

다중 카메라를 이용한 겹침 상황에서의 사람 추적

류정훈¹, 남윤영², 조위덕¹

¹아주대학교 유비쿼터스 시스템연구실, ²아주대학교 유비쿼터스 시스템연구센터
{dreami2, youngman, chowd}@ajou.ac.kr

Tracking People under Occlusion using Multiple Cameras

Junghun Ryu¹, Yunyoung Nam², We-Duke Cho¹

¹Ubiquitous System Lab, Ajou University

²Center of Excellence for Ubiquitous System, Ajou University

요 약

개인과 공공의 안전에 대한 요구가 증가함에 따라 카메라를 이용한 영상 감시 시스템이 점차적으로 증가하고 있다. 보안의 필요성에 따라 한 지역에 여러 대의 카메라를 설치하여 FOV(Field Of View)를 겹치는 경우도 있다. 이처럼 FOV의 중첩 영역에서 다중 카메라들로부터 얻은 영상을 처리하여 객체의 위치를 파악하고 추적하는 연구가 활발히 이루어지고 있다. 본 논문에서는 다수의 카메라를 이용해 감시 영역이 중첩되는 지역에서의 겹침이 발생해도 객체를 계속적으로 추적하는 방법을 제안한다. 이 방법으로 단일 카메라 상에서의 외형 식별자를 이용하여 추적하고 다중 카메라 간의 호모그래픽 매트릭스를 이용하여 노이즈에 강건한 시스템을 구현하였다.

1. 서 론

카메라 영상을 이용한 감시 시스템은 특정 상황을 인식하거나 인식할 수 있는 영역을 확장시킬 수 있다. 이를 통해 혼잡한 상황에서도 일련의 장면들을 통해서 객체를 추적할 수 있다. 영상을 통한 감시 시스템은 여러 잠재적인 응용 분야를 가지고 있다. 빌딩 내의 감시, 침입 감시 및 특정인의 위치 인식, 군사적 용도 등 다양한 분야에 응용이 가능하다.

감시 시스템에서 사람의 추적은 보안과 안전을 위해 필요한 기능 중에 하나이다. 영상 분석을 통한 사람 추적은 크게 3 가지 단계를 거친다. 먼저 추적하고 하는 대상을 인식하고 그 다음 영상 프레임에 따라 사람의 동일성을 판단하여 추적을 하며, 마지막으로 이를 통해 얻어진 이동 경로 및 정보를 통해 이동 방향 및 행동을 유추 한다. 이러한 과정에서 일어날 수 있는 객체 간의 겹침, 객체와 배경 간의 겹침, 객체 모양의 유동성, 카메라의 움직임, 3D 공간을 2D 공간으로 투영 시 정보의 손실, 이미지의 노이즈, 복잡한 객체의 움직임, 조도의 변화, 실시간 처리에 대한 요구 등에 의해 영상을 분석하는데 문제가 발생한다. 이러한 문제는 다중 카메라를 이용하여 해결할 수 있다.

본 논문에서는 단일 카메라 내에서 프레임 간의 객체의 위치를 고려하여 겹침 현상이 발생했을 경우 객체 외형 정보를 이용한 히스토그램을 통해 추적을 가능하게 하였다. 또한 다중 카메라에서 카메라 간의 호모그래픽 매트릭스를 계산하여 카메라 간의 사람 위치의 동일함을 판단하였으며 노이즈에 의한 추적 에러를 보완하였다.

본 논문의 구성은 다음과 같다. 2장에서는 단일 카메라와 다중 카메라에서의 객체 추적과 관련된 연구를 소개하고, 3장에서는 단일 카메라상에서의 객체의 탐지 및 외형 정보를 이용한 추적을 기술한다. 4장에서는 다중 카메라상에서 호모그래픽 매트릭스를 이용한 연속적인 추적을 서술한다. 5장에서는 실험 결과를 보이고 6장에서는 결론 및 향후 연구를 보인다.

