

영역 최적화를 이용한 움직이는 물체의 추적

장원달^u 윤태수 양황규
 동서대학교 인터넷 공과부
 xdmz2000@hanmail.net

Moving Object Tracking Using Region Optimization

Won dai Jang^u Tae Soo Yun Hwang Kyu Yang
 Division of Internet Engineering, Dongseo University

요약

본 논문에서는 고정된 카메라로부터 입력된 연속 영상에서 움직이는 오브젝트의 영역을 오브젝트의 형태 변화에 덜 민감하게 추적하는 방법을 제안한다. 제안된 방법은 차영상 분석에 의해 오브젝트의 움직임을 감지하는 단계와 연결성 검사 방법을 적용하여 오브젝트의 영역을 찾아내는 단계와 최적 영역 알고리즘을 이용하여 오브젝트에 최적화된 사각의 영역을 구하는 단계로 구성되어 있다. 제안된 방법을 카메라로부터 입력된 연속영상들을 대상으로 실험해본 결과 오브젝트의 형태변화에 덜 민감한 최적영역을 잘 찾아낼 수 있었다.

1. 서론

연속 영상에서 움직이는 물체를 추적하는 일은 비디오 감시 시스템, 침입감시 시스템 등 컴퓨터 비전 응용 프로그램에 기본적인 작업이다. [1] 지금까지의 관련 연구로는 Optical Flow [2], Kalman filter [3], Template Matching [4] 등이 있다. 이런 방법들은 정확한 움직임을 추적해 내지만 픽셀 기반의 많은 복잡한 연산을 필요로 하므로 움직이는 카메라를 이용한 단순한 유인감시 시스템 등과 같은 실시간 처리에 적용할 경우 보안되어야 할 점들이 있다.

본 논문에서는 연속 영상 속 오브젝트의 전체적인 움직임을 오브젝트의 형태 변화에 덜 민감하게 추적하는 방법을 제안한다. 오브젝트의 전체적인 움직임은 오브젝트의 세세한 움직임(예 : 팔의 움직임 등)에 거의 영향을 받지 않으므로 우리는 오브젝트의 영역을 사각형으로 가정한다. 제안된 방법의 첫 번째 단

계에서는 차영상 분석방법을 사용하여 영상 속 오브젝트의 움직임을 감지한다. 두 번째 단계에서는 연결성 검사를 이용하여 추출한 움직임 정보에서 오브젝트가 나타나는 영역을 찾는다. 마지막 단계에서는 이전 단계에서 추출한 영역을 본 논문에서 제시하는 영역 최적화(Region Optimization)알고리즘을 이용하여 오브젝트에 최적화된 사각의 영역을 찾아낸다. 그림 1 은 본 논문에서 제안된 방법의 전체적인 개요도이다.

2. 움직임 감지

움직임 감지 단계에서는 시공간정보를 이용한다[5]. 두 장의 연속 영상 속의 오브젝트는 짧은 시간 간격에도 미세한 움직임을 가질 것이므로 I_k 와 I_{k+1} 의 시간차 영상 D_k 에서는 오브젝트 움직임에 의해 발생한 명도변화가 나타날 것이다. I_{k+1} 번째 프레임과 배경(B)의 차영상 Db_{k+1} 은 오브젝트의 공간정보를 나타낸다. 시간차는 식(1), 공간차는 식(2)와 같이 구한다. 차영상을 구하기 전에 입력된 영상을 그레이 레벨영상으로 변환을 한다. 차영상 방법에서는 Saturation기법을 적용한다.

$$D_k(x, y) = |I_k(x, y) - I_{k+1}(x, y)| \quad (1)$$

$$Db_{k+1}(x, y) = |I_{k+1}(x, y) - B(x, y)| \quad (2)$$

다음으로 D_k 와 Db_{k+1} 의 차영상을 구하고 영상을 이진화한다. 시공간 정보를 나타내는 차영상을 다음과 같이 정의한다. (t_b 는 영상을 이진화하는 임계값이다.)

$$M_k(x, y) = \begin{cases} 1, & \text{if } (D_k(x, y) - Db_{k+1}(x, y)) > t_b \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

시공간 정보를 나타내는 차영상 속에는 오브젝트의 움직임에 의한 명도값의 변화 이외에 조명의 영향에 의한 명도값 변화 정보 등 노이즈가 포함되어 있다. 따라서 노이즈를 효과적으로 제거하기 위해 영상을 이진화 시키는 적당한 임계값을 결정하는 것이 중요하다.

