



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0078098
(43) 공개일자 2020년07월01일

(51) 국제특허분류(Int. Cl.)
G06T 7/292 (2017.01) G06T 7/11 (2017.01)
(52) CPC특허분류
G06T 7/292 (2017.01)
G06T 7/11 (2017.01)
(21) 출원번호 10-2018-0167679
(22) 출원일자 2018년12월21일
심사청구일자 2018년12월21일

(71) 출원인
인천대학교 산학협력단
인천광역시 연수구 아카데미로 119 (송도동)
(72) 발명자
전광길
인천광역시 연수구 경원대로119번길 21, 114동
904호(동춘동, 연수2차풍림아파트)
(74) 대리인
특허법인충정

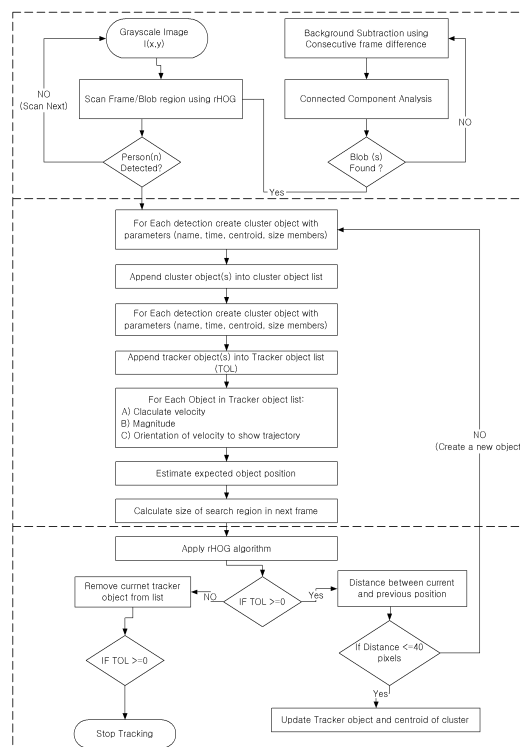
전체 청구항 수 : 총 6 항

(54) 발명의 명칭 오버헤드 카메라를 이용한 사람 추적 방법

(57) 요약

본 발명은 사람 추적 장치를 개시하여, 영상 내에 존재하는 사람을 검출하고, 상기 검출된 사람이 위치한 클러스터를 반환하는 단계, 상기 클러스터 각각에 대하여 하기 (수학식 1)로 표현되는 클러스터 객체(CO)를 생성하는 단계, 각 클러스터 객체(CO)마다 하기 (수학식 2)으로 표현되는 추적 객체(TO)를 생성하는 단계, 상기 추적 (뒷면에 계속)

대표도 - 도2



객체(TO)를 통하여 하기 (수학식 3)으로 표현되는 추적 오브젝터 목록(TOL)을 생성하는 단계, 상기 추적 오브젝터 목록(TOL)에서 다음 프레임의 동일한 영역 내 추적 객체를 선택하여, 상기 다음 프레임에서 선택된 추적 객체의 위치를 추정하는 단계, 상기 다음 프레임의 탐색 영역(S_g)을 하기 (수학식 4)와 같이 계산하는 단계 및 상기 다음 프레임의 탐색 영역 내에 존재하는 사람을 검출하는 단계를 포함한다.

본 발명은 정지된 사람을 오랜 시간 동안 탐지하는 것이 가능하고, 가려진 것을 처리하는 것, 환경의 급격한 변화를 가져오는 것 등을 보정하여 추적을 효과적이고 정확하게 수행할 수 있다.

명세서

청구범위

청구항 1

영상 내에 존재하는 사람을 검출하고, 상기 검출된 사람이 위치한 클러스터를 반환하는 단계;

상기 클러스터 각각에 대하여 하기 (수학식 1)로 표현되는 클러스터 객체(CO)를 생성하는 단계;

(수학식 1)

$$CO = \{N, t, CC(x, y), S, P\}$$

(여기서 N 은 캡처된 이미지, t 는 이미지의 캡처 시간, P 는 형성된 클러스터가 검출된 위치, $CC(x, y)$ 는

클러스터의 중심(centroid)으로 $CC(x, y) = \frac{1}{n} \sum_{i=1}^n P$ 으로 표현되며, S 는 클러스터 구성원의 합계임.)

