

論文2000-37SP-11-1

윤곽선 재조정을 통한 의미 있는 객체 추적 알고리즘 (A Semantic Video Object Tracking Algorithm Using Contour Refinement)

任 貞 恩*, 李 在 淵*, 羅 鍾 範*

(Jungeun Lim, Jaeyoun Yi, and Jong Beom Ra)

요 약

이 논문은 동영상에서 의미 있는 객체를 추적하기 위해, 첫 번째 프레임에서 사용자가 관심 대상인 객체를 정의하고, 그 다음 프레임부터 자동으로 그 객체를 추적하는 반자동 기법을 제안한다. 제안한 객체 추적 알고리즘은 객체 경계 투영, 불확실 영역 추출, 경계 재조정 단계 등 모두 세 단계로 구성되며, 첫 단계에서는 움직임 추정을 통해 이전 프레임에서 현재 프레임으로 객체를 투영하고, 두 번째 단계는 투영한 결과를 이용하여 윤곽선 부근에서 투영이 불확실한 영역을 MC 오류 및 색채 유사성 검사를 거쳐 추출하며, 마지막으로 투영이 불확실한 영역을 재조정함으로써 정확한 객체의 경계를 찾는다. 모의 실험을 통해 제안한 알고리즘이 기존의 반자동 알고리즘에 비해 다양한 영상에서 대해 만족할 만한 결과를 보임을 확인하였다.

Abstract

This paper describes an algorithm for semantic video object tracking using semi-automatic method. In the semi-automatic method, a user specifies an object of interest at the first frame and then the specified object is to be tracked in the remaining frames. The proposed algorithm consists of three steps: object boundary projection, uncertain area extraction, and boundary refinement. The object boundary is projected from the previous frame to the current frame using the motion estimation. And uncertain areas are extracted via two modules: MC error-test and color similarity test. Then, from extracted uncertain areas, the exact object boundary is obtained by boundary refinement. The simulation results show that the proposed video object extraction method provides efficient tracking results for various video sequences compared to the previous methods.

I. 서 론

MPEG-4 동영상 압축 표준은 VOP(Video Object Pane)의 새로운 개념을 도입하였다^[1, 2]. MPEG-1, 2가 정사각형 블록 단위의 처리에 기반한 반면, MPEG-4는 각 프레임을 객체(video object)들로 분리하여 객체 단위로 처리하는 방식을 택하고 있다. MPEG-4 동영상 부호화를 위해서는 전처리 과정으로

동영상 내의 의미 있는 객체를 분리해내는 영상 분할 기법이 필요하다.

지난 몇 년 동안 자동으로 객체를 추출하는 방법들이 많이 연구되어 왔다. 자동 객체 분할 기법들로는 부호화 관점에서 균질한(homogeneous) 영역을 추출하거나^[3, 4], 영상의 공간(spatial) 정보와 시간(temporal) 정보를 함께 이용하여 분할하는 방식^[5] 등이 있다. 그러나, 두 방식 모두 여러 색상, 질감, 움직임 정보를 함께 갖고 있는 동영상 객체를 자동으로 추출하기에는 아직 어려움이 많으며 만족할 만한 결과를 얻지 못하고 있다.

따라서 최근에는 자동 분할 방식의 문제점을 극복하기 위한 반자동 분할 방식을 이용한 연구가 활발하

* 正會員, 韓國科學技術院 電子電算學科
(Department of Electrical Engineering & Computer Science, KAIST)
接受日: 1999年9月15日, 수정완료일: 2000年11월17일