2. 관련연구

Adam Baumber[1]는 사람을 추적하는 시스템을 개발하였다. 이 시스템은 사람 외형 2D 정보를 이용하여 추적을 수행한다. 보행자의 모습을 담은 일련의 비디오 이미지를 훈련시켜 걷는 사람의 모습을 모델링 하여 생성하였다. 모델링은 PCA(Principal Component Analysis) 방법을 통해서 생성하였다.

Olson[2]이 개발한 단일 사람 추적 시스템은 모션의 변화를 이용해 실내에서 실험하였다. 프레임을 통해 관계성이 있는 객체를 그래프로 형성한 다음 예측

본 연구는 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 지식경제부의 유비쿼터스컴퓨팅및네트워킹원천기반기술개발사업의 08B3-S3-10M 과제로 지원된 것임

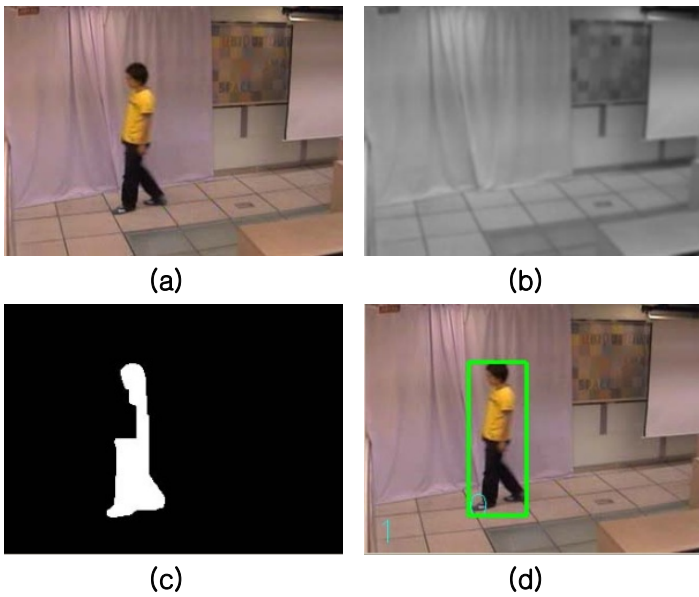


그림 1 전경 분리를 통한 객체 탐지
(a) 원본 이미지 (b) 배경 이미지
(c) 전경 분리된 이미지 (d) 탐지된 이미지

모듈을 통해 행동을 인식하지만, 작은 움직임은 처리하지 못하는 단점이 있다.

Haritaoglu[3]는 누가, 언제, 어디서, 무엇을 하는지를 파악하는데 중점을 둔 연구를 하였다. 기본 개념은 전경 분리를 통해 얻어진 데이터에 Feature-based 방식을 이용하여 객체 추적 및 행동에 대한 해석을 하는 것이다. 몸체, 다리, 손, 머리를 분류하여 사람의 위치 및 행동 해석에 이용하였다.

다중 카메라에 관련되어서는 다수의 카메라가 바라보는 중첩 FOV(overlapping fields of view)와 비중첩 FOV(non-overlapping fields of view)를 이용하여 각각의 카메라에서 보이는 객체의 동일성을 판단하고 객체를 추적하는 연구를 하였다.

Lipton[4]은 다수의 네트워크 카메라를 이용하여 넓은 지역을 감시하였으며, 장시간 동안 행동감시 및 자동차와 사람을 분류하여 추적이 가능하게 하였다.

M₂Tracker[5]는 영역 기반의 스테레오 알고리즘을 이용하여 3차원 위치를 파악하고 혼잡한 사항에서의 사람을 추출하기 위해 베이지안 픽셀 분류를 이용하였다.

본 논문에서는 단일 카메라로부터 얻어진 영상과 다중 카메라들로부터 얻은 영상을 이용하였으며, 각각의 카메라들 사이의 관계를 이용하여 객체간의 겹침을 보완하여 지속적인 추적을 가능하게 하였다.

