

사람의 움직임 추적에 의한 다중 카메라의 네트워크 위상 추론

남윤영⁰¹, 류정훈¹, 조용원¹, 최유주², 조위덕¹

¹아주대학교 유비쿼터스시스템연구센터

²서울벤처정보대학원대학교

{young022, ryujunghun, c62026}@gmail.com, yjchoi@suv.ac.kr, chowd@ajou.ac.kr

Inference of Multiple Cameras Network Topology by Tracking Human Movement

Yunyoung Nam¹, Junghun Ryu¹, Yongwon Cho¹, Yoo-Joo Choi², We-Duke Cho¹

¹Center of Excellence for Ubiquitous System, Ajou University

²Seoul University of Venture & Information

요 약

보안첨단화의 필요성 증대로 인하여 복합적이고 고기능의 보안 감시 시스템의 수요가 급속도로 확대되면서 보안은 안전하고 행복한 생활을 만드는데 없어서는 안될 중요한 역할을 하게 되었다. 최근, 디지털 영상기술의 급속한 발달과 보급은 이러한 보안 감시 시스템을 가능하도록 하였다. 본 논문은 다수의 카메라로부터 사람들의 움직임을 연속적으로 식별하고 추적할 수 있는 향상된 지능화 방법을 제안한다. 이 방법을 통해 카메라들 간의 위상이 자동으로 구성되고 객체의 움직임을 기반으로 학습하여 카메라들간의 거리, 객체와 카메라와의 거리, 카메라의 각도를 자동적으로 연산할 수 있도록 하였다. 이러한 자가 구성 단계 이후에 사람의 움직임을 추적하게 된다. 추적에서 사람들을 식별하는 단계가 선행되어야 하며, 이를 위해 머리, 몸, 손, 다리로 분리하여 각각의 정보들을 식별자로 사용하였다. 이러한 외형 식별자와 객체의 출몰간의 시간차를 이용해 다수의 카메라들로부터 객체를 연속적으로 추적하였다.

1. 서 론

현대사회에서 발생하는 안전사고나 범죄들로 인해 안전과 감시의 요구가 증대됨에 따라, 최근 주요 시설물의 감시나 우범지대의 치안유지를 목적으로 하는 자동화된 무인 감시 시스템의 개발이 진행되고 있다. 이러한 개발은 단순한 DVR(Digital Video Recorder)뿐만 아니라 이동 객체의 인식(detecting)과 추적(tracking)을 수행하는 지능적 감시 시스템 개발도 실용적이고도 경제적인 대안으로써 병행되고 있다. 이러한 지능적 감시 시스템의 기술은 영상 시퀀스에서 움직이는 물체를 탐지하고 스스로 판별할 수 있는 기술으로써 다수의 감시 카메라의 환경에서 자동적인 객체 식별 및 추적기술은 필수적인 기능이라 할 수 있다.

카메라는 구성에 있어서 단일 카메라와 다중 카메라로 구분할 수 있으며, 다중 카메라는 영상의

겹침(overlapping)에 따라 겹침 카메라와 비겹침(non-overlapping) 카메라로 구분할 수 있다. 단일 카메라는 과속단속 카메라와 같이 특정한 공간 영역에서 정의된 기능을 수행하는데 주로 사용되며, 다중 카메라는 실내외의 넓은 영역에서 안전과 감시를 위해 사용된다. 겹침 카메라는 2대 이상의 카메라가 일정 영역을 공유함으로써 영상을 자세하게 분석하거나 스테레오 비전기술을 통해 3차원 영상을 제작하는데 사용된다. 그러나 카메라의 설치와 유지비용의 이유로 넓은 영역을 감시하는 시스템의 경우 비겹침 카메라 구성이 대부분이며, 이러한 비겹침 카메라에서 객체를 추적하기 위해서는 객체의 외형적인 식별자와 객체의 출몰의 시간간격을 고려해야 한다.

본 논문에서는 객체의 움직임을 연속적으로 추적하기 위해 카메라들간의 관계와 거리를 계산하여 공간위상(spatial topology)을 자동으로 구성하도록 하였으며, 이러한 카메라 위상과 객체의 출몰 시간간격을 바탕으로 동일한 객체를 연속적으로 추적하도록 하였다.

* 본 연구는 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 정보통신부의 유비쿼터스컴퓨팅및네트워크원천기술개발사업의 지원에 의한 것임.

