은 위치 벡터들로부터 식 (10)을 구할 수 있다.

$$\begin{pmatrix} u \\ v \end{pmatrix} = (1-p-q) \begin{pmatrix} u_a \\ v_a \end{pmatrix} + p \begin{pmatrix} u_b \\ v_b \end{pmatrix} + q \begin{pmatrix} u_c \\ v_c \end{pmatrix} \quad (10)$$

여기서,

$$\begin{aligned} u &= x - x', v = y - y' \\ u_a &= x_a - x'_a, v_a = y_a - y'_a \\ u_b &= x_b - x'_b, v_b = y_b - y'_b \\ u_c &= x_c - x'_c, v_c = y_c - y'_c \\ p &= ((x-x_a)v_{y1} - (y-y_a)v_{x1}) / (v_{x0}v_{y1} - v_{x1}v_{y0}) \\ q &= ((y-y_a)v_{x0} - (x-x_a)v_{y0}) / (v_{x0}v_{y1} - v_{x1}v_{y0}) \\ v_{x0} &= x_b - x_a, v_{y0} = y_b - y_a \\ v_{x1} &= x_c - x_a, v_{y1} = y_c - y_a \\ 0 \leq p \leq 1, 0 \leq q \leq 1, 0 \leq p+q \leq 1 \end{aligned}$$

삼각형의 각 꼭지점에서 발생하는 움직임 벡터  $(u_a, v_a)$ ,  $(u_b, v_b)$ ,  $(u_c, v_c)$ 는 움직임 벡터 추정 과정을 통해 구할 수 있으며, 삼각형 내부의 각 화소 X에서 발생하는 움직임 벡터는 식 (10)을 통해 구할 수 있다.

### 3.4. 영상 합성

영상 합성은 각각의 삼각형 꼭지점에서 발생하는 움직임 벡터를 이용, 영상 와핑 기법을 통해 행해진다. 즉, 그림 4에서 보는 바와 같이 움직임 벡터에 따라 이전 프레임의 삼각형을 변형하고, 현 프레임의 삼각형 내부에 있는 모든 화소에 대해 각각의 위치 X와 대응되는 이전 프레임의 화소 위치 X'를 식 (10)을 이용하여 구한다. 이때 구해진 위치 X'가 정수가 아닐 수 있으므로, 그림 5에서 보는 바와 같이, 움직임 보상 프레임의 화소 값  $P(x, y)$ 는 식 (11)와 같은 이중 선형 보간법에 의해 구한다.

$$P(x, y) = ([y'] + 1 - y')R_1 + (y' - [y'])R_2 \quad (11)$$

여기서,

$$\begin{aligned} R_1 &= ([x'] + 1 - x')R([x'], [y']) + (x' - [x'])R([x'] + 1, [y']) \\ R_2 &= ([x'] + 1 - x')R([x'], [y'] + 1) + (x' - [x'])R([x'] + 1, [y'] + 1) \end{aligned}$$

$R([x'], [y']), R([x'] + 1, [y']), R([x'], [y'] + 1), R([x'] + 1, [y'] + 1)$ 는 위치  $X' = (x, y)$  주변의 4 화소 값이며,  $[x']$ 는 실수  $x'$ 를 넘지않는 정수의 최대값을 나타낸다.

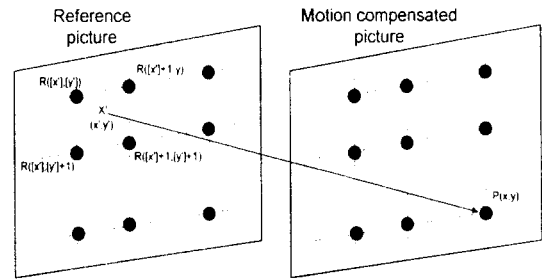


그림 5. 움직임 보상 영상 예측  
Fig. 5. Motion-compensated image prediction

## IV. 모의 실험

제안한 방법의 성능을 입증하기 위해 컴퓨터 모의 실험을 행하였다. 성능은 8x8 움직임 보상이 사용되지 않은, 같은 조건하의 H.263 부호화기와 비교하였다. 두가지 방법 모두 제한된 움직임 벡터 모드가 사