다^[6,7]. 이 방법에서는 첫 프레임의 영상 분할은 사람의 지식에 기반을 하여 객체를 추출하고, 그 다음 프레임부터는 사람의 도움 없이 이전 결과를 이용하여 자동으로 객체를 추적한다. 그리고 간혹 추적 결과가 어긋났다고 사람이 판단하는 경우는 잘못된 프레임에서 다시 사람의 지식에 기반하여 객체 분할을 하고, 그 다음 프레임부터 다시 자동으로 객체를 추적하는 방식이다. 이와 같은 접근 방식에 의해서 의미 있는 객체를 사람의 지식을 바탕으로 정의하고, 컴퓨터를 이용하여 적은 노력으로 의미 있는 객체를 추적할 수 있다. Gu는 반자동 방법으로 이전 프레임의 객체를 현재 프레임에 투영하고 투영된 윤곽선을 재조정하여 현재 프레임의 객체를 추적하는 방법을 제안하였다^[7]. 그러나, 이 방법에서는 투영된 객체 윤곽선을 중심으로 일정한 두께 내부만을 불확실 영역으로 선택하므로, 객체의 실제 윤곽선이 불확실 영역 내에 존재하지 않는다면 객체의 윤곽선을 찾을 수 없다. 또, 객체의 윤곽선이 불확실 영역 내에 들어오도록 하기 위해서, 투영된 윤곽선을 중심으로 불확실 영역을 넓게 선택하면, 다른 가장자리로 객체의 윤곽선이 수렴하는 단점이 있다.

이 논문에서는 이전 프레임의 객체를 현재 프레임에 투영한 결과를 고려하여 객체의 윤곽선이 존재할 것으로 예상되는 불확실 영역을 추출하는 방법을 제안함으로써, 이전 방법과 같이 필요 이상으로 불확실 영역이 추출되지 않으면서 객체의 실제 윤곽선이 불확실 영역에 존재하도록 한다. 또, 추출한 영역을 바탕으로 정확한 객체의 윤곽선을 구하기 위해 여러 단계로 윤곽선을 재조정하는 방법을 제안하였다.

이 논문의 구성은 다음과 같다. II장에서 제안하는 객체 추적 알고리즘에 대해 설명하고, III장에서 실험 결과를 보이고, IV장에서 결론을 맺는다.

II. 객체 추적 알고리즘

그림 1은 제안하는 객체 추적 알고리즘의 전체구조를 보여 주고 있다. 제안한 구조는 3단계로 구성된다. 객체 투영 단계에서는 이전 프레임의 결과 O_{n-1} 의 현재 프레임 쪽으로 움직임을 추정하고 현재 프레임에 투영하여 대략적인 현재 프레임의 객체를 얻는다. 두번째 단계는 투영된 객체의 윤곽선을 이용하여 투영의 결과가 불분명한 영역을 추출한다. 이 과정에서

투영의 결과가 불분명한 영역은 움직임 보상 오차(motion estimation error)와 투영된 객체의 윤곽선에 인접한 배경과 객체의 색상 비교의 두 과정을 통해 추출된다. 그리고, 마지막 단계에서 투영이 불분명한 영역과 그 부근을 다시 재조정함으로써 정확한 객체의 윤곽선을 구한다. 이 윤곽선 재조정은 다시 3단계로 이루어져 있다.

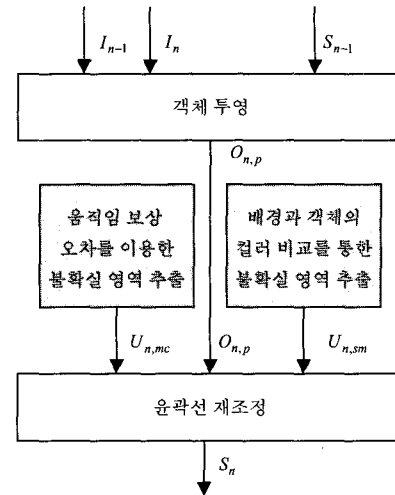


그림 1. 제안한 알고리즘의 전체 구조

Fig. 1. The scheme of the proposed algorithm.

1. 객체 투영

객체 투영(temporal projection)은 이전 프레임에서 현재 프레임 쪽으로 객체의 움직임을 추정하여 현재 프레임에서 객체를 예측하는 과정이다. 여기서 객체의 움직임은 매개 변수 움직임 모델(parametric motion model)을 이용하였다.

