

## 객체의 대칭성과 거리 벡터를 사용한 그림자 제거 방법

이정원<sup>○\*</sup> 최창규<sup>\*</sup> 조정현<sup>\*\*</sup> 김승호<sup>\*</sup>

<sup>\*</sup>경북대학교 컴퓨터공학과, <sup>\*\*</sup>영남이공대학 컴퓨터정보기술계열

jwlee<sup>○\*</sup>@mmlab.knu.ac.kr, cgchoi@borami.knu.ac.kr, petercho@ync.ac.kr, shkim@knu.ac.kr

### Shadow casting method using symmetric and distance feature of the object region.

JungWon Lee<sup>○\*</sup> C.G Choi<sup>\*</sup> J.H Cho<sup>\*\*</sup> SungHo Kim<sup>\*</sup>

<sup>\*</sup>Dept. of Computer Engineering, Kyungpook National University

<sup>\*\*</sup>Division of Computer Information Technology, Yeungnam College of Science & Technology

#### 요 약

본 논문에서는 감시 시스템 내에서 검출된 객체에 대해 정확한 특징벡터를 추출하기 위한 그림자 제거 (shadow casting) 방법을 제안한다. 그림자에 의해 부정확한 특징벡터를 가지게 되는 객체는 동일한 객체임에도 불구하고 서로 다른 객체로 인식하는 잘못된 결과를 가져온다. 이러한 문제점을 해결하기 위해 객체가 가지는 대칭성을 사용하여 그림자 후보 영역을 추출한 후 중심축으로부터의 거리에 비례한 가중치값을 사용하여, 추출한 영역에 대해 그림자를 제거를 수행한다.

#### 1. 서 론

최근 실시간 영상에서 움직이는 객체를 추출하고 추적하는 일은 감시 시스템, 패턴 인식 시스템에서 주요한 문제가 되고 있으며, 관련된 연구가 이루어지고 있다 [2]. 특히, 연속된 영상 내에서 검출된 객체의 주요한 특징벡터를 추출하고, 경로 추적하는 것은 감시 시스템에서 이벤트 검출 (event detection) 시 중요한 특징벡터가 된다. 그러나, 추출된 객체가 그림자 영역을 포함하여, 추출된 경우 정확하지 않은 특징벡터추출의 원인이 된다. 본 논문에서는 사람이 가지는 대칭성과 중심축에서의 거리를 사용한 그림자 영역을 제거하는 방법을 제안한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 그림자 제거에 대한 관련 연구를 소개하고, 3장에서는 본 논문에서 제안한 대칭성과 거리벡터를 사용한 그림자 제거 방법을 설명한다. 4, 5장에서는 실험 결과와 결론을 기술한다.

#### 2. 관련 연구

그림자에 의해 발생하는 잘못된 객체 인식과 경로 추적의 문제를 해결하기 위해 많은 연구가 이루어 지고 있다. 그 중에서 대표적인 방법으로는 통계적인 방법과 모델 기반 (model based) 방법이 있다. 전자의 경우 노이즈 (noise)의 민감도를 줄일수 있는 장점이 있고, 후자의 경우는 좋은 결과를 얻을수 있으나 복잡하고 시간이 많이 걸리는 단점이 있었다 [7]. 그림자 제거 시에 주로 사용하는 특성은 스펙트럼, 공간적, 시간적 특성이다. 스펙트럼 정보는 명암 (brightness), 칼라 (color) 정보를 이용하는 것을 말하며 추가적으로 영역 (region)과 프레임 (frame) 레벨의 공간적인 특성과 시간적인 특성을 함께 사용하면

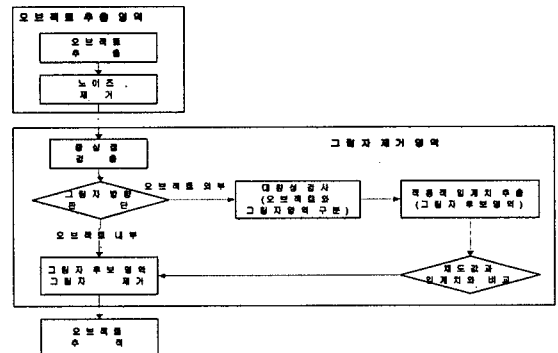


그림 1. 그림자 제거 순서도

보다 향상된 결과를 얻을 수 있다. 최근에는 물리적인 특성을 사용하는 그림자 제거 방법이 많이 연구되고 있다 [3, 4].