용되었으며, 사용된 부호화 구조는 IPPPP... 로써 PB-프레임 모드가 적용되지 않았다. 구분 기반 산술 부호화(syntax-based arithmetic coding) 모드가 쓰이지 않은 반면, 양자화된 DCT(discrete cosine transform) 계수들의 인덱스들을 부호화하기 위해 H.263의 3차원 가변 길이 부호화가 사용되었다. 발생 비트율 제어 위한 양자화 단계 조정 알고리즘은 MPEG-4 VM 3.0 [7]을 따랐다.

표 1. 모의 실험 환경

Table 1. Simulation Conditions

Bitrate, kbps	10	24
Resolution	QCIF	QCIF
Frame Rate(Hz)	7.5	10
Sequence	Hall Container M&Dd	Hall Container M&D Silent

표 1은 모의 실험 환경을 보여준다. 표 1에서 보는 바와 같이 4 종류의 영상 시퀀스가 사용되었는데, 이들은 각각 "Hall Monitor," "Container Ship," "Mother & Daughter," "Silent Voice"이다. 처음 3개의 시퀀스들은 10 kbps와 24 kbps 각각에 대해 부호화 하였으며, 마지막 것은 24 kbps로 부호화 하였다. 모든 실험 영상의 규격은 QCIF(Y:176x144, U, V:88x72)를, 각각의 시퀀스는 10초 분량의 영상인 300 프레임에 대상으로 하였으며, 이중 10 kbps인 경우는 7.5 Hz로 24 kbps인 경우는 10 Hz로 프레임 비를 조절하였다.

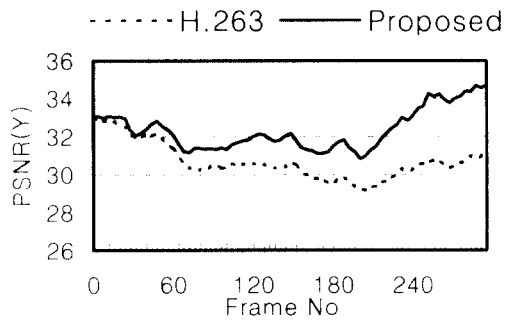
표 2. 모의 실험 결과

Table 2. Simulation Results

Bitrates	Resolution	Sequences	H.263					2-D Triangular Mesh					
			PSNR			Mean	Q	Bitrates	PSNR			Mean	Bitrates
			Y	U	V				Y	U	V		
10kbps	QCIF	Hall	29.06	35.50	38.95	19.20	10.03	29.51	35.99	38.85	18.65	10.20	
		Container	29.18	36.33	35.53	17.88	10.01	29.43	36.79	36.02	17.73	9.99	
		M&D	30.69	37.01	36.99	14.81	10.01	32.36	38.72	39.54	13.83	10.03	
24kbps	QCIF	Hall	32.66	36.92	39.15	10.41	23.98	33.29	37.51	39.69	10.07	24.17	
		Container	31.77	38.00	37.27	11.44	24.01	32.93	39.06	38.20	9.23	23.98	
		M&D	32.89	38.30	38.66	9.52	24.11	35.04	40.36	41.10	8.37	24.23	
		Silent	29.87	34.36	35.56	13.93	23.99	30.80	35.47	36.78	14.01	23.60	