3. 단일 카메라에서의 객체 추적

이미지에서 객체를 추적하기 위해서는 먼저 대상 객체를 탐지할 필요가 있다. 움직이는 객체를 탐지하기

위해서 전경 분리 방법을 이용한다. 전경 분리란 식 1에서처럼 미리 비교할 배경 이미지를 생성시킨 다음 현재의 이미지와의 픽셀 값의 차이를 이용하여 움직임이 있는 영역을 판별해 내는 것이다. 그림 1은 카메라로부터 얻은 이미지와 배경 이미지의 차이를 이용해 등장한 객체를 탐지한 것을 보이고 있다. 이 방법의 문제점은 환경 변화에 따라 배경 모델링에 영향을 줄 수 있다는 것이다.

$$R(P_{ij}) \begin{cases} 1(\text{foreground}), & \text{if } |H_{ij} - H_{b_{ij}}| > \omega\sigma(H_{h_{ij}}) \\ 0(\text{background}), & \text{otherwise} \end{cases} \quad (1)$$

H_{ij} : Hue, P_{ij} : current pixel, $H_{b_{ij}}$: average Hue,

$\sigma(H_{h_{ij}})$: Hue variance, ω : weight

전경 분리 방법을 통해 얻은 블랍(Blob)[6]은 하나의 독립적인 객체이거나 여러 블랍들의 집합체가 될 수도 있다. 그림 2는 트리 구조를 이용하여 블랍의 컨테이너적인 성격을 표현한 것이다.

객체 추적의 목적은 일련의 이미지 상에서 객체의 위치와 속력, 모양, 질감, 색상 등의 정보를 통해서 대상 객체를 지속적으로 파악하는 것이다. 일반적으로 추적 과정은 모양 및 움직임, 그리고 다른 영상 정보를 가지고 일련의 프레임 별 대상 객체를 매칭한다. 전경 분리로 탐지된 각각의 객체를 사각형으로 표시하고 사각형 중심점을 추적의 요소로 지정한다. 추적 시 이전 영상에서의 블랍의 중심점과 현재 프레임의 블랍의 중심점을 식 2와 같이 유클리디안 거리를 이용하여 매칭한다.

$$D(O_t^i, O_{t-1}^j) = w[(x_t^i - x_{t-1}^j)^2 + (y_t^i - y_{t-1}^j)^2]^{\frac{1}{2}} \quad (2)$$