이 논문에서는 6개의 매개변수를 갖는 affine 움직임 모델을 이용한다.

$$\begin{aligned} x &= a_1 + a_2x' + a_3y' \\ y &= a_4 + a_5x' + a_6y' \quad (x', y') \in O_{n-1} \end{aligned} \quad (1)$$

Affine 움직임 모델은 식 (1)과 같다. 여기에서 (x, y) 는 현재 프레임의 화소의 위치, (x', y') 는 이전 프레임의 화소의 위치를 나타낸다. 그리고 O_{n-1} 는 이전 프레임의 추적 결과 S_{n-1} 에서 객체를 나타내며, (x', y') 는 O_{n-1} 를 구성하고 있는 화소이다. 또, $a_i, 1 \leq i \leq 6$ 는 affine 움직임 모델의 계수를 의미한다. 이 논문에서 움직임 모델 계수는 OFE(optical flow equation)의 제곱합이 최소화 하도록 LMS(least

mean square) 방법을 이용하여 구하였다. 그리고, 이전 프레임의 객체를 구성하고 있는 화소들을 affine 움직임 모델을 이용해서 현재 프레임으로 투영하여 $O_{n,b}$ 를 구한다. 그림 2는 객체 투영 과정을 보여주고 있다. 여기서 움직임 모델의 매개변수는 실수이므로, 실수 화소 (x, y) 를 둘러싼 4 정수 화소를 모두 객체로 결정한다.

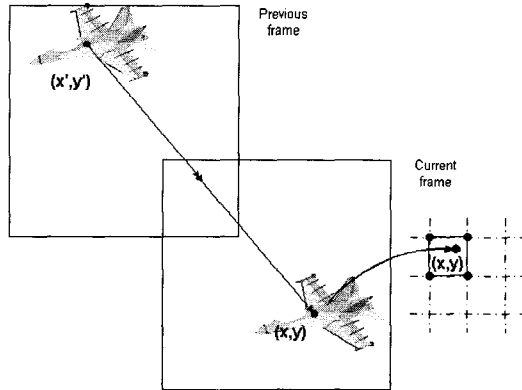


그림 2. 객체 투영
Fig. 2. Object temporal projection.

2. 영역 확장

영상 분할에서 영상은 여러 영역(region)들로 분할된다. 그리고, 분할된 각 영역은 서로 다른 레이블(label)값으로 표시하여 다른 영역으로 구분 지을 수 있다. 또, 영역이 결정되지 않은 부분을 '불확실 영역(uncertain region)'이라고 한다. 그림 3(a)는 불확실 영역과 불확실 영역에 접하고 있는 영역 I, II를 나타내고 있다. 불확실 영역 내 화소들은 영역 I, II를 불확실 영역으로 확장함으로써 영역 I 또는 II로 결정할 수 있다. 이 때, 불확실 영역 내 화소들은 밝기나 색상 그리고 화소의 위치에 따라서 영역 I 또는 II로 결정하며, 이 과정은 기존의 watershed 알고리즘을 응용하여 수행된다.

Watershed 알고리즘은 경사 영상(gradient image) 화소의 밝기 값을 고도 정보로 생각하고, 경사 영상에서 원 영상의 국부 최소치(local minimum)에 연관된 영역을 분할해 내는 알고리즘이다^[9]. 그러나, 경사 영상에서 모든 국부 최소치에 대해 영역을 분할하게 되면, watershed 알고리즘의 결과는 영역 수가 너무 많은 과분할(over-segmentation)이 되는 경향이 있다. 그러므로, 과분할을 막기 위해 국부 최소치들 중에서 특정 조건을 만족하는 최소치들을 표시자(marker)로

결정하고, 이 표시자에 대응하는 영역을 추출하는 방법이 많이 사용된다. 또, Watershed 알고리즘은 침수 시뮬레이션(immersion simulation)에 의한 방법이 가장 효과적이라고 알려져 있다^[9]. 침수 시뮬레이션에서는 대상이 되는 화소와 표시자와의 밝기 값의 차가 작고, 화소와 표시자의 거리가 가까울수록 더 빨리 영역 결정이 이루어진다. 그리고, 영상의 경사도 값이 아니라 영상의 신호값 자체에 적용할 수 있는 watershed 알고리즘이 소개되었다^[10].

영역 확장에서는 watershed 알고리즘을 변형하여 그림 3(b)에서 설명하는 바와 같이 불확실 영역을 둘러싸고 있는 모든 화소들을 표시자로 고려한다^[7]. 그리고, 모든 표시자들로부터 불확실 영역 내 화소에 대해 watershed 알고리즘을 적용한다. 그리고, 영역 I에 포함되는 표시자로부터 확장된 불확실 영역 내의 모든 화소들은 영역 I으로 결정하고, 같은 방법으로 영역 II에 포함되는 표시자로부터 확장된 불확실 영역 내의 모든 화소는 영역 II로 결정한다. 그림 3(c)는 영역 확장의 결과로 불확실 영역 내의 화소들을 영역 결정한 그림이다.