제안한 부호화기의 성능을 확인하기 위해 2차원 삼각형 그물 기반 움직임 보상을 갖는 제안한 방법과 BMA 방법을 사용하는 H.263과의 성능을 비교하였다. 성능 비교 요소로써, 각 실험 환경에서의 영상 시퀀스 당 휘도 성분(Y) 및 색차 성분(U, V)의 PSNR (peak signal-to-noise ratio)을 조사하였으며, 평균 양자화 단계 값(Q) 및 발생 비트율(bitrate)을 구하였다. 표 2는 위에서 언급한 성능 비교 결과를 보여주는데, 이 결과들로부터 제안한 방법이 기존의 H.263에 비해 모든 실험 조건에서 우수함을 나타냄을 알 수 있었다. 그림 6은 비교되는 두가지 방법에 대해 "Mother & Daughter," "Silent Voice," "Hall Monitor," "Container Ship" 시퀀스의 프레임 당 PSNR 결과를 보여주는데, 첫째 시퀀스는 10 kbps로, 나머지 시퀀스들은 24 kbps로 각각 부호화 하였다. 그림 6에서 보는 바와 같이, 실험 대상이 된 영상 시퀀스들의 모든 프레임에 대해 제안한 방법이 H.263 보다 PSNR이 우수한 결과를 나타냈다.

Mother & Daughter(10kbps)



(a)

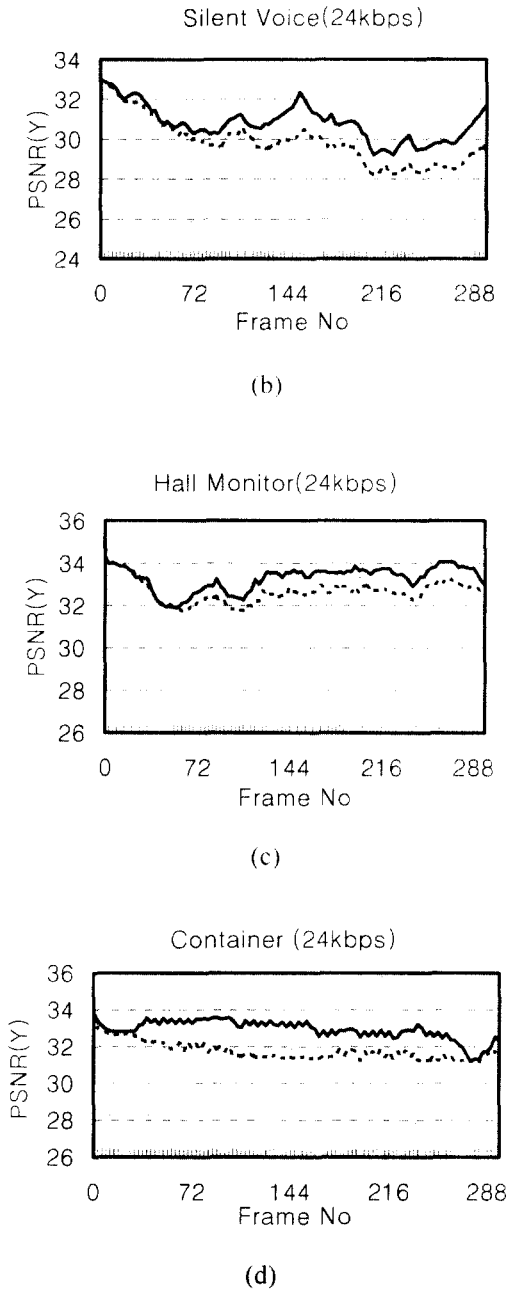
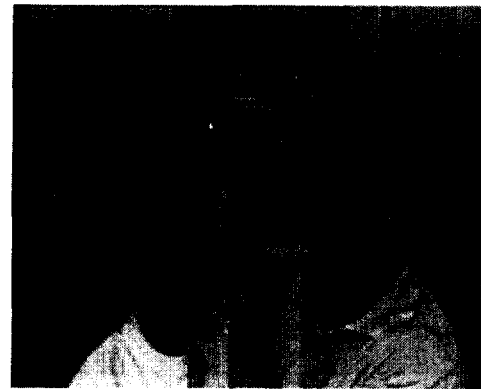
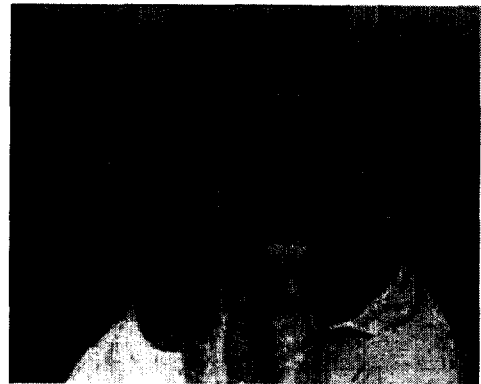