동일한 객체를 판별하기 위해서, 단일 카메라로부터 얻어진 객체 영상데이터를 분석하여 객체의 신체를 머리, 몸, 손, 다리로 나누어 머리색, 모자의 착용유무, 옷 색상을 객체의 식별자로 사용하였고, 이러한 객체의 외형정보와 객체가 카메라 영상에서 사라지고 다시 나타나는 시간을 계속적으로 관찰하고 학습함으로써 카메라의 공간 위상(spatial topology)을 자동으로 구성하게 된다.

본 논문의 구성은 다음과 같다. 2절에서는 다중 카메라에서의 객체 추적과 관련된 연구를 소개하고, 3절에서는 전체적인 시스템의 구조에 대해서 설명한다. 4장에서는 객체를 식별할 수 있도록 신체의 부분별로 나누는 방법에 대해서 서술한다. 5절에서는 카메라들간의 자가 구성(Self Configuration)을 하는 방법에 대해 기술하고, 6장에서는 제안한 방법을 이용하여 실제 건물 내에서 실험한 결과를 보이고, 마지막 7장에서 결론을 맺는다.

2. 관련연구

초기 추적 알고리즘에서는 camera calibration과 overlapping FOV(Fields Of View)를 사용해서 여러 대에 카메라에서 객체들을 추적하였다. Cai와 Aggarwal[1]은 보정된 카메라를 이용해서 뷰어를 서로 겹치게 구성한 후 카메라들간에 추적된 객체들의 핸드오버(handover)를 계산하였다.

Hung과 Rusell[2]는 카메라 뷰어가 서로 겹치지 않도록 구성한 다음 객체들의 전이시간(transition time)과 외형을 매칭 하는 방식을 이용했다. Kettner와 Zabith[3]은 카메라 뷰어가 겹치지 않는 환경에서 객체들을 추적하는 Bayesian solution을 제안했다. 이 방법에서도 객체들의 전이시간을 이용하였으며 사람들의 움직임 방식을 알고 있어야 하는 한계가 있다.

Dick and Brooks[4]는 두 카메라들로부터 객체의 행동 패턴을 기술하기 위해 확률적 전이 행렬(stochastic transition matrix)을 이용했으며, 모든 카메라 사이에 일관성을 확인하기 위해 학습된 데이터를 필요로 한다.

Ellis et. al[5]는 학습된 데이터를 가지고 일관성을 확인하는 방법 대신에 재등장 기간(period)을 구성한 후 주어진 시간 내 관찰된 객체들을 이용하였다. Ellis는 재등장 시간을 히스토그램(histogram)으로 표현하기 위해 관찰된 객체들을 이용하였고 이를 바탕으로 카메라들간의 링크(link)를 찾았다.

Andrew et. al[6]은 카메라들간에 시-공간 링크들의 사후 확률 분포(posterior probability distributions)와 칼라 변화(color variations)를 모델링하기 위해 점진적인 학습을 이용하였다. 겹치지 않는 다중 카메라 구성에서 객체의 일치성을 점진적인 칼라의 유사성

학습과정을 토대로 검증을 하였다. CCCM(Consensus-Color Conversion of Munsell color space)을 이용해서 칼라의 유사성을 판단한 후 카메라 사이의 링크를 결정하게 된다. 링크는 주어진 시간동안 객체의 등장과 재등장을 조건부 전이 확률(conditional transition probability)로 계산해서 결정한다. 좀 더 정확하게 링크를 결정하기 위해 카메라 화면내의 블록을 최대한 쪼개어서 필요 없는 링크는 제거하고 유효한 링크만 선별을 한다. 링크 선별 작업후 각 링크사이에 관찰된 객체들의 일치성을 더 정확하게 판단하기 위해 사후 확률 분포를 이용하였다.

Javed O et. al[7]은 실시간 상황에서 카메라 뷰어가 서로 겹치지 않는 환경에서 객체의 일치성을 확인하기 위해 FOV라인 사이의 관계를 이용했다. 서로 겹침이 없는 카메라의 FOV라인을 연장시켜 가상 라인(virtual line)을 만든 다음 예측된 객체의 위치와 가상 라인과의 최소거리를 구해서 동일한 객체인지를 식별하게 된다.