i, j : center of blobs, t : frame number

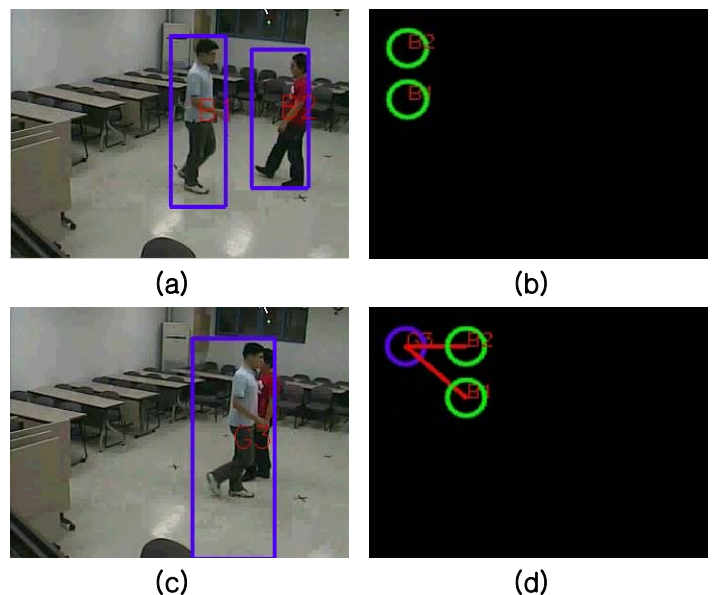


그림 2 블랍의 성격을 표현한 트리 구조
(a) 이미지 1 (b) 이미지 1의 블랍 구조
(c) 이미지 2 (d) 이미지 2의 블랍 구조



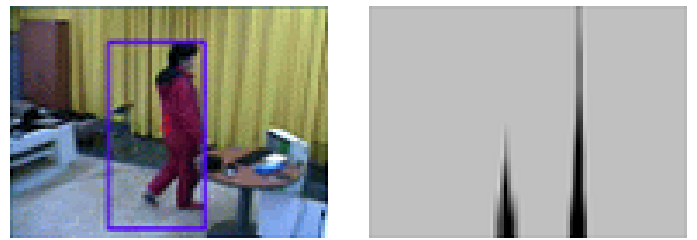
(a) (b) (c)
그림 3 MS 방법을 이용한 겹침 처리

겹쳐진 영역 때문에 대상 객체의 색상과 크기, 위치가 변할 수 있기 때문에 추적에 어려움이 발생한다. 이러한 겹침 문제를 해결하기 위해 크게 두 가지 방법[6]으로 연구되었다. 첫 번째 방법으로는 Merge-Split(MS) 방법으로 두 블랍의 겹침이 발생한 경우 하나의 블랍으로 새롭게 정의하는 방법이다. 이 때 겹치기 이전의 각 블랍들의 정보들을 새로운 블랍에 상속시키며 이러한 정보를 바탕으로 다시 분리가 발생할 경우 분리된 블랍과 겹쳐지기 이전의 블랍의 유사성을 판단할 수 있는 근거로 사용한다. 다른 방법은 Straight-Through(ST) 방법이다. 이 방법은 만약 겹침이 발생한 경우에도 두 개의 블랍들을 하나의 블랍으로 합치지 않고 각각의 블랍들의 영역을 유지하면서 추적을 계속한다. 본 논문에서는 다중 카메라에서의 객체 추적의 보완과 실시간 추적을 위해 MS 방법을 사용하였다. 그림 3은 두 사람의 추적 과정에서 겹침이 발생하여 하나의 블랍으로 인식한 후 다시 분리가 된 경우 겹치기 이전의 블랍과 이후의 블랍의 동일성이 유지되는 것을 보여 주고 있다.

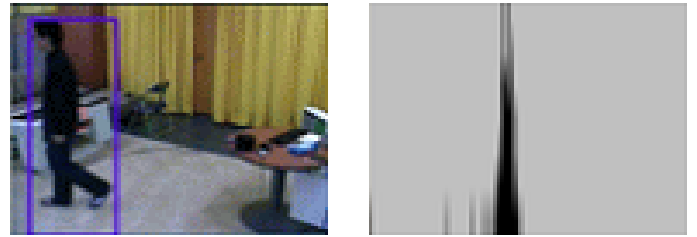
객체 추적에서 겹침 과정을 구분하기 위해서 사건을 출현, 퇴장, 결합, 분리로 정의하였다. 카메라는 전체 블랍의 수를 가지고 있다. 블랍의 수의 변화가 이벤트 감지의 요소가 될 수 있다. 우선 블랍의 숫자가 동일하고 사건이 발생하지 않은 경우에는 유클리디안 거리를 이용하여 블랍 간의 매칭을 수행한다.