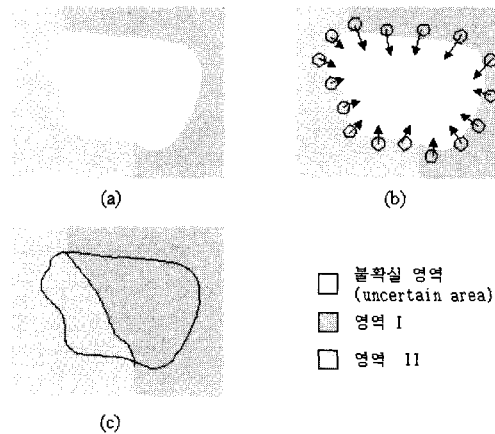


그림 3. 영역 확장
Fig. 3. Region growing.

3. 투영이 불분명한 영역 추출

움직임 모델은 객체의 윤곽선 부근에 투영 결과가 정확하지 않은 것으로 알려져 있다. 투영이 불분명한 영역 추출 단계는 투영의 결과와 영상의 공간적 정보를 이용하여 객체의 윤곽선 부근에서 움직임 모델의 신뢰도가 적은 불확실 영역을 추출한다. 즉 객체 투영이 잘못되었다고 고려되는 부분과 투영된 객체와 배경

이 비슷하여 재조정이 필요하다고 고려되는 부분을 선택한다. 이 과정은 움직임 보상 오차를 통한 객체 단계와 투영된 객체 윤곽선에 인접한 배경과 객체의 화소 색상 비교의 두 단계로 구성하여 제안하였다.

(1) 움직임 보상 오차

투영이 불확실한 화소는 움직임이 잘못 추정되었기 때문이라고 할 수 있다. 이전 프레임과 현재 프레임의 추정된 움직임으로 객체를 보상했을 때, 그 보상 오차가 큰 화소의 경우 추정된 움직임은 불확실하다고 생각할 수 있다. 이에 근거하여 불확실 영역 U_{mc} 를 구할 수 있다.

이전 프레임의 추적 결과 O_{n-1} 를 제외한 배경 부분을 B_{n-1} 라 하면 O_{n-1} 와 에 대해 각각 이전 프레임 I_{n-1} 로부터 현재 프레임 I_n 로의 매개 변수 움직임 모델을 추정할 수 있다. 그리고 식 (1)에서 볼 수 있듯이 affine 움직임 모델은 간단한 2×2 행렬 연산을 통해 현재프레임 I_n 에서 이전 프레임 I_{n-1} 으로의 움직임 모델로 변환할 수 있다. 이렇게 구한 객체의 움직임 모델을 f_2 , 그리고 배경의 움직임 모델을 f_1 이라 하면, 식 (2)와 같이 현재 n번째 프레임의 움직임 보상 오차 e_n 이 계산된다.

$$e_n(x, y) = \min_j |I_n(x, y) - I_{n-1}(x'_j, y'_j)|$$

$$\text{where } (x, y) \in I_n, (x'_j, y'_j) \in R(x', y')$$

$$(x', y') = f_i(x, y), i = 1 \text{ or } 2 \quad (2)$$

여기에서 I_n 은 현재 프레임의 밝기 영상, I_{n-1} 은 이전 프레임의 밝기 영상이고, (x, y) 는 현재 프레임의 화소, (x', y') 는 (x, y) 에 대해 매개 변수 움직임 모델로 구한 이전 프레임의 실수형 화소이다. 그리고, $R(x', y')$ 는 (x', y') 를 둘러싸고 있는 4개의 정수형 화소를 뜻한다. 이 때, (x, y) 가 II.1.의 투영 결과 $O_{n,p}$ 에 속하면 (x', y') 는 객체의 움직임 모델 f_2 를 이용하여 구하고, (x, y) 가 $O_{n,p}$ 에 포함되지 않으면 f_1 을 이용하여 (x', y') 을 구한다.

