

# 저비트율 동영상 전송을 위한 움직임 기반 동영상 분할

이 범 로<sup>†</sup> · 정 진 현<sup>††</sup>

## 요 약

움직임 기반 동영상 분할은, 동영상 압축 시스템이 보다 효과적으로 동영상을 기술할 수 있도록, 유사한 움직임을 가지는 세그먼트를 정의함으로써, 우수한 동영상 압축을 가능하도록 한다. 본 논문에서는 기존의 K-means 분할 알고리즘을 사용하여 동영상 분할을 구현하였고, 이 과정에서 나타난 문제점들을 개선하기 위하여 수정 퍼지 경쟁 학습 알고리즘을 제안한다. 각 비디오 시퀀스의 움직임은 6개의 매개변수를 사용하는 affine 모델을 사용하였다. 이 affine 모델은 각 프레임의 픽셀들의 움직임을 기술하는 optical flow를 사용하여 계산된다. 이러한 기술들은 화상 회의와 같은, 저속의 전송 선로를 이용하는 저비트율의 비디오 전송에 응용될 수 있다

# The Motion-Based Video Segmentation for Low Bit Rate Transmission

Bum-Ro Lee<sup>†</sup> · Chin-Hyun Chung<sup>††</sup>

## ABSTRACT

The motion-based video segmentation provides a powerful method of video compression, because it defines a region with similar motion, and it makes video compression system to more efficiently describe motion video. In this paper, we propose the Modified Fuzzy Competitive Learning Algorithm (MFCLA) to improve the traditional K-means clustering algorithm to implement the motion-based video segmentation efficiently. The segmented region is described with the affine model, which consists of only six parameters. This affine model was calculated with optical flow, describing the movements of pixels by frames. This method could be applied in the low bit rate video transmission, such as video conferencing system.

## 1. 서 론

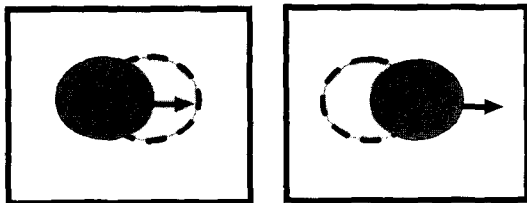
디지털 동영상은 뛰어난 가공성과 잡음에 강한 장점을 가지고 있는 반면, 그 자체의 방대한 데이터 크기로 인하여, 데이터의 전송 및 저장에 있어서 압축이라는 과정이 필수적으로 요구된다는 단점을 가지고 있다. 디지털 동영상의 압축에는 여러 가지 알고리즘이

제시되어 있고, MPEG I, MPEG II, H.261을 비롯한 다양한 산업 표준들 또한 이미 제정되어 있는데, 대부분의 알고리즘이 전체 이미지를 작은 블록으로 나누어 독립적으로 처리하도록 하는 블록 기반(block-based) 동영상 압축 기법을 채택하고 있다. 또한 일반적으로 동영상 압축에서는 나누어진 블록들이 시간의 경과에 따라 어떤 방향으로 이동하였는지를 추정하는 방식으로 전체 프레임 이미지의 움직임 추정(motion estimation)을 근사적으로 수행하게 되는데, 동영상상의 프레임

† 정 회 원 : 광운대학교 대학원 제어계측공학과

†† 정 회 원 : 광운대학교 제어계측공학과 교수  
논문접수 : 1998년 7월 24일, 심사완료 : 1999년 9월 10일

내부 전체의 동적인 움직임을 이러한 단순한 블럭들의 이동으로 표현하게 되면 지역적으로 큰 왜곡이 발생하게 되는 경우가 생긴다. 이러한 왜곡은 (그림 1)에 나타난 것과 같이 동영상 내부의 움직이는 물체에 의해서 가려져 있던 부분이 그 물체의 움직임으로 인해 새롭게 출현하는 부분과 물체의 움직임으로 인해 가려지는 부분에서 크게 나타나는데, 이것을 출몰 문제(occlusion problem)라 한다. 또한 단순한 병진 움직임(translational motion)으로 전체 블럭 내부의 변화를 단일하게 묘사하는 방식은 회전 이동 등의 복잡한 움직임이 묘사되는 부분에서 큰 왜곡을 일으키기도 한다. 따라서 동영상 내부의 움직임에 대한 정확한 표현은 동영상의 프레임간 압축에 있어서 상당히 중요한 부분이다.