각 클러스터 객체(CO)마다 하기 (수학식 2)으로 표현되는 추적 객체(TO)를 생성하는 단계;

(수학식 2)

$$TO = \{Names[], Positions[], times[], CO, TP\}$$

(여기서 $Names$ 는 이미지의 이름, $Positions$ 은 사람들이 검출된 위치, $times$ 는 프레임이 기록된 시간, CO 는 클러스터 객체, TP 는 추적기위 위치임.)

상기 추적 객체(TO)를 통하여 하기 (수학식 3)으로 표현되는 추적 오브젝트 목록(TOL)을 생성하는 단계;

(수학식 3)

$$TOL = \{TO_1, TO_2, TO_3, \dots, TO_n\}$$

(여기서 TO_n 은 n 번째 추적 객체이고, n 은 자연수임.)

상기 추적 오브젝트 목록(TOL)에서 다음 프레임의 동일한 영역 내 추적 객체를 선택하여, 상기 다음 프레임에서 선택된 추적 객체의 위치를 추정하는 단계;

상기 다음 프레임의 탐색 영역(S_g)을 하기 (수학식 4)와 같이 계산하는 단계; 및

(수학식 4)

$$S_g = PW + (\Delta S)$$

(여기서 PW 는 32 by 32 pixels, ΔS 는 추적 객체의 위치 변화임.)

상기 다음 프레임의 탐색 영역 내에 존재하는 사람을 검출하는 단계;

를 포함하는 사람 추적 방법.

청구항 2

제1항에 있어서,

상기 클러스터 객체를 생성하는 단계 이후,

상기 클러스터 객체를 통하여 하기 (수학식 5)로 표현되는 클러스터 오브젝터 목록(COL)을 생성하는 단계를 더 포함하는 것을 특징으로 하는 사람 추적 방법.

(수학식 5)

$$COL = \{CO_1, CO_2, CO_3, \dots, CO_n\}$$

(여기서 CO_n 은 n 번째 클러스터 객체이고, n 은 자연수임.)

청구항 3

제1항에 있어서,

상기 추적 객체의 위치는 하기 (수학식 6)를 통해 계산되는 추적 객체의 속도(v) 및 속력($|v|$)을 기초로 추정되는 것을 특징으로 하는 사람 추적 방법.

(수학식 6)

$$\begin{cases} v = \frac{\Delta S}{\Delta t}, & \text{if cluster size} = 2 \\ v = \frac{1}{n} \sum_{k=1}^n \frac{\Delta S(k-1)}{\Delta t(k-1)}, & \text{if cluster size} > 2 \end{cases}$$

$$|v| = \sqrt{(CCx_2 - CCx_1)^2 + (CCy_2 - CCy_1)^2}$$

청구항 4

제3항에 있어서,

상기 추적 객체의 위치는 하기 (수학식 7)으로 추정되는 것을 특징으로 하는 사람 추적 방법.

(수학식 7)

$$Px(x, y) = Pc(x, y) + v * (\Delta t)$$

(여기서 $Px(x, y)$ 는 추적 객체의 예상 위치, $Pc(x, y)$ 는 추적 객체의 현재 위치임.)

청구항 5

제1항에 있어서,

상기 다음 프레임의 탐색 영역내에 존재하는 사람을 검출하는 단계 이후,

(a) 사람이 검출되지 않는 경우, 해당 추적 객체를 제거하고,

(b) 사람이 검출되는 경우, 이전의 위치와 현재 위치의 거리를 비교하여, (b-1) 상기 거리가 40 픽셀 이하라면 추적 객체 및 클러스터의 중심을 업데이트하고, (b-2) 상기 거리가 40 픽셀보다 크다면 새로운 추적 객체를 생성하는 단계를 더 포함하는 것을 특징으로 하는 사람 추적 방법.

청구항 6

제1항에 있어서,

상기 사람을 검출하는 과정은,

하기 (수학식 8)을 통하여 모션맵 이미지를 얻는 단계;

(수학식 8)

$$M_t = \text{abs}(I_{t+1} - I_t)$$

(여기서 M_t 는 모션맵 이미지, I_t , I_{t+1} 는 연속된 프레임임)

상기 모션맵 이미지를 8×8 픽셀의 하위 영역들로 나누고, 상기 하위 영역들의 픽셀 강도 값을 합산하는 단계; 및

하기 (수학식 9)를 통하여 사람을 검출하는 단계;

(수학식 9)

$$\begin{cases} \text{Blob,} & \text{if } B(x, y) > \tau \\ \text{No Blob,} & \text{if } B(x, y) \leq \tau \end{cases}$$

(여기서 $B(x, y)$ 는 상기 하위 영역들의 합산된 픽셀 강도 값, τ 는 임계값임.)