본 논문에서는 카메라들 간의 거리, 객체와 카메라와의 거리, 카메라의 각도를 자동적으로 연산해서 자가 구성 단계를 거친 후 사람의 움직임을 추적 하게 된다. 추적 단계에서 외형적인 식별자와 객체의 출몰방향과 시간차를 이용해 다수의 카메라들로부터 객체를 연속적으로 추적한다.

3. 시스템 구조

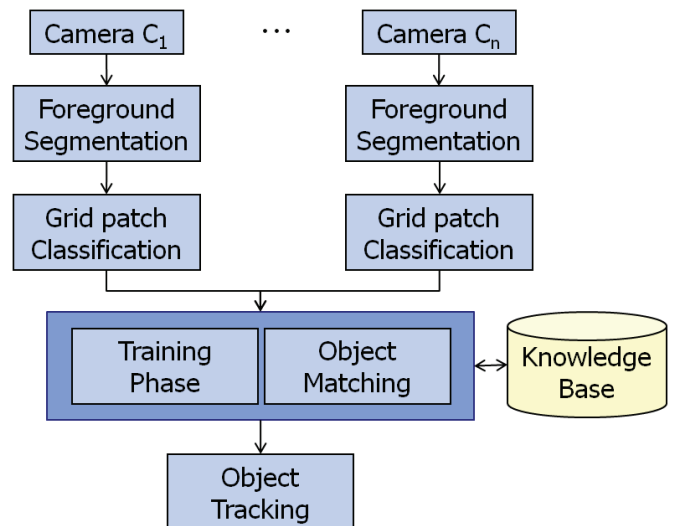


그림 1 시스템 구조

그림 1은 전체적인 시스템 구조를 보이고 있다. N개의 카메라로부터 입력받은 영상으로부터 전경이미지(foreground)를 추출한다. 전경이미지를 추출하기 위해 색상, 에지, 모션, 디스페리티(disparity) 정보를 복합적으로 이용한 강건한 세그멘테이션을 사용하였다. 다음으로 현재 장면의 smooth disparity를 계산하고 색상, 에지, 모션, 디스페리티를 비교하여

정적 배경 장면에서 움직이는 대상을 분리하였다. 관심 있는 대상물만 분리하기 위해 위 대상물이 제한된 디스퍼티 범위 즉 깊이를 가지고 있고 위 대상물의 디스퍼티가 부드럽게 변한다고 가정하여 깊이 정보를 포함하고 있는 분리된 대상을 추적하였다. 이렇게 추출된 전경이미지는 그리드 패치 분류(grid patch classification) 단계를 거치면서 불랍(blob)형태의 사람의 모습을 각 부분으로 분리하게 된다.

위의 과정을 통해 추출된 정보를 바탕으로 각 카메라로부터 나타나는 객체들의 외형정보를 비교하여 동일한 객체인지 판별하고 이동경로, 등장시간, 재등장 시간을 데이터베이스에 저장한다. 일정한 기간 동안의 학습과정을 거치면 카메라 연결관계를 구성하고 이를 바탕으로 객체를 추적하게 된다.

4. 객체 식별 및 추적

본 논문에서는 좀더 정확한 객체 식별을 위해서 객체의 신체부분별로 분류하였다. 부분별로 분류하는데 카메라와 객체간의 거리에 따라 머리와 손 부분을 추출하기 위해 그리드 기반으로 처리하였다.

우선, foreground 픽셀과 background 픽셀을 분류하기 위해 background subtraction을 적용했다. 다른 조명에서 촬영한 50개의 frame을 이용한 HSI(Hue, Saturation, Intensity)칼라 이미지에서 일정한 임계값(식 1)보다 subtraction값이 클 경우 foreground pixel로 판단하게 되고 그렇지 않은 경우에는 background pixel로 판단하게 된다.

$$R(P_{ij}) = \begin{cases} 1(\text{foreground}), & \text{if } |H_{ij} - H_{b_j}| > \omega_1 \sigma(H_{b_j}) \\ & \text{and } |S_{ij} - S_{b_j}| > \omega_2 \sigma(S_{b_j}), \\ 0(\text{background}), & \text{otherwise} \end{cases} \quad (1)$$

H_{ij} : Hue, S_{ij} : saturation, P_{ij} : currentpixel
 H_{b_j} : average Hue, S_{b_j} : average Saturation
 $\sigma(H_{b_j})$: Hue variance, $\sigma(S_{b_j})$: Saturation variance

background subtraction을 통해 얻은 결과 이미지에 두 개의 patch feature값(식 2)을 적용시켜 세 개의 grid patch(foreground patch, skin patch, background patch)(식 3)로 구분한다.