이 단계를 통해서 화소 (x, y) 에서 움직임 추정이 옳지 않아 움직임 보상 오차가 크거나 혹은 객체가 움직여서 나타난 이전 프레임에는 없었던 배경 화소를 추출할 수 있다.

(2) 인접한 배경과 객체의 화소 색상 비교

만약 배경과 객체가 확실한 색상 차이가 있다고 가

정한다면, 객체의 투영이 잘못된 경우에는 투영된 객체의 윤곽선에 인접한 객체와 배경은 비슷한 색을 가지고 있을 것이다. 또, 객체와 배경이 비슷한 색을 가지고 있는 경우, II.3.1의 움직임 보상 오차로는 투영이 옳지 않는 화소들을 추출할 수 없으며 자칫 잘못된 결정을 내리기가 쉽다. 이러한 점을 고려하여, 인접한 배경과 객체의 화소 색상 비교 과정에서는 $O_{n,p}$ 의 윤곽선 부근에서 색상이 비슷한 배경과 객체의 영역을 추출하여 불확실 영역에 포함하도록 제안하였다.

그림 4는 이 과정을 설명하고 있다. 그림 4(a)는 이 과정이 첫째, 배경과 비슷한 색상을 가지는 객체 영역 내 불확실 화소 추출과 둘째, 객체와 비슷한 색상을

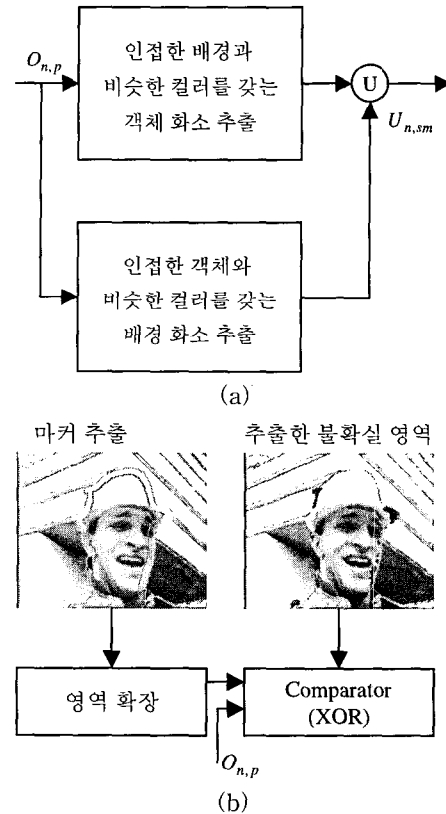


그림 4. 화소 색상 비교를 통한 불분명한 영역 추출. (a) 화소 색상 비교를 통한 불분명한 영역 추출 구조 (b) 인접한 배경과 비슷한 색상을 갖는 객체 화소 추출의 예

Fig. 4. Uncertain area extraction by color similarity test. (a) The scheme for uncertain area extraction based on color similarity test, and (b) extraction of object pixels having the color similar to the adjacent background pixels.

가진 배경 영역 내 불확실 화소 추출의 두 과정으로 나누어 실행됨을 설명하고 있다. 그리고, 각 추출 단계는 II.2에서 설명한 영역 확장을 이용한다. 그림 4(b)는 이 과정을 더 자세히 설명하고 있다. 특히 그림 4(b)는 배경과 비슷한 색상을 갖는 객체 영역 내 불확실 화소 추출 단계를 설명한다. 그림에서 알 수 있듯이 투영 결과 $O_{n,p}$ 을 이용하여 표시자를 선택한다. 이때 배경을 대표하는 표시자는 $O_{n,p}$ 의 윤곽선에 접한 배경의 화소들로 선택하고, 객체를 대표하는 표시자는 $O_{n,p}$ 윤곽선 내부에 위치한 화소들로 결정한다. 그리고 두 영역을 대표하는 표시자들 사이의 영역을 영역 확장에서의 불확실 영역으로 고려한다. 이 때, 영역 확장에서 watershed 알고리즘은 표시자와 불확실 영역 내 화소와 RGB 색차를 이용한다. 또, 비슷한 색상 값을 갖는 화소만을 추출하기 위해서, watershed 알고리즘에서 색차가 문턱치를 넘는 화소를 만나면, 더 이상 성장하지 않도록 했다.