로 수행되는 것을 특징으로 하는 사람 추적 방법.

발명의 설명

기술 분야

[0001] 본 발명은 오버헤드 카메라를 이용한 사람 추적 방법에 관한 것으로, 더욱 상세하게는 이미지와 비디오를 분석하여 사람을 추적 및 식별하는 방법에 관한 것이다.

배경 기술

[0003] 현재의 비디오 분석 기술은 연구 및 기술 개발을 통해 사물 인터넷을 빠르게 변화시키고 있으며, 이에 따라 새로운 기회를 창출하고 있다. 비디오 분석에서 얻어지는 이미지 및 비디오에 기계 학습 알고리즘을 사용하여 카메라가 사람 및 기타 움직이는 물체를 자동으로 인식 할 수 있도록 하는 것은 사물인터넷(IoT)의 응용 분야 중 하나라고 할 수 있다.

[0004] 이러한 사물인터넷(IoT)과 컴퓨터에서의 연구 분야 중 하나는 이미지와 비디오에서 사람들을 추적하고 식별하는 기술이다. 사람을 추적하는 기술은 운전 보조 시스템, 사람의 걷기 행동 분석 및 감시 시스템 등과 같은 영역에 적용될 수 있다.

[0005] 한편, 산업 현장에서 근무하는 근로자의 다양한 형세를 관찰하여 실제 환경에서의 작업 흐름을 분석하고자 하는 경우 사람을 식별 및 추적하는 기술이 이용될 수 있다. 이 경우, 사이드/노말 뷰(side/normal view) 또는 오버헤드 뷰(overhead view) 카메라가 이용될 수 있는데, 일반적으로 오버헤드 뷰 카메라가 보다 정확하게 움직임을 추적할 수 있다.

[0006] 도 1의 (a)는 사이드/노말 뷰(side/normal view)를 통해 얻은 사람들의 이미지이고, 도 1의 (b)는 오버헤드 뷰(overhead view)를 통해 얻은 사람들의 이미지이다. 도 1 (a)와 (b)는 서로 다른 두 대의 카메라를 통해 동일한 장면을 동시에 촬영한 이미지이다.

[0007] 도 1의 (a), (b)에서 사람들은 흰 타원으로 표시되었다. 도 1의 (a)를 참조하면, 사람들이 기계, 전선 및 선반

부분 등 다양한 장비로 가려진 것을 확인할 수 있다. 사이드/노말 뷰(side/normal view)를 이용하는 경우, 사람들의 움직임을 완전히 파악할 수 없는데, 예를 들어, 큰 기계 뒤에서 일하는 경우에는 사람을 식별하기 어렵다. 이 경우, 사람들을 식별하고 움직임을 파악하는 것이 매우 어렵다.

[0008] 도 1의 (b)는 넓은 각도의 렌즈가 구비된 오버헤드 뷰(overhad view, 또는 top view) 카메라를 사용하여 촬영된 이미지를 도시한 것이다. 도 1의 (b)를 참조하면, 넓은 각도의 렌즈가 구비된 오버헤드 카메라가 사용되어 기계 등에 의해 가려져 있는 사람들의 이미지가 명확히 파악되는 것을 확인할 수 있다. 이와 같이, 오버헤드 뷰 카메라가 사용되는 경우, 가려지거나 어수선한 환경에서도 객체를 보다 명확하고 분명하게 투사할 수 있다.

[0009] 한편, 사람을 추적하는 과정을 어렵게 하는 요인으로는 이미지의 노이즈, 물체 모양의 변화, 물체의 급작스러운 움직임, 카메라의 동작 등이 있다. 종래의 사람 추적 방법은 상기의 문제점을 완전히 해결하지 못해 정확도가 매우 떨어진다는 문제점이 있었다.

발명의 내용

해결하려는 과제

[0011] 상기의 문제점을 해결하고자 본 발명은 오버헤드 카메라를 이용하여 높은 정확도가 나타나는 사람 추적 방법을 제공하는 것을 목적으로 한다.