(a) 사라지는 배경

(b) 나타나는 배경

(그림 1) 출몰 문제

본 논문에서 제안하는 움직임 추정 기반 동영상 영상 분할은 동영상 프레임 이미지 내의 복잡한 움직임 벡터들을 기반으로 프레임 이미지의 각 부분들을 회전(rotational), 병진(translational), 뒤틀림(shearing) 등의 복잡한 움직임을 묘사할 수 있게 모델링하고, 그 모델을 기반으로 전체 프레임 이미지를 분할하여, 동영상의 내부 움직임을 표현한다. 각 동영상에서 optical flow를 추출하는 과정을 시작으로, 추출된 optical flow를 이용해서 각 동영상의 움직임 모델을 추정하는 과정을 거쳐, 추정된 움직임 모델을 K-means 클러스터링 알고리즘을 사용하여 동영상 내의 계층을 분할하는 과정과 그 후처리 과정을 통해 각 동영상의 분할을 얻게 된다. 동영상의 움직임 모델을 기반으로 하는 표현 방식은 높은 압축율을 가지는 동영상 압축을 가능하게 하여, 저비트율을 가지는 동영상 전송을 하는데 적합하며, 특히 저속망을 이용한 화상 회의 시스템의 동영상 압축의 경우 거의 유사한 환경과 유사한 움직임의 동영상이 전송될 가능성이 높다고 할 수 있으므로, 이 분야에 응용될 경우 우수한 성능을 보일 것으로 예상된다.

## 2. 본 론

동영상은 한 프레임 내부의 픽셀 간의 상관성을 제거하여 압축을 하는 프레임 내(intraframe) 압축과 프레임간의 중복성에 관계되는 프레임 간(interframe) 압축에 의해서 데이터 압축을 하게 된다. 본 논문에서 주로 다루는 프레임 간 압축을 구현하기 위해서 가장 많이 사용되는 것이 움직임 추정 기법(motion estimation method)이라 할 수 있는데 이러한 움직임 추정 기법은 연속되는 두 프레임간의 변화를 각 픽셀의 이동 방향과 크기를 나타내는 움직임 벡터로써 나타내는 방법이다. 일반적으로 움직임벡터를 구하기 위해서 block matching algorithm(BMA), optical flow method(OFM), pel-recursive method(PRM) 등의 기법들이 많이 사용되는데, 본 논문에서는 OFM을 사용하여 구현된 optical flow를 움직임 벡터로 사용하였다. 이러한 움직임 벡터 정보만을 가지고 동영상 내의 복잡한 움직임들을 표현하기에 한계가 있으므로, 계산된 움직임 벡터를 이용하여 6개의 매개변수로 동영상의 3D 움직임을 기술하는 affine model을 얻어 내어 동영상 움직임을 기술하는데 이용하였다. 각 affine model은 클러스터링 과정을 거쳐서 클러스터를 형성하게 되고 각 클러스터의 센터값은 전체 동영상의 움직임을 기술하는 대표 움직임 모델로 사용할 수 있다. 따라서 전체 동영상의 각 픽셀들이 가지는 움직임들은 각각의 픽셀이 속하는 클러스터의 대표 움직임 모델로 대표되어 높은 압축율을 가지게 되며, 특히 움직임이 없는 배경 등에서는 상당한 압축 효과를 가져오게 된다.