투영이 불분명한 화소는 위에서 설명한 영역 확장 결과와 투영된 결과 $O_{n,p}$ 와 비교하여 다르게 결정된 화소, 즉 투영의 결과에서는 객체였으나 영역 확장 과정을 통해서 배경과 비슷한 색상을 갖고 있기 때문에 배경을 나타내는 표시자로부터 성장한 화소들로 결정된다.

배경 영역 역시 비슷한 방법으로 객체를 나타내는 표시자를 투영된 객체 $O_{n,p}$ 의 윤곽선에서, 배경을 나타내는 표시자는 배경 내부의 화소로 고려함으로써, 객체와 비슷한 색상을 가지는 화소를 추출할 수 있다.

4. 윤곽선 재조정

만약 객체의 윤곽선 부근에서 배경이나 객체가 복잡한 가장자리(edge)를 가지고 있거나 객체의 색상이 배경과 비슷하면 단 한번의 윤곽선 재조정은 잘못된 결과를 낼 수 있다. 앞서 II.1에서 구한 투영 결과 $O_{n,p}$ 는 윤곽선 부분에서 정확한 결과는 아니나, 객체의 모양은 유지한다고 볼 수 있다. 이 점을 이용하여 잘못 결정될 수 있는 영역을 여러 단계로 다시 고려함으로써, 기존의 방법에 비해 정확한 객체의 윤곽선을 얻을 수 있는 가능성을 증가시킨다.

앞 II.3에서 설명한 투영이 불분명한 화소 추출 결과는 U_{mc} 와 U_{sm} 에서 추출한 화소들로 구성된다. 그리고, 불확실 영역과 그 부근의 화소들은 영역 확장을 이용한 3단계 윤곽선 결정을 통해 최종 윤곽선을 결정

한다. 윤곽선 결정은 RGB 값을 이용하였다. 그림 5는 제안한 알고리즘을 설명한다.

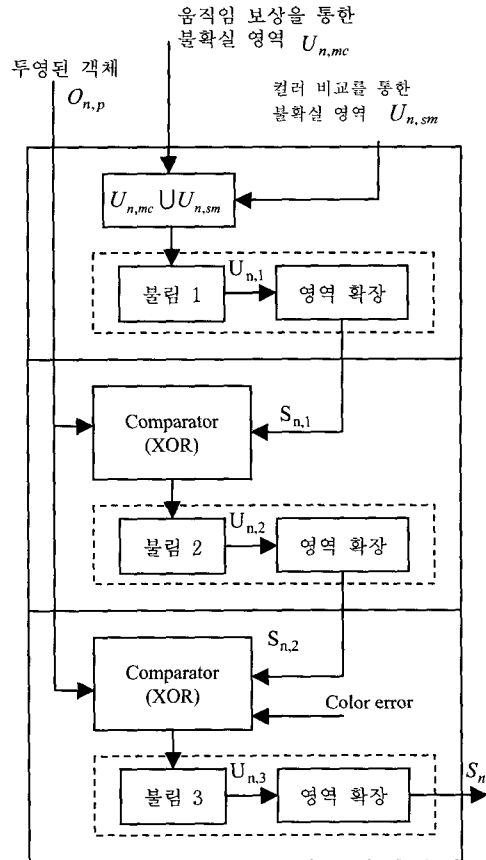


그림 5. 윤곽선 재조정
Fig. 5. Boundary refinement.

첫 단계는 수리 형태학(mathematical morphology)의 불림(dilation)을 이용하여 불확실 영역을 일정한 폭만큼 확장한다. 그리고, 확장된 불확실 영역에 대해 II.2에서 설명한 영역 확장을 통해 불확실 영역 내의 화소가 객체 또는 배경인지를 결정한다. 윤곽선 재조정의 첫 단계 결과는 $S_{n,1}$ 이다.

다음 단계에서는 첫 단계에서 구한 $S_{n,1}$ 과 $O_{n,p}$ 를 비교하여 영역이 서로 다르게 결정된 화소들만을 불확실 영역으로 고려한다. 그리고, 첫 단계와 같은 방법을 통해 $S_{n,2}$ 를 얻는다.