과제의 해결 수단

[0013] 상기 기술적 과제를 해결하기 위하여 본 발명은 사람 추적 장치를 개시하여, 영상 내에 존재하는 사람을 검출하고, 상기 검출된 사람이 위치한 클러스터를 반환하는 단계, 상기 클러스터 각각에 대하여 하기 (수학식 1)로 표현되는 클러스터 객체(CO)를 생성하는 단계, 각 클러스터 객체(CO)마다 하기 (수학식 2)으로 표현되는 추적 객체(TO)를 생성하는 단계, 상기 추적 객체(TO)를 통하여 하기 (수학식 3)으로 표현되는 추적 오브젝터 목록(TOL)을 생성하는 단계, 상기 추적 오브젝터 목록(TOL)에서 다음 프레임의 동일한 영역 내 추적 객체를 선택하여, 상기 다음 프레임에서 선택된 추적 객체의 위치를 추정하는 단계, 상기 다음 프레임의 탐색 영역(S_g)을 하기 (수학식 4)와 같이 계산하는 단계 및 상기 다음 프레임의 탐색 영역 내에 존재하는 사람을 검출하는 단계를 포함한다.

[0014] 또한 상기 클러스터 객체를 생성하는 단계 이후, 상기 클러스터 객체를 통하여 하기 (수학식 5)로 표현되는 클러스터 오브젝터 목록(COL)을 생성하는 단계를 더 포함하는 것을 특징으로 한다.

[0015] 또한 상기 추적 객체의 위치는 하기 (수학식 6)를 통해 계산되는 추적 객체의 속도(v) 및 속력($|v|$)을 기초로 추정되는 것을 특징으로 한다.

[0016] 또한 상기 추적 객체의 위치는 하기 (수학식 7)으로 추정되는 것을 특징으로 한다.

[0017] 또한 상기 다음 프레임의 탐색 영역내에 존재하는 사람을 검출하는 단계 이후 (a) 사람이 검출되지 않는 경우, 해당 추적 객체를 제거하고, (b) 사람이 검출되는 경우, 이전의 위치와 현재 위치의 거리를 비교하여, (b-1) 상기 거리가 40 픽셀 이하라면 추적 객체 및 클러스터의 중심을 업데이트하고, (b-2) 상기 거리가 40 픽셀보다 크다면 새로운 추적 객체를 생성하는 단계를 더 포함하는 것을 특징으로 한다.

[0018] 또한 상기 사람을 검출하는 과정은, 하기 (수학식 8)을 통하여 모션맵 이미지를 얻는 단계, 상기 모션맵 이미지를 8×8 픽셀의 하위 영역들로 나누고, 상기 하위 영역들의 픽셀 강도 값을 합산하는 단계 및 하기 (수학식 9)를 통하여 사람을 검출하는 단계로 수행되는 것을 특징으로 한다.

발명의 효과

- [0020] 본 발명의 실시 예들에 따른 오버헤드 카메라를 이용한 사람 추적 방법의 효과에 대해 설명하면 다음과 같다.
- [0021] 본 발명은 종래의 방법의 정확도(48 %)에 비하여, 높은 정확도(약 99 %)가 나타날 수 있다.
- [0022] 또한 본 발명은 종래의 방법과는 달리, 추가적으로 사람이 존재 유무까지 정확하게 식별할 수 있다.
- [0023] 또한 본 발명은 정지된 사람을 오랜 시간 동안 탐지하는 것이 가능하고, 가려진 것을 처리하는 것, 환경의 급격한 변화를 가져오는 것등을 보정하여 추적을 효과적이고 정확하게 수행할 수 있다.
- [0024] 다만, 본 발명의 실시 예들에 따른 오버헤드 카메라를 이용한 사람 추적 방법이 달성할 수 있는 효과는 이상에서 언급한 것들로 제한되지 않으며, 언급하지 않은 또 다른 효과들은 아래의 기재로부터 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에게 명확하게 이해될 수 있을 것이다.