출현이나 분리가 발생하면 블랍의 총 수가 증가된다. 출현이 발생한 경우는 블랍 매칭 수행 후 이전 프레임과 현재 프레임의 블랍의 중심점의 유클리디안 거리가 작은 순서대로 매칭을 시킨 후 남은 블랍을 새로운 블랍이라고 정의하고 새로운 추적자를 할당한다. 분리가 발생한 경우는 현재 프레임의 하나의 블랍 영역이 이전 프레임에서 두 개의 블랍 영역과 동시에 겹쳐지는 부분이 발생했을 경우이다. 이 때 이전의 블랍이 다른 블랍들을 포함하고 있는지 파악하고 다른 블랍들을 가지고 있는 경우에는 히스토그램에 따라 두 개의 블랍들에 다시 속성을 전달해 준다. 분리가 발생시 결합 이전의 블랍과 매칭을 하기 위해 색상 히스토그램[7]을 이용하였다. 그림 4는 분리가 발생할 경우 참조되는 히스토그램을 보이고 있다. 이 때 히스토그램의 x축은 색상(Hue)를 뜻하고 y축은 픽셀 비율을 나타낸다.

퇴장과 결합은 블랍의 총 수가 감소된 경우에



(a) (b)



(c) (d)

그림 4 겹침 처리하기 위해 사용된 색상 히스토그램

발생한다. 결합이 발생한 경우는 현재 프레임에서 하나의 블랍 영역이 이전 프레임의 블랍들의 영역과 겹칠 때이다. 이 때는 새로운 추적 블랍을 생성 시켜 주고 이전 프레임의 블랍들의 속성을 상속 시킨다. 결합이 발생되지 않고 퇴장이 발생한 경우에는 출현과 마찬가지로 이전 프레임과 현재 프레임의 블랍 위치점의 유클리디안 거리를 이용하여 가장 먼 거리에 있는 블랍을 삭제한다. 이 방법을 이용하여 단일 카메라 상에서 다수의 객체들이 겹치더라도 지속적인 추적이 가능하게 하였다.

4. 다중 카메라에서의 객체 추적

하나의 카메라를 통한 객체의 추적은 일반적으로 외형 정보인 색상이나 크기 같은 특성들을 이용한다. 그러나 이러한 특성들은 다른 객체나 배경의 겹침에 의해서 변하는 경우가 발생한다. 이를 보완하기 위해서 다중 카메라를 고려할 수 있다. 하나의 카메라로 해결될 수 없는 문제가 다른 카메라를 이용해 FOV를 확장시킬 수 있으며, 이를 이용하여 객체 추적을 보완할 수 있다. 다중 카메라에서 중요한 문제는 각각의 카메라 상에서 추적되고 있는 객체가 동일한 객체 인지 아닌지를 구분하는 것이다. 이러한 문제의 해결하기 위해 일반적으로 두 가지 방법이 연구되고 있다. 첫 번째는 기하학적 특성을 이용하여 카메라 간의 공간을 하나의 동일한 공간으로 변형시켜 대응되는 객체를 찾는 방법이며, 이러한 방법은 카메라 보정이 필요하다. 다른 하나는 카메라 상에서 보이는 이미지에서 같은 특성을 찾아서 대응시키는 방법이다. 본 논문에서는 외형 정보의 변화를 극복하기 위해 기하학적인 특성을 이용하였다.



(a) (b) (c)
그림 5 호모그래픽 매트릭스를 이용한 공간 투영

두 카메라간 중첩된 FOV를 가지고 있다는 조건에서 기하학적 위치 관계를 파악하기 위해 호모그래픽 매트릭스를 이용하였다. 호모그래픽 매트릭스를 얻어내기 위해서 객체가 움직이는 공간이 평면이라는 가정을 하였다. 그리고 각 카메라 간에 대응되는 바닥의 임의의 특징 좌표들을 미리 설정한 후 두 이미지에서 움직임이 발생하는 공간들의 관계 매트릭스를 식 3을 이용하여 계산 하였다[8]. 이러한 매트릭스에서 z값이 1이라고 가정을 하였고 회전 매트릭스와 이동 매트릭스를 고려하였다. 그림 5는 호모그래픽 매트릭스를 통해서 두 공간의 바닥 평면을 하나의 공간으로 표현한 것이다. 이를 통해 한 카메라에서 나타나는 블랍의 가장 아래 점이 평면에 위치한 점으로 가정하고 다른 카메라에서 대응되는 점으로 표현 가능하게 된다. 이러한 정보를 이용하여 FOV에서

나타난 객체가 전체 카메라 집합에서 새롭게 등장한 것인지 아니면 다른 카메라에서 추적이 되고 있는 객체인지 판단 할 수 있게 된다. 그림 6은 두 카메라에서 나타난 객체의 바닥 점을 각각 다른 카메라로 투영된 모습을 보여주고 있다.