$$F_{ij} = \frac{\# \text{ of foreground pixels in Patch}(i, j)}{64} \quad (2)$$

$$S_{ij} = \frac{\# \text{ of skin pixels in Patch}(i, j)}{\# \text{ of foreground pixels in Patch}(i, j)}$$

F_{ij} : 전체 $8 \times 8 \text{ pixels}$ 에서 foreground pixels의 비율
 S_{ij} : foreground pixels에서 skin pixels의 비율

임계값이 고정되었을 경우 객체와 카메라 사이의 거리가 멀어지면 foreground 픽셀, background 픽셀, skin 픽셀을 구분하기가 쉽지 않기 때문에 사용자가 원하는 부분(ROI-손과 머리 부분)을 정확하게 얻을 수 없게 된다. 이런 경우 거리에 따라 다르게 적용시킨 임계값(식 4)을 이용해서 원하는 부분을 추출하게 된다.

$$G_{ij} = \begin{cases} \text{foreground patch,} & \text{if } F_{ij} \geq X_{fg} \text{ and } S_{ij} < X_{sk} \\ \text{skin patch,} & \text{if } F_{ij} \geq X_{fg} \text{ and } S_{ij} \geq X_{sk} \\ \text{background patch,} & \text{otherwise} \end{cases} \quad (3)$$

$$X_{fg} = \arg \min(f_{fg}(x)), f_{fg}(x) > (1 - P_{fg})$$

$$X_{sk} = \arg \min(f_{sk}(x)), f_{sk}(x) > (1 - P_{sk}) \quad (4)$$

where $f_{fg}(x) = \sum_{i=0}^x H_{fg}(i), f_{sk}(x) = \sum_{i=0}^x H_{sk}(i)$

아래 그림은 이러한 적응적인 그리드 기반의 방법을 사용하여 객체의 부분을 추출한 결과를 보이고 있다.

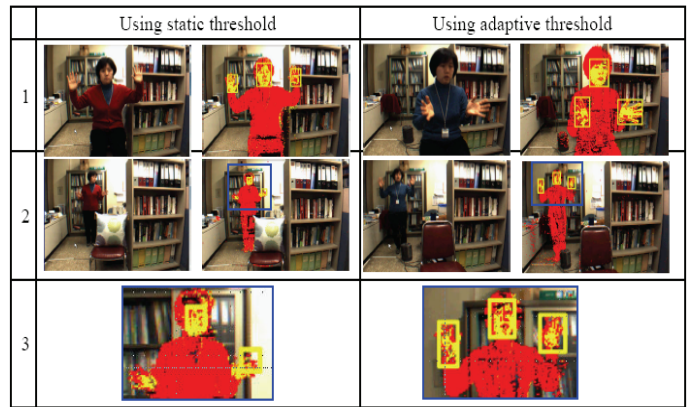


그림 2 그리드 기반의 객체 부분 추출

5. 카메라 네트워크 위상 학습 (Camera Network Topology Learning)

카메라들간의 공간적 위상을 학습하기 위해서 객체의 외형정보(appearance)와 객체의 출몰시간을 이용한다. 외형정보는 4장에서 서술하였던 객체의 머리, 몸의 색상정보를 이용한다. 특히, 모자를 착용하였거나 신체의 사이즈도 외형정보에 포함이 된다. 객체의 출몰시간은 등장 시간(entry time), 퇴장 시간(exit time), 재등장 시간(reentry time)으로 구분하여 퇴장시간과 재등장 시간 간의 시간이 임계값(threshold) 이내라면 카메라간의 링크가 존재하는 것으로 가정하고 연결을

구성한다.