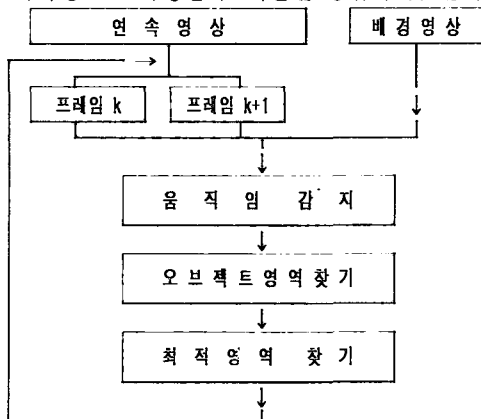


그림 1. 제안된 방법의 개요도

3. 오브젝트 영역 찾기

이전 단계에서 추출한 움직임 정보는(M_k) 움직임에 의해 명도값이 변화된 영역들을 나타낸다. 이 단계에서는 움직임 정보에 분포하는 움직임값들에 연결성 검사를 수행하여 움직임이 연결된 영역들을 모두 찾아낸다.



(a) I_k (b) I_{k+1} (c) B
(d) M_k (e) M_k 의 연결된 영역 (f) Box

그림 2. 연결성 검사를 수행한 결과영상

그림 2는 카메라로부터 입력된 영상과 본 단계까지 수행한 결과 영상을 나타낸다. 연결성 검사를 수행하여 움직임이 연결된 영역들을 모두 찾아낸 그림(e)에서는 영상의 조명영향에 의해 오브젝트 이외에 다수의 연결된 영역들이 나타나고 있다. 명도값의 변화가 큰 군집화를 이루는 영역이 오브젝트가 나타나는 부분일 것이므로 연결된 영역들중 가장 큰 영역이 이 단계에서 구하고자 하는 오브젝트 영역(Box)이 된다. 연결된 영역과 Box의 관계를 수식으로 표현하면 다음과 같다.

$$C = \{c_i : c_i \text{ connected block in } M_k\} \quad (4)$$

$$RC = \{rc_i : rc_i = \text{connected block } c_i \text{ 를 포함하는 minimum area rectangle}\} \quad (5)$$

$$Box = \max_{rc_i \in RC} \text{area}(rc_i) \quad (6)$$

여기서 C는 연결된 영역들의 집합이고 RC는 연결된 영역 c_i 를 포함하는 최소 크기의 사각(rectangle)영역들의 집합이다. rc_i 는 M_k 속의 시작좌표 (x,y)와 넓이 w, 높이 h로 표현되어진다.

4. 최적영역 찾기

마지막 단계에서는 오브젝트 영역을 오브젝트의 형태변화에 치우치지 않게 최적화 시킨다. M_k 속에 있는 object와 Box 그리고 이 단계에서 찾고자 하는 최적 영역 Box_{Opt} 의 관계를 그림으로 표현하면 다음과 같다.

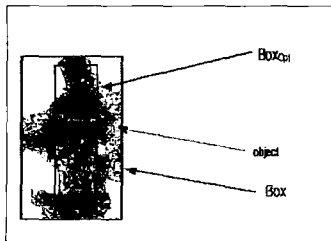


그림 3. 오브젝트 영역과 최적영역

그림에서 바깥쪽 사각형은 오브젝트 영역(Box)이고 이 영역 내부의 검은색 점들이 오브젝트(object)이고 내부의 사각형은 최적 영역(Box_{Opt})을 나타낸다. 영상에 노이즈들이 심하게 출현하고 있지만 무시한다. 최적영역을 찾는 영역 최적화(Region Optimization) 알고리즘은 오브젝트 영역내 오브젝트의 중심 픽셀들을포함하면서 종료조건을 만족 할 때까지 상하좌우 방향으로 줄여나간다. 그림 4는 본 논문에서 제시하는 영역 최적화 알고리즘을 나타낸다.