도면의 간단한 설명

- [0026] 본 발명에 관한 이해를 돕기 위해 상세한 설명의 일부로 포함되는, 첨부도면은 본 발명에 대한 실시예를 제공하고, 상세한 설명과 함께 본 발명의 기술적 사상을 설명한다.
- 도 1의 (a)는 사이드/노말 뷰(side/normal view)를 통해 얻은 사람들의 이미지이고, 도 1의 (b)는 오버헤드 뷰(overhead view)를 통해 얻은 사람들의 이미지이다.
- 도 2는 본 발명에 따른 사람 추적 방법의 흐름도이다.
- 도 3은 본 발명에 따른 얼룩 검출 과정을 도시한 것이다.
- 도 4는 본 발명의 성능을 시험하기 위한 테스트 시퀀스이다.
- 도 5는 검출된 모든 위치의 거리에 따른 정확도를 나타낸 그래프이다.
- 도 6은 도 4의 시퀀스 1에서, 본원 발명의 추적 방법과 MS 알고리즘을 비교한 것이다.
- 도 7은 도 4의 시퀀스 2에서, 본원 발명의 추적 방법과 MS 알고리즘을 비교한 것이다.
- 도 8은 도 4의 시퀀스 3에서, 본원 발명의 추적 방법과 MS 알고리즘을 비교한 것이다.
- 도 9는 도 4의 시퀀스 4에서, 본원 발명의 추적 방법과 MS 알고리즘을 비교한 것이다.

발명을 실시하기 위한 구체적인 내용

- [0027] 본 명세서 및 특허청구범위에 사용된 용어나 단어는 통상적이거나 사전적인 의미로 한정하여 해석되어서는 아니 되며, 발명자는 그 자신의 발명을 가장 최선의 방법으로 설명하기 위해서 용어의 개념을 적절하게 정의할 수 있다는 원칙에 입각하여, 본 발명의 기술적 사상에 부합되는 의미와 개념으로 해석되어야만 한다. 따라서 본 명세서에 기재된 실시예와 도면에 도시된 구성은 본 발명의 가장 바람직한 하나의 실시예에 불과할 뿐이고, 본 발명의 기술적 사상을 모두 대변하는 것은 아니므로, 본 출원시점에 있어서 이들을 대체할 수 있는 다양한 균등물과 변형예들이 있을 수 있음을 이해하여야 한다. 이하에서는, 첨부된 도면을 참조하여 본 발명의 실시예에 따른 오버헤드 카메라를 이용한 사람 추적 방법을 상세하게 설명하기로 한다.
- [0029] 도 2는 본 발명에 따른 사람 추적 방법의 흐름도이다.
- [0030] 본 발명은 이미지 내의 관련 이동 영역을 검출할 수 있는 추적 알고리즘을 기반으로 한다. 본 발명은 이러한 추적 알고리즘에 그리고 rHOG 알고리즘을 통합함으로써 간단하면서도 효과적인 사람 추적 방법을 제공한다. rHOG 알고리즘은 널리 알려진 알고리즘으로서 여기서의 자세한 설명은 생략한다.
- [0031] 도 2를 참조하여 본 발명에 따른 사람 추적 방법을 설명한다.
- [0032] 먼저 주어진 영상 내에 존재하는 사람을 검출하고, 상기 검출된 사람이 위치한 클러스터(cluster)를 반환한다. 이 경우, rHOG 알고리즘을 사용하여, 해당 영상을 스캔함으로써 영상 내에 존재하는 사람을 식별할 수 있다. 이때, rHOG 알고리즘은 계속하여 영상을 스캔하고, 사람이 검출되는 경우, 검출된 사람과 해당 위치의 클러스터를 반환(returning)(또는 검출)한다.

[0034] 다음으로 상기 클러스터 각각에 대하여 하기 (수학식 1)로 표현되는 클러스터 객체(**CO**)를 생성한다.

[0035] (수학식 1)

$$[0036] \text{CO} = \{N, t, \text{CC}(x, y), S, P\}$$

[0037] 여기서 N 은 캡처된 이미지, t 는 이미지의 캡처 시간, P 는 형성된 클러스터가 검출된 위치, $\text{CC}(x, y)$ 는

클러스터의 중심(centroid)으로 $\text{CC}(x, y) = \frac{1}{n} \sum_{i=1}^n P$ 으로 표현되며, S 는 클러스터 구성원의 합계이다.

$\text{CC}(x, y)$ 는 클러스터의 모든 멤버(member)의 평균 위치를 나타낸다. 이후, 상기 클러스터 객체를 통하여 하기 (수학식 2)와 같이, 클러스터 오브젝트 목록(**COL**)을 생성한다.