### 2.1 Optical flow<sup>[4]</sup>

Optical flow는 원래 비전 시스템에서 움직임을 추정하는 방법의 하나로 사용되어 오던 것으로, 동영상의 밝기의 변화와 관련된 2-D velocity field를 추정한다. 이것은 실제 객체의 움직임과 정확히 일치하는 것은 아니지만 근사적으로 3-D motion field를 나타낼 때 주로 사용된다. 주어진 두 프레임에 대해서 영상 내부의 각 화소값과 그에 대한 움직임 벡터의 관계는 다음 식과 같이 나타낼 수 있다.

$$S(x - v_x(x, y), y - v_y(x, y), t + 1) = S(x, y, t) \quad (1)$$

식 (1)에서  $S(x, y, t)$ 가 화소값이고  $v_x(x, y)$ ,  $v_y(x,$

y)를 그 화소의 움직임 벡터라 하면, 식 (1)은 동영상의 화소값 변화가, 그 화소가 움직임 벡터만큼 이동함에 기인한다라는 것을 의미한다. 이것은 바꾸어 말하면 프레임 내의 한 화소가 그 화소의 시간에 따른 움직임 경로에 대해서 그 화소값을 유지한다는 것인데 식 (2)은 그와같은 개념을 나타내 준다.

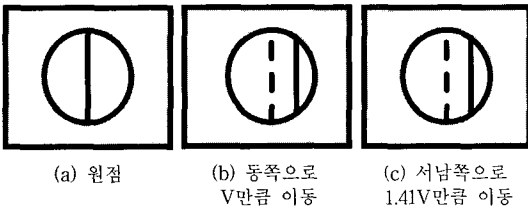
$$\frac{dS(x, y, t)}{dt} = 0 \quad (2)$$

다시 식 (2)을 Taylor's expansion을 이용하여 전개한 후 고차항을 버리고 초기 몇 개의 항만을 선택하여 정리하면, 아래 식 (3)의 형태로 표현될 수 있다.

$$\begin{aligned} \frac{\partial S(x, y, t)}{\partial x} v_x(x, y, t) + \frac{\partial S(x, y, t)}{\partial y} v_y(x, y, t) \\ + \frac{\partial S(x, y, t)}{\partial t} = 0 \quad (3) \end{aligned}$$

(단  $v_x(x, y, t) = \frac{dx}{dt}$ ,  $v_y(x, y, t) = \frac{dy}{dt}$ )

식 (3)를 일반적으로 optical flow constraint 또는 optical flow equation이라 하며, 식 (3) 하나만으로는 2개의 변수 ( $v_x, v_y$ )를 가지는 각 화소의 optical flow의 확정적인 해를 구할 수 없기 때문에 발생하는 문제를 불확정적인 움직임 기술의 문제 (ill-posed problem)이라 한다.



(그림 2) 불확정적인 움직임 기술의 문제

(그림 2)는 이와같은 문제점들의 예를 나타내고 있는데, (그림 2)(a)에 나타난 바와 같이 작은 원 모양의 틈을 통해 물체를 관찰할 경우 동쪽으로 V 만큼의 속도로 움직이는 막대를 나타낸 (그림 2)(b)와 남동쪽으로 1.41V만큼의 속도로 움직이는 막대를 나타낸 그림 (c)는 같은 optical flow를 나타낸다는 것이다. 이러한 문제들의 해결을 위하여, 추가적인 가정을 이용해서 확정된 optical flow를 구하게 된다. 본 논문에서는 특정한 블럭 내의 움직임 벡터들이 그 블럭 내부에서 변하지 않는다

는 가정을 통하여 optical flow를 계산하는 Lucas and Kanade의 알고리즘을 사용한다. 특정 영역 R 내부의 움직임 벡터를  $v_x, v_y$ 이라 가정하면, 그 영역에서 실제 에러값을 다음 식 (4)와 같이 나타낼 수 있다.

$$\begin{aligned} E = \sum_R \left( \frac{\partial S(x, y, t)}{\partial x} v_x(x, y, t) \right. \\ \left. + \frac{\partial S(x, y, t)}{\partial y} v_y(x, y, t) + \frac{\partial S(x, y, t)}{\partial t} \right)^2 \quad (4) \end{aligned}$$

식 (4)를  $v_x, v_y$ 에 대하여 각각 미분하여 정리하면 다음과 같다.