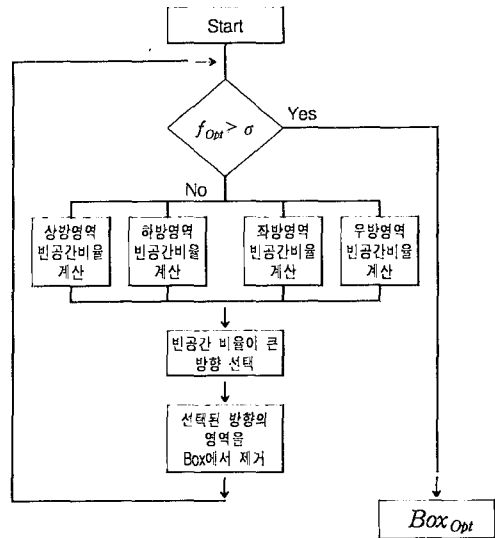


그림 4. 영역최적화(Region Optimization) 알고리즘

f_{Opt} 는 Box내 존재하는 오브젝트의 비율이며 수식 (7)과 같이 표현된다.

$$f_{Opt}(Box) = \frac{\sum_{i=y_1}^{y_2} \sum_{j=x_1}^{x_2} M_k(i,j)}{w \times h} \quad \therefore 0 < f_{Opt} < 1 \quad (7)$$

σ 는 Box_{Opt} 를 결정지어주는 0과 1사이의 상수값이며 영역제거를 위한 상방영역, 하방영역, 좌방영역, 우방영역은 그림 5에서 나타낸바와 같이 α 에 의해서 결정되는데 이 값을크게하면 최적영역을 빨리 찾을수 있겠지만 부드럽지 못하고 이 값을 너무 작게 주면 최적영역을 찾는데 많은 시간이 소요된다.

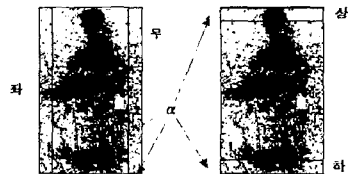


그림 5. 상,하,좌,우 방 영역

각 방향 영역의 빈공간 비율은 다음과 같이 계산한다.

$$\text{빈공간비율} = \frac{\text{빈공간}}{\text{영역의면적}} \quad (8)$$

5. 실험결과 및 고찰

제안한 방법을 검증하기 위해서 본 논문에서 제시한 방법으로 연구실내 정지된 카메라로부터 초당 30 프레임으로 입력된 이미지(크기 : 320×240)에서 오브젝트의 전체적인 움직임을 7 프레임 간격으로 추적해 보았다. 실험에 사용한 시스템은 Matrox meteo II 프레임 그래픽이 장착된 Pentium III 650 Mhz이고 프로그램 환경은 Visual C++ 6.0이다. 실험에 사용한 t_b, α, σ 는 각각 7, 4, 0.75으로 실험적으로 구했다. 그 결과가 그림 7에 나타나 있다. 그림에서 알수 있는 바와 같이 제안된 방법은 오브젝트의 전체적인 움직임을 잘 추적할수 있었다. 전체적인 움직임을 확인해 보기 위해 최적영역의 중심좌표를 다음표에 나타내었다.

표 1. 최적영역의 중심좌표

Frame	0	k	k+7	k+14	k+21	k+28	k+35	k+42
(x,y)	(203,127)	(190,124)	(166,109)	(145,94)	(130,99)	(116,102)	(98,104)	
Frame	k+49	k+56	k+63	k+70	k+77	k+84	k+91	k+98
(x,y)	(71,106)	(59,103)	(49,98)	(48,85)	(45,78)	(38,90)	(32,78)	(27,74)

표에 나타난 값을 그래프로 그려보면 다음과 같다.