이러한 카메라의 시공간적 위상은 가중치 그래프(weighted graph)를 확장하여 표현할 수 있다. 위상 그래프 G 는 노드(node)와 에지(edge)로 구성이 된다. 노드는 카메라 노드, 겹침노드, 비겹침 노드, 가상 노드로 구분되며 에지는 일반에지와 겹침에지, 비겹침에지로 구분한다.

$$G = \langle V, E \rangle$$

$$weight(G_k) = \begin{cases} |entrytime - exittime| \\ |exittime - reentrytime| \end{cases} \quad (5)$$

$V = camera\ node, overlapping\ node,$
 $non-overlapping\ node, virtual\ node$
 $E = edge, overlapping\ edge, non-overlapping\ edge$

객체의 등장 시간과 퇴장 시간의 차를 구하여 카메라에서 객체가 움직인 시간으로 간주하고 이동시간을 노드들간의 에지의 가중치로 표현하였으며 객체가 등장하고 퇴장하는 방향정보를 이용하여 노드들간의 링크를 구성하였다. 또한 겹침에지의 경우 객체의 움직임이 두 개 이상의 카메라에서 나타나기 때문에 객체의 이동방향을 분석하여 카메라들간의 각도를 추론할 수 있다. 건물내의 복도나 건물외의 차량이 이동하는 도로의 경우, 사람이나 차량의 움직임은 패턴과 경로가 일정하기 때문에 이러한 추론이 가능하다. 이러한 시공간적 위상은 일정시간동안의 학습을 통해 자동적으로 구성이 된다.

6. 실험

실험을 위하여 인텔 듀얼 코어 2.4GHz CPU와 1GB 메모리 사양의 Windows XP 환경에서 C++와 OpenCV[8]를 이용하여 구현하였으며, 설치된 Samsung Dome Camera SID-45C에서 획득한 영상(320×240)을 초당 15프레임으로 처리하였다.

실험 환경은 그림 3과 같은 레이아웃으로 된 빌딩 내부에 5개의 카메라가 설치되어 있고 사용자가 빌딩 내부를 이동함에 따라 카메라의 FOV에서 entry node와 exit node를 찾고 카메라 간의 edge를 분석하였다.

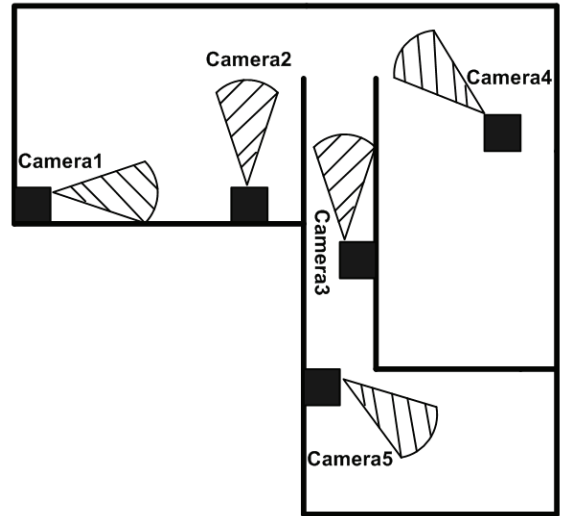


그림 3 카메라의 레이아웃

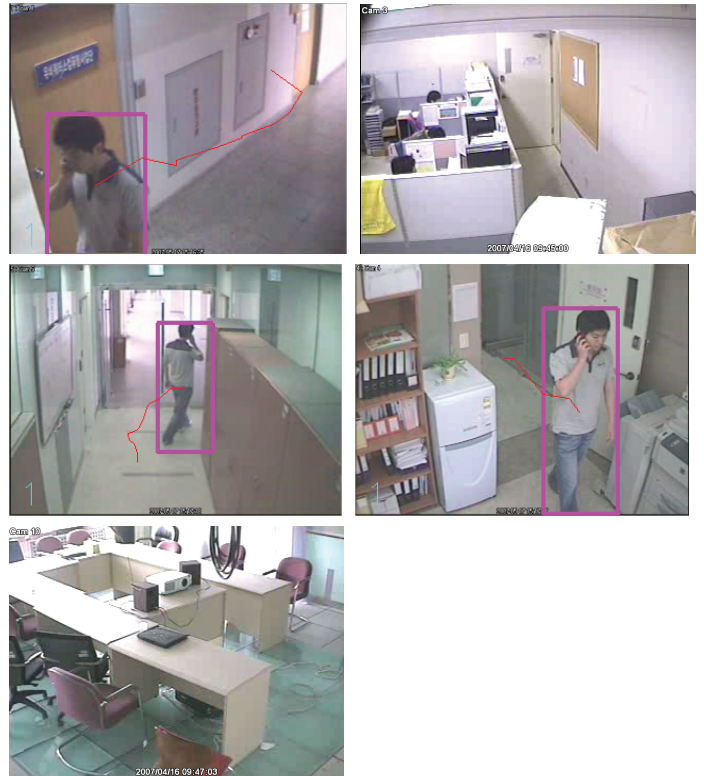


그림 4 다중 카메라에 나타난 객체의 식별 및 추적

그림 5는 위의 영상으로부터 객체의 움직임을 추적하여 얻은 카메라간의 위상정보와 객체의 이동시간을 표현한 가중치 그래프이다. 이 그래프를 얻기 위해 20분의 학습과 10분의 실험을 수행하였다.