$$\begin{aligned} \sum_R \left( \frac{\partial S(x, y, t)}{\partial x} v_x(x, y, t) + \frac{\partial S(x, y, t)}{\partial y} v_y(x, y, t) \right. \\ \left. + \frac{\partial S(x, y, t)}{\partial t} \right) \frac{\partial S(x, y, t)}{\partial x} = 0 \quad (5) \end{aligned}$$

$$\begin{aligned} \sum_R \left( \frac{\partial S(x, y, t)}{\partial x} v_x(x, y, t) + \frac{\partial S(x, y, t)}{\partial y} v_y(x, y, t) \right. \\ \left. + \frac{\partial S(x, y, t)}{\partial t} \right) \frac{\partial S(x, y, t)}{\partial y} = 0 \quad (6) \end{aligned}$$

위 식 (5), 식 (6)을  $v_x, v_y$ 에 대해서 각각 정리하면 다음과 같은 식을 얻는다.

$$\begin{bmatrix} \sum_R E_x^2 & \sum_R E_x E_y \\ \sum_R E_x E_y & \sum_R E_y^2 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} \sum_R -E_x E_t \\ \sum_R -E_y E_t \end{bmatrix} \quad (7)$$

위 식에서  $E_x, E_y$ 와  $E_t$ 는 x, y 지점에서의 화소값의 x, y, t 각각에 대한 편미분값이며, 식 (7)을 이용하여 각 영역 R의 근사적인 optical flow를 구하게 된다. 영역 R은 9×9 크기의 Gaussian 윈도우를 이용하여 선택되었고 이렇게 얻어진 optical flow는 움직임 필드를 발생시키는 각 화소의 움직임을 추정할 수 있도록 한다.

## 2.2 움직임 모델 (Motion model) <sup>[1][2]</sup>

동영상의 optical flow가 추정되면, 유사한 움직임을 가지는 화소들을 분류하여 클러스터를 만들어 낼 수 있다. 그러나 BMA와 같은, 단순히 속도 방향으로의 병진 움직임만을 표현하는 모델을 이용한 화소의 그룹화는, 각 객체를 지나치게 작은 조각으로 묘사할 가능성이 있다. 예를 들어 회전운동을 하는 객체를 병렬 움직임만을 기반으로 그룹화 한다면 서로 다른 속도를 가지는 많은 작은 영역들이 생겨날 것이며, 이것은 동영상 압축에 있어서 상당한 비효율을 초래하게 된다.

본 논문에서는 각 객체의 움직임을 affine model을 사용하여 묘사했는데, affine model은 객체의 병진, 회전 확대, 늘림 등을 포함한 다양한 움직임을 표현할 수 있다. affine model은 다음 방정식에 의해서 정의된다.

$$\begin{aligned} v_x(x, y) &= a_{x0} + a_{xx}x + a_{xy}y \\ v_y(x, y) &= a_{y0} + a_{yx}x + a_{yy}y \end{aligned} \tag{8}$$

윗식에서 움직임을 정의하는 6개의 변수  $a_{x0}, a_{xx}, a_{xy}, a_{y0}, a_{yx}, a_{yy}$ 가 존재하는데 이것을 t식 (9)와 같이 6차원의 벡터로서 정리한 이후에 와 x와 y성분으로 나누어 표현하면 식 (10)과 같고  $\phi^T$ 를 식 (11)과 같이 정의하면 식 (8)은 식 (12)와 같이 표현된다.

$$a_i^t = [ a_{x0i} \ a_{xxi} \ a_{xyi} \ a_{y0i} \ a_{yxi} \ a_{yyi} ] \tag{9}$$

$$a_{xi}^T = [ a_{x0i} \ a_{xxi} \ a_{xyi} ] \ a_{yi}^T = [ a_{y0i} \ a_{yxi} \ a_{yyi} ] \tag{10}$$

$$\phi^T = [ 1 \ x \ y ] \tag{11}$$

$$\begin{aligned} v_x(x, y) &= \phi^T a_{xi} \\ v_y(x, y) &= \phi^T a_{yi} \end{aligned} \tag{12}$$

영역 R에 대한 affine 움직임 모델의 최소 자승근은 다음 식 (13)와 같다.