[0038] (수학식 2)

$$[0039] \text{COL} = \{\text{CO}_1, \text{CO}_2, \text{CO}_3, \dots, \text{CO}_n\}$$

[0040] 여기서 CO_n 은 n 번째 클러스터 객체이고, n 은 자연수이다.

[0042] 다음으로, 각 클러스터 객체(**CO**)마다 하기 (수학식 3)으로 표현되는 추적 객체(**TO**)를 생성한다.

[0043] (수학식 3)

$$[0044] \text{TO} = \{\text{Names}[], \text{Positions}[], \text{times}[], \text{CO}, \text{TP}\}$$

[0045] 여기서 **Names**는 이미지의 이름, **Positions**은 사람들이 검출된 위치, **times**는 프레임이 기록된 시간, **CO**는 클러스터 객체, **TP**는 추적기위 위치이다.

[0046] 추적 객체는 미리 정의된 각 클러스터 객체마다 도입된다. 이미지 또는 영상의 이름, 사람들이 검출된 위치, 프레임이 기록된 시간, 추적기의 위치 및 추적기 객체와의 연관성으로 구성된 클러스터 객체가 추적 객체의 속성이 될 수 있다.

[0047] 상기 추적 객체(**TO**)를 통해 하기 (수학식 4)와 같이 추적 오브젝트 목록(**TOL**)을 생성한다.

[0048] (수학식 4)

$$[0049] \text{TOL} = \{\text{TO}_1, \text{TO}_2, \text{TO}_3, \dots, \text{TO}_n\}$$

[0050] 여기서 TO_n 은 n 번째 추적 객체이고, n 은 자연수이다. 이후 추적기 객체가 지속적으로 업데이트될 수 있다. 상기 수학식 4는 목록 형식으로 추가된 추적(Tracker) 객체를 보여준다.

[0052] 다음으로, 상기 추적 오브젝트 목록(**TOL**)에서 다음 프레임의 동일한 영역 내의 추적 객체를 선택하여, 상기 다음 프레임에서 선택된 추적 객체의 위치를 추정한다. 즉, 다음 프레임의 동일한 영역 내에서, 추적 오브젝트 목록의 각 객체를 선택하여 추적 객체의 위치를 계산한다. 이는 반복적으로 수행될 수 있다.

[0053] 추적 객체의 위치를 계산하기 위한 속도(v)와 속력($|v|$)(즉, 속도의 크기)는 하기 (수학식 5)를 통해 계산될 수 있다.

[0054] (수학식 5)

$$\begin{cases} v = \frac{\Delta S}{\Delta t}, & \text{if cluster size} = 2 \\ v = \frac{1}{n} \sum_{k=1}^n \frac{\Delta S(k-1)}{\Delta t(k-1)}, & \text{if cluster size} > 2 \end{cases}$$

[0055]

[0056] 이후 다음과 같은 과정 중 하나를 따른다.

[0057] a) 추적 객체가 1 개의 감지된 위치와 동일한 경우 v 는 0이 된다. 그러나 이는 탐지되는 첫 번째 프레임의 경우에만 해당된다.

[0058] b) 추적 객체가 2 개의 감지된 위치와 동일한 경우 $v = (\Delta S / \Delta t)$ 이다. 여기서 ΔS 와 Δt 는 각각 거리와 시간의 차이를 나타낸다.

[0059] c) 추적 객체가 2개 이상의 감지된 위치를 가진다면, 평균 속도는 최대 10 개의 이전 탐지 위치를 사용하여 결정된다.

[0061] 다음으로, 추적 객체의 위치를 하기 (수학식 6)과 같은 방식으로 추정하여 근사화한다.

[0062] (수학식 6)

$$Px(x, y) = Pc(x, y) + v * (\Delta t)$$

[0064] 여기서 $Px(x, y)$ 는 추적 객체의 예상 위치, $Pc(x, y)$ 는 추적 객체의 현재 위치이다. 여기서 Δt 는 다음 프레임에서의 얼룩의 움직임을 추정하는데 사용된다.

[0065] 다음으로, 다음 프레임의 탐색 영역(S_g)을 하기 (수학식 7)과 같이 계산한다.