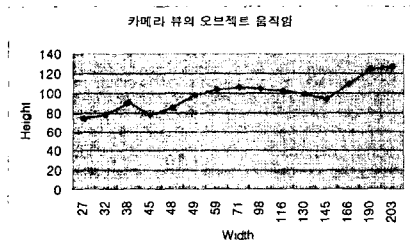


그림 6. 오브젝트의 전체적인 움직임

위의 그래프는 320×240 이미지 상에서 위에서 좌로 움직인 오브젝트의 중심의 이동을 보여주고 있다. 그래프 상에 곡선이 심하게 굴곡되는 부분이 나타나는데 이는 영상속에 있는 오브젝트의 그림자에 의한 영향으로 추측된다.

6. 결론

본 논문에서는 카메라로부터 입력된 연속 영상에서 오브젝트의 전체적인 움직임을 오브젝트의 형태 변화에 덜 민감하게 추적하는 방법을 제안하였다. 제안된 방법은 차영상을 이용하여 움직임을 감지하였고 연결성 검사를 이용하여 오브젝트 영역을 찾아내었고 최적영역 알고리즘을 이용하여 오브젝트에 최적화된 영역을 찾아내었다. 본 논문에서 제시한 최적영역을 이용하여 오브젝트의 전체적인 움직임을 추적해본 결과 개략적인 움직임을 잘 추출할 수 있었다. 향후과제로 영상 내부 오브젝트의 그림자를 제거하는 연구가 필요하다.

참고 문헌

[1] Jea Sig Chang, Jong Bae Kim, Chang Woo Lee, Hang Joog Kim, "Moving Object Segmentation Using Spatio-Temporal Information," Proc. of The KIPS Summer Conference, Vol. 1 No.1, pp 217-220, 2001
 [2] Ryuzo Okada, Yoshiaki Shirai and Jun Miura, "Object Tracking based on Optical Flow and Depth," Proc. of RWC, pp 432-439, 1997
 [3] Vigus S.A, Bull D.R, Canagarajah C.N, "Video object tracking using region split and merge and a kalman filter tracking algorithm," Proc. International Conference on Image Processing, Vol. 1, pp 650-653, 2001
 [4] Atallah M.J, "Faster image template matching in the sum of the absolute value of differences measure," IEEE Transactions on Image Processing, Vol. 10 Issue: 4, April 2001 Page(s): 659 -663
 [5] M. Betke, E. Haritaoglu and L. S. Davis, "Multiple Vehicle Detection and Tracking in Hard Real-Time," Proc. of IEEE Intelligent Vehicles Symposium, pp. 351-356, 1996

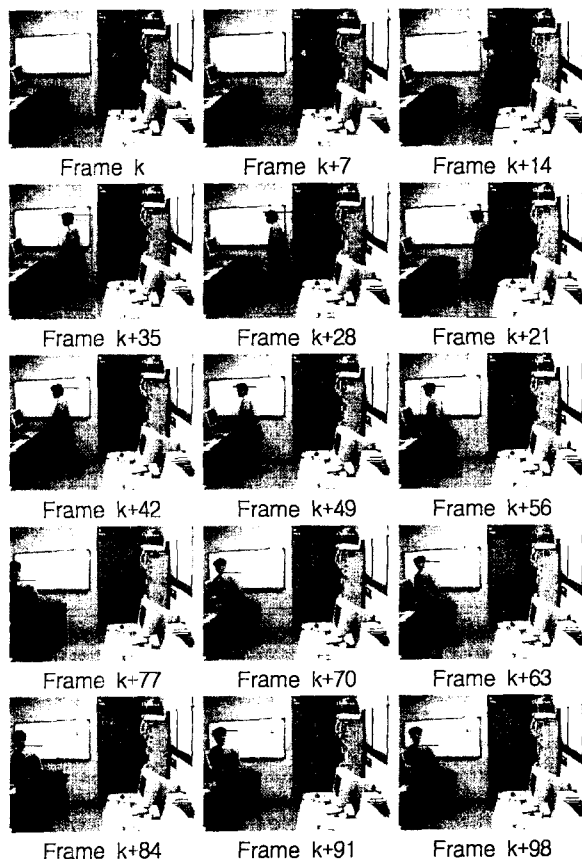


그림 7. 실험결과영상