$$[ a_{yi} \ a_{xi} ] = [ \sum_{R_i} \phi \phi^T ]^{-1} \sum_{R_i} [ v_y(x, y) \ v_x(x, y) ] \tag{13}$$

계산에 이용되는 영역은 안정된 움직임 추정을 할 수 있을 만큼 충분히 넓어야 하며 움직임이 잡음에 의해 손상되었을 경우 특히 충분한 영역을 확보하여야 한다. 실험적으로 검증된 가장 효율적인 영역은 21X21의 사각 영역이다. 계산된 affine model을 사용하여, 각 프레임 내에 객체들의 움직임을 6개의 변수를 통해 묘사할 수 있으며, 따라서 전체 동영상의 비트율이 현저하게 감소하게 된다.

### 2.3 K-means 움직임 모델 클러스터링<sup>[1]</sup>

언어진 영역의 affine model 중에는 대부분의 경우에 많은 유사한 변수를 가진 모델들을 발견할 수 있다. 유사한 affine model을 가지는 영역은 유사한 움직임을 가진다는 것을 의미하므로, 이러한 모델들은 클러스터링 알고리즘을 이용하여, 서로 병합되고 병합된

움직임 모델들은 하나의 대표값 움직임 모델로 대표되게 된다. 일반적으로 이 과정의 클러스터링 알고리즘으로서 K-means 클러스터링 알고리즘을 사용하게 되는데 이 알고리즘에서는 초기에 각 클러스터의 센터값들을 계산된 움직임 모델값으로부터 선택하고, 거리값에 의해서 각 클러스터를 분리시킨다. 각 모델들간의 거리는 다음 식 (14)과 같이 나타낼 수 있다.

$$D_m(a_1, a_2) = [ (a_1 - a_2)^T (a_1 - a_2) ]^{1/2} \tag{14}$$

초기에 각 클러스터의 센터값들은 적당한 거리를 두고 임의로 세팅한 다음 반복 계산을 통해서 각 센터값들을 갱신해 나간다. 주어진 한 움직임 모델에 대해서 센터와의 가장 거리가 가까운 클러스터를 결정한 다음 그 클러스터의 센터값을 평균값만큼 주어진 움직임 모델 방향으로 갱신시키는 시키는 방법으로 각 클러스터를 분류해 나간다. 본 논문의 실험에서는 초기에 임의로 10개의 클러스터를 설정하고 실험을 하였다.

### 2.4 수정 퍼지 경쟁 학습 알고리즘<sup>[6]</sup>

K-mean 클러스터링 알고리즘을 이용하여 각 움직임 모델을 클러스터링 하기 위해서는 미리 클러스터의 개수를 설정한 이후에 분류 작업을 수행하여야 한다. 클러스터의 개수는 각 동영상의 움직임의 특성상 상당히 넓은 범위를 가지고 변할 것이므로 이를 실험적으로 일일이 결정한다는 것이 매우 번거로운 작업일 것이다. 따라서 본 논문에서는 이러한 클러스터의 개수를 클러스터링 과정 중에 결정할 수 있도록 수정 퍼지 경쟁 학습 알고리즘<sup>[6]</sup>(Modified Fuzzy Competitive Learning Algorithm : MFCLA)을 적용하도록 하였다. 수정 퍼지 경쟁 학습 알고리즘은 최소 소속도라고 하는 각 클러스터가 허용하는 최대 왜곡값의 역수를 정의하고 적용되는 움직임 모델 벡터가 모든 클러스터의 최소 소속도를 벗어날 경우에는 그 움직임 모델 벡터값을 센터값으로 하는 새로운 클러스터를 설정하는 방법으로 생성되는 대표값 움직임 모델의 개수를 결정하도록 하였다. 또한 보다 최적화된 클러스터링을 위하여 하나의 벡터 당 한 클러스터만을 갱신시키는 K-means 알고리즘과 달리 존재하는 모든 클러스터를 그들이 가지는 소속도만큼 갱신시킴으로써 보다 최적화된 클러스터링을 가능하게 한다. 다음은 적용된 수정 퍼지 경쟁 학습 알고리즘이다.