하나의 카메라에서 찾은 바닥 위치 점을 다른 카메라의 바닥 위치 점으로 매칭이 가능하지만 노이즈로 인해 정확하게 같은 점을 나타내기 어렵다. 이를 해결하기 위해 식 4를 이용하였다. 카메라 C^i 에 객체 O 가 등장했다고 가정을 하고 S 는 C^i 에 투영되는 매트릭스를 가지고 있는 카메라의 집합이며, C^j 는 카메라 집합 S 에 포함되어 있고 C^i 에 속하고 C^i 의 대응되는 점들을 P_m^{ij} 라고 정의하고 M 은 대응되는 점들의 라벨이다. L 은 C^j 에서 보여지는 객체들의 집합이며 각각의 객체는 L_k 표현되고 k 는 객체의 라벨이다. 여기서 유클리디안 거리를 이용하여 가장 가까운 점을 찾아서 카메라 C^j 와 C^i 에서 나타나는 객체들을 매핑할 수 있다.



(a) (b)
그림 6 카메라간의 객체의 바닥점의 투영



(a) (b)



(c) (d)



(e) (f)

그림 7 다중 카메라 시스템에서의 객체 추적

$$\text{Label}(O) = \min_k D(L_k^j, P_m^{ij}) \quad (4)$$

5. 실험

실험을 위해 2.0GHz 펜티엄 4 프로세서와 1GB의 램을 가진 PC에서 개발하였으며, 두 대의 카메라를 한 대의 PC에 연결을 하고 320 x 240 사이즈의 이미지를 5 frame/sec로 처리하였다. 시스템 구현을 위해서 C++과 OpenCV 라이브러리[9]를 이용하였다.

그림 7는 두 대의 카메라가 중첩된 FOV 영역이 존재하고 두 명의 사람이 나타났을 때 이들을 추적하는 모습을 보이고 있다. (a), (c), (e)는 하나의 카메라 영상이며 (b), (d), (f)는 다른 하나의 카메라 영상이다. (a)와 (b)에서 두 명의 사람이 등장하였을 때, 하나의 카메라 시스템에서 먼저 블랍에 라벨을 부여하게 된다. 이때 나머지 카메라는 이미 존재하는 블랍이라고 판단하여 먼저 부여된 라벨을 전달 받는다. (c)와 (d)는 카메라가 설치된 위치와 각도에 의해서 하나의 카메라에서는 하나의 블랍으로 보이고 다른 하나의 카메라에서는 두 개의 블랍으로 보이게 되며, 바닥 점 매칭으로 하나의 블랍이 두 개의 블랍을 가졌다는 정보를 추가로 얻는다. (d)와 (e)는 하나의 시스템에서 조명 변화 혹은 잘못된 배경 생성에 의해 발생하는