[0066] (수학식 7)

$$S_g = PW + (\Delta S)$$

[0067]

[0068] 여기서 PW 는 32 by 32 pixels, ΔS 는 추적 객체의 위치변화이다. 본 발명에 따른 사람 추적 방법은 사람의 평균 너비를 나타내는 약 64 픽셀의 영역을 분석한다. 검출된 각 얼룩이 적어도 사람의 50 % 크기 이상인 것이 일반적이다. 따라서 이러한 정보를 사용하여 (수학식 7)과 같이 적절한 정사각형(square) 크기의 검색 영역 S_g 를 계산한다. 첫째로, 크기가 PW 가 32×32 픽셀인 얼룩을 찾는다. 나머지 프레임들 내의 모든 후속 탐색들에 대해, S_g 는 거리의 이전 궤적의 정보를 통해 갱신될 수 있다.

[0070] 다음으로, 상기 다음 프레임의 탐색 영역 내에 존재하는 사람을 검출한다.

[0071] 탐색 영역 내에 사람이 검출되지 않는 경우, 해당 추적 객체를 제거하고, 사람이 검출되는 경우, 이전의 위치와 현재 위치의 거리를 비교하여, 상기 거리가 40 픽셀 이하라면 추적 객체 및 클러스터의 중심을 업데이트하고, 상기 거리가 40 픽셀보다 크다면 새로운 추적 객체를 생성한다.

[0072] 추적된 사람 객체는 프레임에서 사람의 새로운 탐지된 위치를 사용하여 업데이트 된 다음 앞의 단계를 반복한다. 감지된 위치가 기존 추적 객체에서 40 픽셀 이상 떨어져 있는 다른 모든 거리의 경우 추적 모듈의 새

로운 추적 객체가 만들어진다. 이후 새로운 추적 객체에 대해 앞의 단계를 반복한다. rHOG 알고리즘이 최대 5개 프레임까지 기존 추적 객체에 대한 적합성을 탐지할 수 없는 경우 해당 객체는 제거된다. 나머지 추적 오브젝터 목록은 적합성 여부를 검사하여, 비어 있지 않은 목록이 발견되거나 추적 과정을 중단할 때 앞의 단계를 거친다.

[0074] 한편, 본 발명은 성능을 높이기 위해 얼룩 탐지 모듈이 추가적으로 보완될 수 있다. 도 3은 본 발명에 따른 얼룩 검출 과정을 도시한 것이다.

[0075] 이는 이동 가능성이 더 높은 영상 영역만 찾을 수 있다는 점에서 상당한 이점이 있다. 얼룩 탐지 모듈을 통하여 사람을 검출하는 과정은 하기와 같은 단계로 수행된다.

[0076] 먼저 하기 (수학식 8)과 같이 연속된 프레임을 통해 모션맵 이미지를 얻는다. 도 3의 (a)는 원본 이미지, (b)는 모션 맵 이미지이다.

[0077] (수학식 8)

[0078]
$$M_t = \text{abs}(I_{t+1} - I_t)$$

[0079] 여기서 M_t 는 모션맵 이미지, I_t , I_{t+1} 는 연속된 프레임이다.

[0080] 이후 상기 모션맵 이미지를 8×8 픽셀의 하위 영역들로 나누고, 상기 하위 영역들의 픽셀 강도 값을 합산한다. 이같은 과정을 거친 이미지가 도 3의 (c)에 도시되어 있다. 이후 합산된 픽셀 강도 값과 임계값을 비교하여 최종적으로 얼룩여부를 판단한다. 이는 (수학식 9)와 같이 표현될 수 있다.

[0081] (수학식 9)

[0082]
$$\begin{cases} \text{Blob,} & \text{if } B(x, y) > \tau \\ \text{No Blob,} & \text{if } B(x, y) \leq \tau \end{cases}$$

[0083] 여기서 $B(x, y)$ 는 상기 하위 영역들의 합산된 픽셀 강도 값, τ 는 임계값이다. 도 3의 (d)는 임계 값인 τ 가 32×32 픽셀로 설정된 경우를 도시한 것이고, 도 3의 (e)는 임계 값이 적용되고 작은 얼룩이 제거 된 후 분할 된 이미지를 도시한 것이다.

[0085] 도 4는 본 발명의 성능을 시험하기 위한 테스트 시퀀스이다. 도 4의 (a)는 시퀀스 1, (b)는 시퀀스 2, (c)는 시퀀스 3, (d)는 시퀀스 4이다.