- (a) 각 클러스터 내에 존재할 수 있는 최소 소속도 ( $T_{min}$ )을 초기화하고, 적용되는 움직임 모델 벡터의 갯수에 대한 카운터 *counter* 를 0으로 한다.
- (b) 첫번째 움직임 모델 벡터를 입력 받아 첫번째 클러스터의 센터값으로 초기화한다.

$$y_0 = x_0$$

- (c) 적용되는 움직임 모델 벡터에 대해서 각 센터값과의 소속도를 계산한다. 이때 소속도는 왜곡값의 역수를 정규화하여 나타낸다.

$$m_i(t) = \|y_i(t) - x(t)\|^{-1} / \sum_{j=0}^N \|y_j(t) - x(t)\|^{-1}$$

$$(i = 1, 2, \dots, counter)$$

- (d) (c)에서 계산된 소속도가  $T_{min}$ 보다 작은 경우에는 입력받은 움직임 모델 벡터를 센터값으로 새로운 클러스터를 만들고, 클러스터의 수를 나타내는 *counter*를 1만큼 증가시키고  $T_{min}$ 보다 크거나 같을 경우에는 다음 규칙에 의해 모든 센터값을 갱신한다.

$$y_i(t+1) = y_i(t) + m_i(t)[x(t) - y_i(t)]/c_i(t)$$

$$(i = 1, 2, \dots, \tilde{counter})$$

$$c_i(t+1) = c_i(t) + m_i(t)$$

$$(i = 1, 2, \dots, counter)$$

- (e)  $t = t + 1$ 로 놓고 (c)를 반복한다.

### 2.5 영역 분할 (Region classification)<sup>[2]</sup>

동영상 내에 유사한 움직임을 가지는 한 영역이 어떤 대표 움직임 모델값을 가지는지 테스트하고, 테스트 결과 가장 근사한 모델값을 그 영역의 대표 움직임 모델값으로 할당하는 작업을 통해 각 영역 분할한다. 각 대표 움직임 모델값과 영역내의 움직임 모델과의 유사성을 판별하기 위해서 다음과 같은 왜곡 함수를 사용한다.

$$G(i(x, y)) = \sum_{x,y} [v(x, y) - v_{a_i}(x, y)]^2 \quad (15)$$

식 (15)에서  $i(x, y)$ 는  $x, y$ 에 할당된 움직임 모델이고,  $v(x, y)$ 는 추정된 움직임 필드이며,  $v_{a_i}(x, y)$ 는  $i$ 번째

대표 움직임 모델에 의한 affine 움직임 필드이다. 그러나  $G(i(x, y))$ 가 0일 수 있는 경우는 특정 영역의 움직임이 대표 움직임 모델에 의해서 완전하게 묘사되는 경우인데, 대부분의 경우 잡음등의 외부적인 요인 등에 의해서  $G(i(x, y))$ 는 0이 될 수 없다. 본 논문에서는 아래 식 (16)를 이용하여 왜곡을 최소화하는 대표 움직임 모델을 그 지역을 분할하는 움직임 모델로 지정한다.

$$i_0(x, y) = \arg \min [v(x, y) - v_{a_i}(x, y)]^2 \quad (16)$$

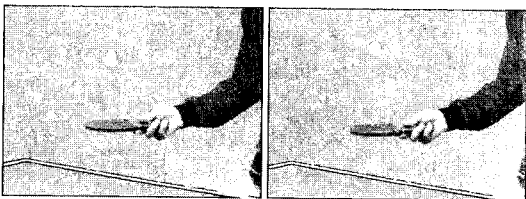
윗 식에서  $i_0(x, y)$ 는 최소 왜곡 대표 움직임 모델값이다.

어떤 움직임을 가진 영역은 대표 움직임 모델을 이용하여 쉽게 표현되지 못해서 여전히 분류되지 못하고 남게되는 경우가 발생하기도 한다. 이 경우 분류되지 못한 영역만을 모아서 다시 새로운 대표 움직임 모델값들로 표현되는 영역을 생성하는 과정을 반복한다.