마지막 윤곽선 재조정 단계는 $S_{n,2}$ 와 $O_{n,p}$ 를 비교하여 불확실 영역으로 고려한다. 움직이는 객체의 경우 객체의 윤곽선 근처는 움직임에 따른 영상의 흐림(blurring)이 발생하게 되므로, 불확실 영역에 대해 표

시자가 너무 가깝게 뽑히면 잘못된 가장자리로 윤곽선이 이동할 수 있다. 또, 잡음이 강한 화소와 영역확장에서 표시자가 가까이 위치할수록 잡음의 영향으로 영역 확장 결과가 어긋날 수 있다. 이러한 두 가지 사실을 고려하여 마지막 재조정 단계는 $O_{n,b}$ 의 윤곽선 근처의 화소 중 프레임과 색상 차가 문턱치 이상인 화소에 대해서는 $S_{n,2}$ 의 결정을 따르도록 했다

III. 모의 실험

이 논문에서 제안한 객체 추적 알고리즘의 실험을 위해서, MPEG-2 및 MPEG-4의 표준 영상 중 'Miss America,' 'Akiyo,' 'Mother and Daughter,' 'Foreman' 영상을 이용하였다. 각 실험 영상은 모두 176×144 크기의 QCIF 30Hz 영상을 사용했으며, 실

험 영상의 첫 프레임 (intra frame)의 영상 분할 결과는 사람에 의해서 직접 주어진 것으로 사용하였다.

각 영상의 추적 알고리즘에 사용한 문턱치 값은 다음과 같다. 투영이 불분명한 화소를 추출하는 과정에서 움직임 보상 오차는 투영된 객체 윤곽선부터 폭 2내의 화소만을 고려하였고, 윤곽선 부근의 객체와 배경의 색상 비교 과정에서 객체 또는 배경을 나타내는 표시자는 각각 투영된 윤곽선에서 폭 4만큼 안쪽에서 추출하였다. 또, 색상은 그 차가 30이하인 경우 비슷하다고 고려하였다. 또한 윤곽선 재조정에서 불림 (dilation)의 형태소 크기는 첫 단계와 두 번째 단계는 2로 선택하였고, 마지막 단계에서는 1로 결정하였다. 또, 윤곽선 재조정 마지막 단계에서는 투영된 윤곽선 부근에서 움직임 보상 오차 단계에서 추출된 화소들 중 이전 프레임과 현재프레임의 화소의 색상 차가 50



그림 6. 실험 결과 (a) Akiyo, (b) Foreman, (c) Miss America, (d) Mother and daughter
Fig. 6. Simulation results. (a) Akiyo, (b) Foreman, (c) Miss America, and (d) Mother and daughter.

이상인 화소들은 2번째 윤곽선 조정 결과에 따르도록 했다.

그림 6은 위의 실험환경을 이용하여 제안한 방법으로 객체를 추적한 결과를 보여주고 있다. 실험 결과는 첫 번째 장에서 객체를 정의한 후, 그 이후 프레임은 자동으로 사람의 도움 없이 객체를 추적한 결과이다. 실험 결과를 살펴 보면 이전 방법에서 추적 오차가 쉽게 누적되었던 배경과 객체가 비슷한 색상을 가진 영역에서도 추적이 잘 수행됨을 볼 수 있으며, 객체의 움직임이 큰 경우도 추적이 비교적 잘 수행됨을 확인할 수 있다. 또, 추적 오차가 작은 경우에는 몇 프레임이 지난 후에 다시 정확히 수정되는 장점을 보여준다.

IV. 결 론

동영상에서 의미 있는 객체를 추적하는 방법을 제안하였다. 자동으로 의미 있는 객체를 추적하는 것은 아직까지 많은 문제점을 안고 있다. 따라서 이 논문은 동영상의 첫 프레임에서는 사용자가 원하는 객체를 분할하고, 다음 프레임부터 자동으로 객체를 추적하는 방식을 택했다. 객체를 추적하기 위해 매개 변수 움직임 모델을 이용하여 이전 프레임의 객체를 현재 프레임에 투영하고, 투영된 윤곽선을 재조정하여 객체를 추적한다. 특히 이 논문에서는 투영이 불분명한 화소를 움직임 보상 오차와 색상 비교를 통해 추출하고, 추출한 화소를 중심으로 윤곽선을 3단계에 걸쳐 재조정하는 방법을 제안하였다.