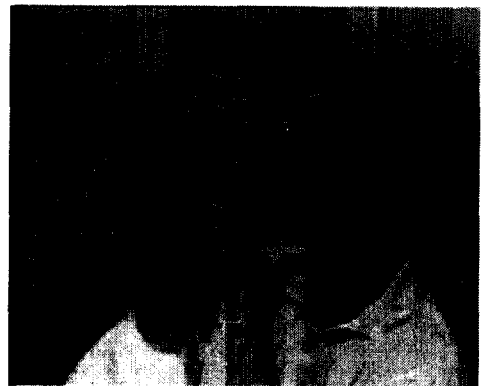
그림 6. 제안한 방법과 H. 263의 PSNR 비교  
 Fig. 6. PSNR comparison of the proposed and BMA



(a)



(b)

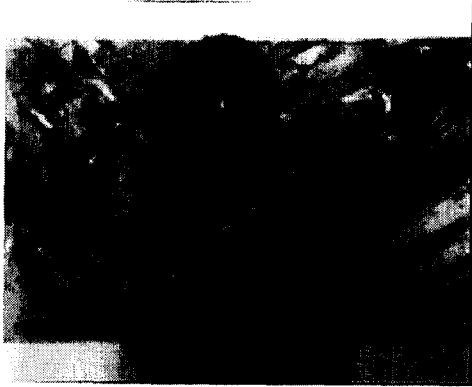


(c)

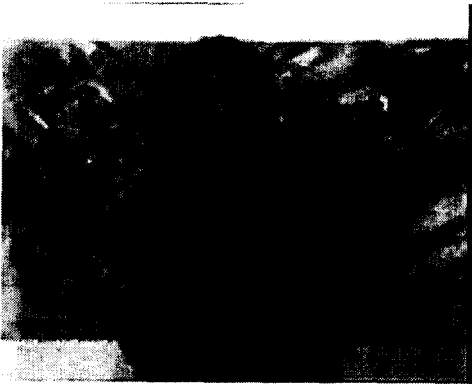
그림 7. 10kbps에서 "Mother & Daughter"(120 번째 프레임)의 복호화된 영상 (a)원영상 (b)H.263 (c)제안한 방법

Fig. 7. Decoded pictures of "Mother & Daughter" (120th frame) at 10 kbps. (a)Original (b)BMA (c)Proposed

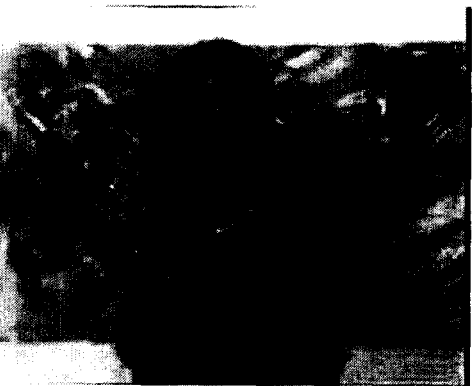




(a)



(b)



(c)

그림 8. 24 kbps에서 "Silent Voice" (180 번째 프레임)의 복호화된 영상 (a) 원영상 (b) H.263 (c) 제안한 방법  
Fig. 8. Decoded pictures of "Silent Voice" (180th frame) at 24 kbps. (a) Original (b) H.263 (c) Proposed