#### 3. 그림자 제거

배경 모델링 (background modeling) 후 생성된 배경 영상과 연속된 입력영상과의 감산 연산에 의해 차감영상이 추출된다. 이 영상에 대해 노이즈를 제거한 후 각각의 객체에 대해서 그림 1의 순서와 같이 중심축을 검출 후, 그림자의 방향을 판단하고 대칭성을 검사해서 그림자와 객체를 구분한다. 그림자가 객체 내부에 있을 때는 색도값 (chromaticity)을 사용하고 외부에 있을 때는 중심축에서의 거리에 대해 가중치값을 적용하여 그림자를 제거한다.

3.1 중심축 검출

그림자 제거를 하기 위해 객체가 가지는 대칭성을 사용하여 그림자 영역과 객체 영역을 일차적으로 구분한다. 이 때, 그 기준이 되는 중심축은 일반적으로 추출된 객체의 머리위치를 가리킨다. 그림 2 (a)와 같이 객체에 대해 수직 히스토그램 (vertical histogram)을 구해서, 그 중에서 가장 큰 값을 가지는 X축의 좌표 값을 가진다 [1]. 그러나 아래 그림 2 (b)와 같이 움직이는 객체의 수직 히스토그램에서 가장 큰 X축의 좌표가 항상 중심축을 나타내는 것은 아니다.



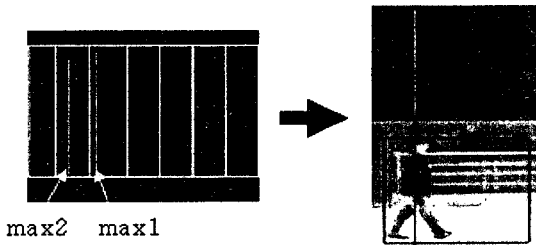
(a) 정확한 중심축을 찾은 경우



(b) 중심축을 잘못 찾은 경우

그림 2. 추출된 객체에 대한 중심축 선정

그림 2 (b)와 같이 잘못된 중심축을 검출함으로써 정확하지 않은 대칭 정보를 얻을 수 있기 때문에 그림 3 (a)와 같이 수직 히스토그램을 수평으로 임의의 개수의 블록 (block)으로 나누어 각 블록 별로 가장 큰 히스토그램의 X축의 위치를 max1, max2라 둘 때, 중심점 CP (center point)는 그림 3의 방법으로 구할 수 있다.



$$CP(x) = (\max 1(x) + \max 2(x)) / 2$$

(a) 블록화된 히스토그램 (b) 추출된 중심축  
그림 3. 중심축 검출

3.2 그림자 방향 검출

객체의 중심점을 추출한 후 그림자의 방향을 판단하여 그림자가 객체 내부에 있는지 외부에 있는지를 알아낸다. 이 정보를 사용해서 그림자가 내부에 있을 때는 실험적인 결과에 의해 구해진 색도값을 사용하여, 삼등분 된 객체의 하위에 적용하여 그림자를 제거한다. 그림자의 방향을  $Q$ 로 두고, 바운딩 박스의 오른쪽과 왼쪽의 끝 좌표를 Lx, Rx로 나타낼 때 그림 4와 같이 나타낼 수 있다.