모의 실험에 의한 이전 방법과 비교를 통해, 제안한 알고리즘이 배경과 객체의 색상이 비슷한 영역에서도 투영된 결과를 고려하여 불확실 영역을 추출함으로써, 객체의 윤곽선을 정확히 추적해 낼 수 있었다. 또, 복잡한 가장자리를 갖고 있는 경우에도, 여러 번 윤곽선을 재조정함으로써 이전 결과에 비해 꽤 정확한 추적 결과를 얻을 수 있었다. 그러나, 윤곽선 부근에서의 배경이나 객체의 색상 정보나 경계 자체가 복잡하고 객체 윤곽선 근처 영역이 객체 내부보다 오히려 배경과 비슷한 경우에는 간혹 잘못된 윤곽선으로 이동하는 단점을 보여준다. 또, 객체의 움직임이 다양한 경우, affine 움직임 모델은 객체의 움직임을 충분히 표현할 수 없어서 객체의 일부분을 잃는 단점이 나타나므로, 영상 분할 기법을 이용한 연구가 더해져야 한다. 향후 이러한 문제점들의 해결을 위한 연구가 필요하다.

참 고 문 헌

- [1] ISO/IEC JTC1/SC29/WG11 N1730, "MPEG-4 Overview," Stockholm MPEG meeting, July 1997.
- [2] ISO/IEC JTC1/SC29/WG11, "MPEG-4 Video Verification Model Version 8.0," July 1997.
- [3] M. Kunt, A. Ikonopoulos, and M. Kocher, "Second generation image-coding techniques," Proc. IEEE, vol. 73, no. 11, pp. 549-574, Apr. 1985.
- [4] H. S. Park and J. B. Ra, "Efficient marker extraction algorithm for fine image segmentation," IEE Electronic Letters, vol. 34, no. 22, pp. 2107-2108, Oct.1998.
- [5] J.G. Choi, S. Jee, and S.D. Kim, "Spatio-temporal video segmentation using a joint similarity measure," IEEE Trans. Circuits Syst. Video Technol., vol. 7, no. 2, Apr. 1997.
- [6] J. Yi, H.S. Park, and J.B. Ra, "A hierarchical semantic video object tracking algorithm using mathematical morphology," Intern. Workshop on Advanced Image Technology, vol. 1, pp. 29-34, Cheju, Korea, Jan. 14-15, 1998.
- [7] C. Gu and M. Lee, "Semiautomatic segmentation and tracking of semantic video object," IEEE Trans. Circuits Syst. Video Technol., vol. 8, no. 5, pp. 572-584, Sept. 1998.
- [8] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," IEEE Trans. Pattern Anal. Machine Intell., vol. 13, no. 6, June 1991.
- [9] F. Meyer and S. Beucher, "Morphological segmentation," J. Visual Commun, Image Representation, vol. 1, no. 1, Sept. 1990.
- [10] P. Salembier and M. Pardas, "Hierarchical morphological segmentation for image sequence coding," IEEE Trans. Image Processing, vol. 3, no. 5, pp. 639-651, Sept. 1994.

 저 자 소 개



任貞恩(正會員)

1997년 2월 한국과학기술원 전기 및 전자공학과 졸업(공학사). 1999년 2월 한국과학기술원 전기 및 전자공학과 졸업(공학석사). 1999년 3월~현재 한국과학기술원 전자전산학과 박사 과정. 주관심분야

는 영상 분할, 비디오 신호처리, 영상시스템



李在淵(正會員)

1996년 2월 한국과학기술원 전기 및 전자공학과 졸업(공학사). 1998년 2월 한국과학기술원 전기 및 전자공학과 졸업(공학석사). 1998년 3월~현재 한국과학기술원 전자전산학과 박사 과정. 주관심분야

는 영상분할, 의료 영상 처리, 영상 시스템

羅鍾範(正會員)

1975년 2월 서울대학교 전자공학과 졸업(공학사). 1977년 2월 한국과학기술원 전기 및 전자공학과 졸업(공학석사). 1983년 2월 한국과학기술원 전기 및 전자공학과 졸업(공학박사). 1987년 7월~현재 한국과학기술원 전자전산학과 교수. 주관심분야는 디지털 영상 처리, 비디오 신호처리, 3차원 시각화