(a)

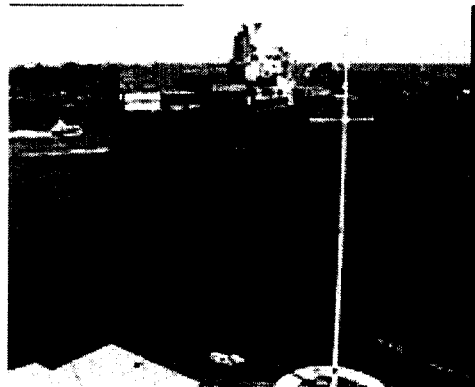


(b)

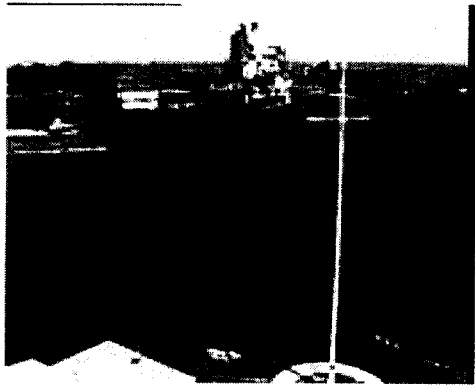


(c)

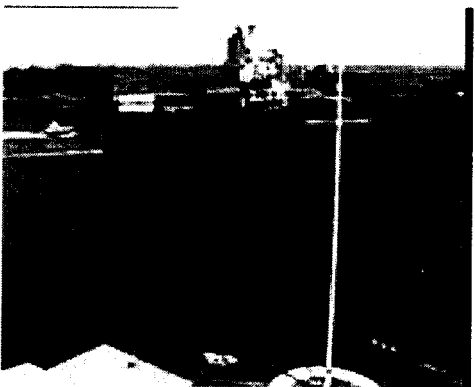
그림 9. 24kbps에서 "Hall Monitor"(102 번째 프레임)의 복호화된 영상 (a) 원영상 (b) H.263 (c) 제안한 방법  
Fig. 9. Decoded pictures of "Hall Monitor"(102th frame) at 24kpbs. (a) Original (b) H.263 (c) Proposed



(a)



(b)



(c)

그림 10. 24kbps에서 “Container Ship”(57 번째 프레임)의 복호화된 영상 (a) 원영상 (b) H.263 (c) 제안한 방법  
 Fig. 10. Decoded pictures of “Container Ship”(57th frame) at 24 kbps. (a)Original (b)H.263 (c)Proposed

그림 7에서부터 그림 10은 각각 H.263과 제안한 방법을 이용하여 얻어진 복호화된 영상들을 보여준 것으로써, 그림 7은 10 kbps로 부호화한 “Mother & Daughter” 영상의 120번째 프레임에 대해, 그림 8은 24 kbps로 부호화한 “Silence Voice”의 180번째 프레임에 대해, 그림 9는 24 kbps로 부호화한 “Hall Monitor” 102번째 프레임에 대해, 그림 10은 24 kbps로 부호화한 “Container Ship” 57번째 프레임에 대해 각각 (a)는 원영상을, (b)는 H.263 으로 복호화된 영상을, (c)는 제안한 방법으로 복호화된 영상을 나타낸다. 각 그림의 (b)에서 보는 바와 같이 H.263을 이용하여 얻어진 복호화된 영상은 심한 블럭 현상과 왜곡이 발생했음을 알 수 있다. 한편, 제안한 방법으로 얻어진 복호화된 영상인 각 그림의 (c)는 삼각형 그물 기반 및 영상 왜핑을 이용한 움직임 보상으로 블럭 현상과 왜곡을 현저하게 줄일 수 있었다. 또한 영상 왜핑 기법에서 사용한 이중 선형 보간법은 부호화기의 순환 필터(Loop Filter)와 같은 지대역 통과 필터 특성을 가지고 있으므로 별도로 부호화기에 순환 필터를 부착하지 않아도 그와 상응하는 효과를 얻을 수 있어, 주관적인 화질이 향상되었다.