### 3. 실험 및 결과

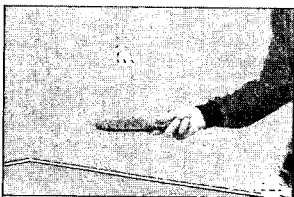
실험에 사용된 비디오 프레임은 (그림 3)과 (그림 4)에 나타난 360×240의 Ping-pong 비디오 프레임이며, Windows NT를 OS로 하는 Pentium급의 PC상에서 Visual C++ 5.0을 이용하여 프로그래밍 되었다. 두 비디오 프레임은 256 gray level을 가지며, Ping-pong 비디오 프레임은 사람의 오른팔과 탁구공, 그리고 라켓 등의 부분에서 큰 움직임이, 왼팔에 약간의 작은 움직임이 보여지며, 그 이외의 부분에서는 움직임이 거의 나타나지 않는 특징을 보인다. 이 두 프레임 사이에서 발생하는 optical flow를 구하기 위해서, 9×9의 윈도우를 사용하여 식 (7)을 적용하였으며, 21×21의 윈도우를 사용하여, 식 (13)를 적용하였다. 클러스터링 알고리즘은 언급한 바와 같이 K-means 알고리즘과 수정 퍼지 경쟁 학습 알고리즘을 사용하여 두 알고리즘을 비교하였으며, K-means 알고리즘의 경우에는 클러스터의 수는 10개로 고정하였고, 각 클러스터의 초기값은 Visual C++에서 제공하는 rand()함수에서 발생하는 의사 난수를 10000으로 나누어 사용하였다. 아래 그림 5는 pong-pong 비디오의 두 프레임 사이에서 구해진 optical flow를 첫 번째 프레임 위에 나타내었다. 큰 움직임이 나타나는 부분에 크기가 큰 움직임 벡터가 발생함을 나타내 주고 있으며, 이러한 optical flow의 계

산에 있어서 적용되는 윈도우의 크기가 클수록 정확한 optical flow가 생성되나 계산량이 급격히 늘어나므로, 적절한 크기의 윈도우 크기를 설정해야 한다. 계산된 움직임 벡터를 바탕으로 각 화소의 affine model을 구하고, 그 결과에 K-mean 클러스터링 기법과 수정 퍼지 경쟁 학습 알고리즘을 적용해서 (그림 6), (그림 7)과 같은 움직임을 기반으로 하는 분할을 얻었다. 두 그림에서 보여주는 것과 같이 수정 퍼지 경쟁 학습 알고리즘은 대표 움직임 모델값의 개수를 클러스터링 과정에서 결정하고 보다 최적화된 학습을 하게 되므로 탁구공 주변과 팔 주변의 각 움직임 분할 내부가 다시 작은 분할로 나뉘어지는 현상을 현저하게 줄여 주었다. (그림 6)과 (그림 7)에서 각 분할을 구별해 주는 색깔은 분할을 나타내는 0부터 9까지의 인덱스에 25를 곱한 값이므로, 색을 밝기와 움직임의 특성과는 무관하며, 많은 부분이 움직임이 없는 배경을 나타내는 검은 색으로 표시되어 있어, 높은 압축율의 비디오 압축이 가능함을 암시하고 있다.

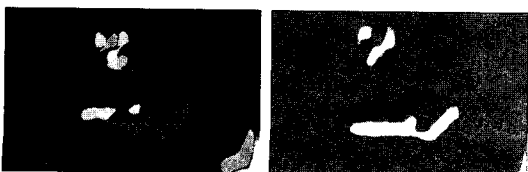


(그림 3) 프레임 1

(그림 4) 프레임 2



(그림 5) Optical flow



(그림 6) 분할된 계층 (K-means)

(그림 7) 분할된 계층 (MFCLA)