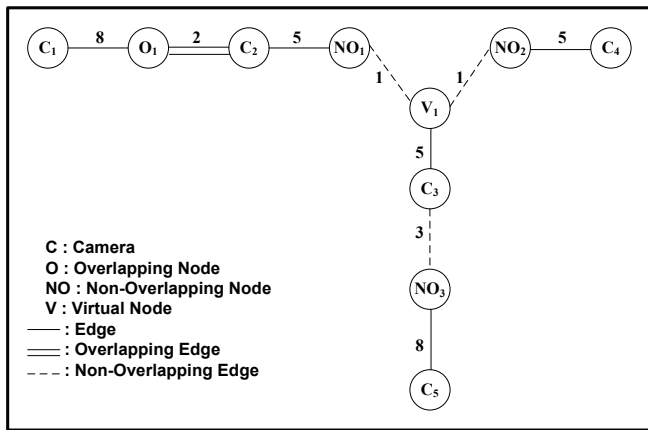


그림 5 가중치 그래프로 나타낸 카메라들의 위상

O₁과 C₂간의 에지는 겹침 지역이며 NO₁과 V₁, NO₂와 V₁, NO₃와 C₃간의 에지는 비 겹침 지역이다. 각 에지(edge)는 객체가 등장하여 퇴장할 때까지의 시간차를 나타내고 있으며, V₁은 비 겹침지역과 카메라와의 연결을 위해 가상으로 생성한 노드이다.

6. 결론 및 향후 연구 과제

본 논문에서는 다중 카메라로부터 객체를 추적하기 위해 카메라들간의 위상과 객체의 등장, 퇴장, 재등장 시간을 가중치 그래프로 표현하였으며, 객체를 좀더 정확하게 식별할 수 있도록 신체 부분별로 분리하고 카메라와 객체간의 거리를 이용한 적응적인 그리드 패치 방법을 사용하여 추출의 정확도를 높였다.

이러한 다중카메라에서의 객체 추적 기술을 이용하여 보안이 필요한 빌딩이나 아파트 단지과 같은 거주지내의 안전관리를 위한 홈 네트워크 분야에서도 폭 넓게 응용될 수 있을 것이다.

향후 연구로는 가중치 그래프를 확장하여 좀 더 정확하게 표현할 수 있는 3차원 형태의 그래프를 개발하고 겹침지역에서의 객체의 움직임을 이용한 카메라간의 좀더 정확한 각도 계산과 싱글 레이어에서 멀티 레이어로의 확장을 해 보는 것이다.

[참고문헌]

[1] Cai, Q., Agrarian, J. "Tracking Human Motion using Multiple Cameras". International Conference on Pattern Recognition 67-72 (1996)
 [2] Huang, T., Russell, S. "Object Identification in a Bayesian Context". International Joint Conference on Artificial Intelligence (IJCAI-97), Nagoya, Japan 1276-1283 (1997)
 [3] Kettner, V., Zabih, R. "Bayesian Multi-Camera

Surveillance". IEEE Computer Vision and Pattern Recognition 253-259 (1999)
 [4] Dick, A., Brooks, M. "A Stochastic Approach to Tracking Objects Across Multiple Cameras". Australian Conference on Artificial Intelligence 160-170 (2004)
 [5] Ellis, T., Makris, D., Black, J. "Learning a Multi-Camera Topology". Joint IEEE Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS) 165-171 (2003)
 [6] Gilbert, A., Bowden, R. "Tracking objects across cameras by incrementally learning inter-camera colour calibration and patterns of activity". European Conference on Computer Vision (ECCV) (2006)
 [7] Javed, O., Rasheed, Z., Alatas, O., Shah, M., "A real time surveillance system for multiple overlapping and non-overlapping cameras". Multimedia and Expo, ICME '03. Proceedings. International Conference on 649-652 (2003)
 [8] [8] Open Source Computer Vision Library. <http://www.intel.com/research/mrl/research/opencv/>