이상의 실험 결과로부터, 제안한 부호화 방법이 기존의 블럭을 기반으로 한 H.263 방법에 비해 객관적인 기준에 의한 화질(그림 6)뿐만 아니라 주관적인 화질(그림 7~그림 10)에서도 우수함을 보였다.

## V. 결 론

본 논문은 조 저속 비트율을 갖는 영상 부호화를 위한 2차원 삼각형 그물 기반 움직임 보상 방법을 제안하였다. 2단계 격자점 움직임 추정과 2차원 삼각형 그물 기반 움직임 보상 방법의 결합은 단순한 블럭 기반 움직임 보상을 이용한 기존의 부호화 방법에 비해 예측 효율을 증진시키는데 매우 효과적이었다. 또의 실험 결과, 객관적이나 주관적인 화질 평가에 있어서 제안한 방법이 H.263에 비해 우수함을 보여주었으며, 부호화를 위한 계산 속도도 블럭 정합과 같은 움직임 추정을 이용했기 때문에 H.263과 거의 비슷한 결과를 나타내었다.

이상의 실험 결과로부터, 낮은 전송율을 갖는 통신 환경에서 제안한 부호화 방법이 화질면에서나 속도

면에서 기존의 부호화 방법에 비해 좋은 성능을 나타내기 때문에 초 저속 비트율을 갖는 영상 부호화기의 응용이나 구현에 매우 유용하리라 생각된다.

### 참 고 문 헌

1. C.S. Choi, K. Aizawa, H. Harashima, T. Takebe, "Analysis and Synthesis of Facial Image Sequences in Model-Based Image Coding," *IEEE Trans. Video Technol.*, vol. 4, pp. 257-275, June 1994.
2. H.G. Musmann, M. Hötter and J. Ostermann, "Object-oriented analysis-synthesis coding of moving images," *Signal Processing: Image Communication*, vol. 1, no. 2, pp. 117-138, Oct. 1989.
3. P. Gerken, "Object-Based Analysis-Synthesis Coding of Image Sequences at Very Low Bit Rates," *IEEE Trans. Video Technol.*, vol. 4, pp. 228-235, June 1994.
4. Y. Nakaya and H. Harashima, "An iterative motion estimation method using triangular patches for motion compensation," *SPIE VCIP '91*, vol. 1605, pp. 546-557, Boston, MA, Nov. 1991.
5. Y. Nakaya and H. Harashima, "Motion compensation based on spatial transformations," *IEEE Trans. Video Technol.* vol. 4, pp. 339-356, June 1994.
6. H. Bruesewitz, "Motion compensation with triangles," in *Proc. 3<sup>rd</sup> International Workshop on 64kbit/s Coding of Moving Video*, free session, Sept. 1990.
7. ISO/IEC JTC1/SC29/WG11 N1277, "MPEG-4 video verification model 3.0," July 1996.



김 학 수(Hak Soo Kim) 정회원  
 1963년 2월 21일생  
 1985년 2월:연세대학교 전자공학과(공학사)  
 1992년 2월:연세대학교 대학원 전자공학과(공학석사)  
 1994년 9월~현재:연세대학교 대학원 전자공학과 박사과정

1984년 12월~현재:삼성전자 멀티미디어 연구소 선임 연구원(1995년 9월~휴직중)  
 ※주관심분야: 영상압축, 멀티미디어 통신, 영상신호 처리

이 규 원(Kyu Won Lee) 정회원  
 현재:연세대학교 전자공학과 박사과정  
 한국통신학회 논문지 제22권 7호 참조

박 규 태(Kyu Tac Park) 정회원  
 현재:연세대학교 전자공학과 교수  
 한국통신학회 논문지 제22권 7호 참조