#### 4. 결론

본 논문에서 제안된 수정 퍼지 경쟁 학습 알고리즘은 기존의 K-means 클러스터링 알고리즘을 동영상 압축 분야에 적용할 경우 발생할 수 있는 문제점들을 해결함으로써 중간 수준 비전 알고리즘의 동영상 압축 응용을 용이하게 한다. 또한 실제적인 클러스터링 과정에 있어서의 최적화 측면에서도 기존 알고리즘의 성능을 개선할 수 있다. 동영상 압축에 있어서 중간 수준의 비전 알고리즘의 적용은 차세대 영상 압축 기술이라 할 수 있을 만큼 여러 가지 측면에서 유용성을 가진다. 본 논문에 구현된 움직임 기반의 비디오 분할에서 나타난 바와 같이 높은 압축율의 비디오 압축을 가능하게 하는 측면 또한 그 유용성의 일부라 할 수 있을 것이다. 특히 저비트율의 비디오 전송을 통한 화상 회의에서의 응용의 경우 거의 대부분 움직임이 없는 배경에 작은 움직임을 가지는 객체들로 구성된 프레임으로 구성될 가능성이 크고, 장면이 전환되는 경우가 드물기 때문에 본 논문에서 구현된 알고리즘의 유용성은 더욱 크다고 할 수 있다. 또한 실제적인 화상 회의 시스템의 동영상 코덱에는 실제 동영상 프레임들은 서브샘플링(subsampling)해서 수신측에 전달하는 것이 보통인 경우이며, 본 논문에서 다루어진 움직임 분할 기법을 적용하여, 송신시 서브 샘플링된 프레임들을 수신측에서 다시 재구성하는 접근도 가능한 응용분야라 할 수 있을 것이다. 추후에는 실제로 이러한 알고리즘을 실시간에서 동작하도록 구현하고 인터넷과 같이 실제적인 저속망에 적용하는 것에 대한 연구가 진행 될 것이다.

#### 참 고 문 헌

- [1] J. Y. A. Wang, E. H. Adelson, "Layered representation for image sequence coding," In Proc. *IEEE ICASSP*, Vol.5, pp221-224, April 1993.
- [2] J. Y. A. Wang, E. H. Adelson, "Layered representation for motion analysis," In Proc. *IEEE Conf. Comput. Vision Pattern Recog.*, pp361-366, June 1993.
- [3] Jose L. Marroquin, Federico Girosi, "Some Extensions of the K-Means Algorithm for Image Segmentation and Pattern Classification," MIT A.I.

Memo No.1390, C.B.C.L Paper No.079, January, 1993

- [4] A. Murat Tekalp, "Digital Video Processing," Prentice Hall, 1995
- [5] M. Ibrahim Sezan, Reginald L. Lagendijk, "Motion Analysis And Image Sequence Processing," KLUWER ACADEMIC PUBLISHERS, 1993
- [6] 이범로, 정진현, "수정 퍼지 경쟁 학습 알고리즘을 이용한 이미지 코딩 응용", 정보처리학회 논문지 제5권 제7호, 1998 7월
- [7] Adrian G. Bors, Ioannis Pitas, "Optical Flow Estimation and Moving Object Segmentation Based on Median Radial Basis Function Network," *IEEE Trans. Image Processing*, Vol.7, No.5, May 1998.
- [8] 변재응, "Optical flow의 sequence coding의 적용에 관한 연구", 광운대학교, 1996



### 이 범 로

e-mail : redcom@channeli.net

1993년 광운대학교 제어계측공학과 졸업(공학사)

1995년 광운대학교 대학원 제어계측공학과(공학석사)

1995년~1996년 현대전자 S/W 연구소 연구원

1996년~1998년 현대정보기술 인터넷 사업본부 선임

1998년~현재 LG 인터넷 시스템 개발팀 대리

관심분야 : 분산 멀티미디어, 영상 압축, 뉴로 퍼지 시스템, 인터넷 응용



### 정 진 현

e-mail : chung@daisy.kwangwoon.ac.kr

1981년 연세대학교 전기공학과(공학사)

1983년 연세대학교 대학원 전기공학과(공학석사)

1990년 Rensselaer Polytechnic Institute(Ph. D)

1991년~현재 광운대학교 제어계측공학과 부교수, 신기술연구소 연구원

관심분야 : fuzzy control, neural network, MPEG II, stochastic control, CIM