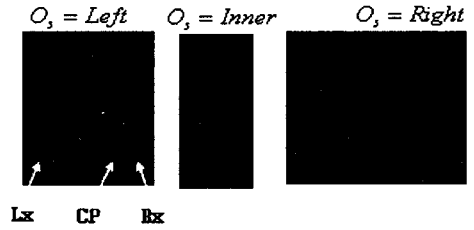


그림 4. 그림자의 방향

추출된 객체의 그림자의 방향은 그림 4에 정의된 변수들을 식 1에 대입해서 구할 수 있다. 그림자가 외부에 있을 때에는 다음 3.3절의 대칭성을 이용한 그림자 후보영역 검출 방법을 사용해서 그림자 영역을 제거 한다.

$$O_s = \begin{cases} \text{Left, if } |Lx-CP| - |Rx-CP| > \epsilon \\ \text{Right, if } |Lx-CP| - |Rx-CP| < \epsilon \\ \text{Inner, otherwise} \end{cases} \quad (1)$$

3.3 대칭성을 이용한 그림자 후보영역 검출

추출된 객체의 중심축과 그림자의 방향성을 알아낸 다음 객체가 가지는 대칭성을 이용해서 객체 영역과 그림자 후보 영역으로 구분한다. 일반적으로 RGB 값은 정확한 그림자 제거를 하지 못하였다 [2]. 따라서 그림자를 제거하기 위해 밝기 변화에 민감도가 적은 색도 값을 사용한다.



그림 5. 대칭성을 이용한 그림자 후보 영역

추출된 객체를  $R_k$ 라 하고, 객체의 바운딩 박스의 좌, 우를 BL, BR이라 한다. 그리고 3.1절에서 검출한 CP를 기준으로 검색하여 객체  $R_k$ 에 포함되는 최초의 픽셀이 가지는 X축좌표  $S_i^x, S_i^y$ 를 식 2에 대입하여 객체 영역과 그림자 후보영역을 구분한다.

$$x = \begin{cases} \text{if } S_i^x > \operatorname{argmin}\{|CP-S_i^x|, |CP-S_i^y|\} + \epsilon \\ \quad \quad \quad : \text{Asymmetric} \\ \text{Otherwise} \\ \quad \quad \quad : \text{Symmetric} \end{cases} \quad (2)$$

여기서,  $S_i^x$ 는  $R_k$ 의 중심축 CP를 따라 움직이는 Y축 상의 X좌표에 있는 픽셀의 거리를 나타낸다.

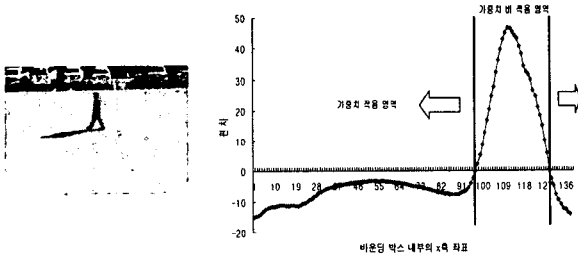
3.4 그림자 제거

객체  $R_k$  에서 3.3절의 대칭성을 이용한 방법을 사용하여 추출된 객체를  $R_k^1$ 으로, 그림자 후보영역을  $R_k^2$ 로 나타낸다. 그림자를 제거하기 위해 객체 영역  $R_k$ 의 각 픽셀이 가지는 색도값  $r_c, g_c$  (red, green)와 그림자 후보 영역  $R_k^2$ 의  $r_c, g_c$ 의 평균을 다음 식 3를 사용해서 구한다.

$$r_c = r / (r + g + b) \tag{3}$$

$$g_c = g / (r + g + b)$$

그림 6 (b)는 입력 영상 (a)의 매 프레임에 대해 수직 히스토그램의 평균을 구하여, 각각 히스토그램편차의 평균을 나타낸 것이다. 여기에서 알 수 있듯이 객체의 중심축으로부터 떨어진 것은 객체가 아닐 경우가 많다. 이 특성을 사용하여, 편차가 임의의 값이 이하인 영역 부터, 거리에 따른 가중치를 적용한다. 실험 결과에 의해 얻은 임계치를 중심축과의 거리에 따라  $R_k^2$ 의 평균값과  $R_k$ 의  $r_c, g_c$  간의 차값에 가중치가 적용된 임의의 값을 차감시킨 후, 비교하여 임계치보다 작은 값을 가지면 배경 영상으