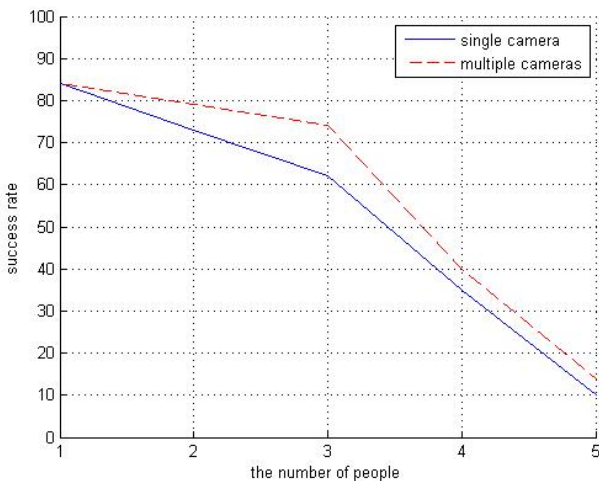


그림 8 추적 성능 비교

오류를 처리하는 장면이다. (d)에서 B1이 B7로 잘못 추적이 되었지만 다른 카메라의 정보를 가지고 (f)에서 B1으로 올바르게 수정이 되어서 지속적인 추적이 가능하게 되었다.

그림 8은 등장하는 사람의 수를 증가시켜 5분 동안 추적한 결과이다. 실험 결과에서 다중 카메라 시스템이 단일 카메라 시스템에 비해 추적 정확율이 더 높은 것을 알 수 있다. 또한, 사람이 증가하면 다중 카메라를 이용한 방법은 단일 카메라를 이용한 방법과 마찬가지로 추적 성공률이 낮아진다. 이것은 다른 하나의 카메라도 겹침이 동시에 발생할 확률이 높아지기 때문이다.

6. 결론

본 논문에서는 카메라의 FOV가 중첩되는 공간에서 사람들을 추적할 수 있도록 다중 카메라를 이용한 사람 추적 시스템을 개발하였다. 우선, 사람의 동일성을 파악하기 위해서 기하학적인 특성을 이용하여 영상을 분석하였으며, 이러한 분석 결과를 바탕으로 사람이 겹치거나 분리되는 경우에도 지속적으로 추적할 수 있도록 하였다.

향후 연구로는 3대 이상의 카메라를 이용하여 중첩된 FOV에 대한 분석 정보의 신뢰성에 따라 객체의 동일성을 좀 더 정확하게 파악할 수 있도록 알고리즘을 개선하고, 각 카메라로부터 얻은 영상 분석 결과가 불일치성이 발생했을 때에도 이러한 불일치를 보완할 수 있는 방법을 연구하는 것이다.

[참고문헌]

[1] Baumberg, A.M.: Learning Deformable Models for Tracking Human Motion. PhD thesis. School of Computer Studies, University of Leeds, Leeds, UK

(1995)

[2] T. Olson and F. Brill.: Moving object detection and event recognition algorithms for smart cameras. DARPA Image Understanding Workshop, pp. 159–175 (1997)

[3] I. Haritaoglu, D. Harwood, L. S. Davis.: W4: Real-Time Surveillance of People and Their Activities. IEEE Trans. Pattern Anal. Machine Intell, vol.22,pp.809–830 (2000)

[4] A. J. Lipton, H. Fujiyoshi, and R. S. Patil.: Moving object detection and event recognition algorithms for smart cameras. IEEE Workshop Applications of Computer Vision, pp. 8–14 (1998)

[5] Anurag Mittal, Larry S. Davis.: M2Tracker: A Multi-View Approach to Segmenting and Tracking People in a Cluttered Scene. IJCV, Vol. 51 (3) (2003)

[6] P. Gabriel, J. Verly, J. Piater, A. Genon.: The State of the Art in Multiple Object Tracking Under Occlusion in Video Sequences. Advanced Concepts for Intelligent Vision Systems, pp. 166–173 (2003)

[7] R. Bodor, B. Jackson, N.P. Papanikolopoulos.: Vision-Based Human Tracking and Activity Recognition. Proc. of the 11th Mediterranean Conf. on Control and Automation, Jun (2003)

[8] R. Hartley, A. Zisserman.: Multiple view Geometry in Computer Vision. Cambridge University Press, Cambridge, UK (2000)

[9] Intel. Open computer vision library. <http://sourceforge.net/projects/opencvlibrary/>