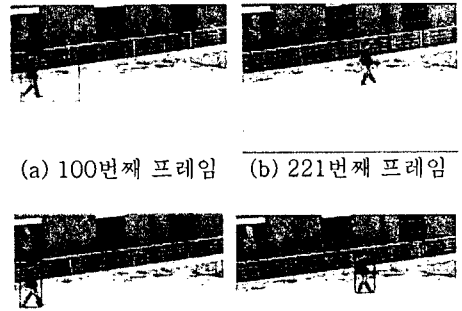
(a) 입력 영상 (b) 수직 히스토그램의 편차의 평균  
그림 6. 수직 히스토그램의 편차에 따른 가중치 적용영역

로 판단한다. 그러나, 전체 영역에 대해 하나의 임계치를 적용하면 객체가 손실되는 결과를 가져올 수 있다. 이러한 문제를 해결하기 위해 바운딩 박스 내의 영역을 상위(Upper), 중간(Middle), 하위바운드(Lower Bound)로 삼등분하고, 각각의 영역에 대한 임계치를 구한다. 그리고 그림자가 하위 바운드에 존재 할 때는 모든 영역에 대해 하지 않고 해당 영역에 대해서만 그림자 제거를 한다.

4. 실험 결과

실험은 RGB값을 가지는 연속된 영상 N 프레임을 사용하여 명암(intensity)의 중심값(median value)을 사용한 배경 모델링 후 N 프레임 이후에 들어오는 연속된 영상에 대해 차영상을 검출 한다. 추출된 영상에 노이즈를 제거 후 추출된 각각의 객체에 대해서 바운딩 박스를 씌운다. 검출된 각각의 객체의 바운딩 박스 내부 영역에 대해, 본 논문에서 제안한 그림자 제거 방법을 적용한 후 바운딩 박스의 사이즈를 업데이트 하는 방법을 사용하였다. 그림 7(a), (b)는 연속된 영상에서 그림자 제거가 되지 않고 단

순히 차영상만 사용한 경우의 오브젝트 추적 결과를 보여준다.



(a) 100번째 프레임 (b) 221번째 프레임

(c) 100번째 프레임 (d) 221번째 프레임

그림 7. 그림자 제거 실험 결과

그림 7 (c), (d) 동일한 프레임에서 그림자가 제거되어 올바른 객체 추적이 이루어지고 있음을 볼 수 있다.

5. 결론

본 논문에서는 차감 연산을 통해 추출된 객체가 가지는 대칭성과 거리 벡터를 사용하여 그림자를 제거하는 방법을 제안하였다. 그림자를 제거함으로써 올바른 특징 벡터를 추출할 수 있었고, 정확한 객체 추적이 가능함을 보였다. 차후에는 그림자와 오브젝트를 구분하는 물리적인 특성을 사용하여 유사한 색상에서도 정확한 그림자 제거가 가능하며, 그룹단위로 추출되는 객체에 대한 그림자 제거가 연구 되어야 할 것이다.

참고 문헌

[1] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Real Time Surveillance of People and Their Activities," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 809-830, Aug. 2000.  
 [2] S. McKenna, S. Jabri, Z. Duric, A. Rosenfeld and H. Wechsler, "Tracking Groups of People", Computer Vision and Image Understanding Volume 80, Issue 1, October 2000.  
 [3] J. Wei Hsieh, W. Fong Hu and C. Jung, "Shadow elimination for effective moving object detection by Gaussian shadow Modeling" Image and Vision Computing 21, 2003.  
 [4] S. Nadimi and B. Bhanu, "Physical Models for Moving Shadow and Object Detection in Video", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 26, no. 8, Aug. 2000.