[0086] 작업 영역을 제한하는 바닥에 특별한 특성이 있지만 사람의 움직임에는 영향일 미치지 않으며, 사람의 옷 색깔은 제한이 없다. 이미지 (640 X 480 픽셀)는 5m 높이의 PTZ 카메라에서 초당 20 프레임의 비율로 캡처 되었다. PTZ 212 카메라의 사양은 Fujinon 렌즈, F1.8, 고정 조리개, 수직 시야각 : 35° 내지 105° 및 수평 시야각 : 44° 내지 140° 이다.

[0087] 알고리즘의 성능을 평가하기 위해 도 4에 도시된 4 가지 테스트 시퀀스를 사용하여 기존의 알고리즘과 비교하였다. 비교를 위해 각 테스트 시퀀스는 수동으로, 즉 실측 자료(ground truth)를 사용하여) 자동으로 검사되고, 테스트 비디오에서 사람들의 궤적 / 경로를 기반으로 비교되었다. 도 4의 (d)에 도시된 마지막 테스트 시퀀스는 다른 시퀀스들에 비해 비교적 복잡한데, 이는 자동차 조립체에서 용접을 수행하는 동안 사람의 지그재그로의 움직임이 많기 때문이다.

[0089] 도 5는 검출된 모든 위치의 거리에 따른 정확도를 나타낸 그래프이다.

[0090] x축은 검출 된 모든 위치의 거리를 나타내고, y축은 검출 속도 또는 정확도를 나타낸다. 도 5를 참조하면, 본 발명에 따른 사람 추적 방법은 기존의 알고리즘과 비교하여 훨씬 높은 적중률이 나타나는 것을 확인할 수 있다. 일 예로 거리가 20 픽셀인 경우 본 발명은 99 %의 정확도가 나타났지만, MS 알고리즘의 경우 64 %의 정확도가

나타났다. 이하에서는 각 시퀀스에 대하여 본원 발명의 추적 방법과 MS 알고리즘을 구체적으로 비교해 본다.

- [0092] 도 6은 도 4의 시퀀스 1에서, 본원 발명의 추적 방법과 MS 알고리즘을 비교한 것이다. 도 6의 (a), (b)는 본 발명에 따른 추적 방법을 통하여 수행된 것이고, 도 6의 (c), (d)는 MS 알고리즘을 통하여 수행된 결과이다.
- [0093] 비디오 테스트 세트의 시작시간 점프는 0.5, 0.55, 0.8, 0.65초이다. 누락된 프레임에서 MS 알고리즘은 사용자의 위치를 감지할 수 있다. 도 6의 (b)는 이러한 시간 차이를 보여준다. 시험 시퀀스 동안에, 사람이 일을 시작하고 여러 번 손을 움직이기 시작하면, MS 알고리즘에서는 얼룩의 중심이 사람에서 팔꿈치로 이동된다는 것이 관찰되었다.
- [0094] 이는 이른바 "탐지되지 않은 위치"(undetected positions)로 이어진다. MS 알고리즘의 약점은 소수의 프레임에서 사용자가 전혀 감지하지 못한다는 것이다. 실제 이와 같은 현상이 테스트 시퀀스 중간에 관찰되었다. 즉, 실제 관측 자료에 대한 위치의 검출에 약점이 나타나고 있음을 알 수 있다.
- [0095] 반면, 본 발명에 따른 추적 방법을 통해 수행된 결과는 도 6의 (c) 및 (d)에 나타나있다. 도 6의 (c), (d)에 나타난 수치를 참조하면, 본 발명은 모든 연속된 프레임에서 사람을 탐지할 수 있는 것을 확인할 수 있다. 이는 본 발명에 따른 추적 방법이 MS 알고리즘 보다 향상된 성능이 나타났음을 의미한다. 구체적으로 MS 알고리즘을 통해 나타난 정확도(55 %) 보다 현저히 향상된 정확도(97 %)가 나타났다. 편차 또한 MS 알고리즘에서 나타난 13 픽셀에 비해 7 픽셀의 더 낮은 편차가 나타난 것으로 확인되었다.
- [0097] 도 7은 도 4의 시퀀스 2에서, 본원 발명의 추적 방법과 MS 알고리즘을 비교한 것이다. 도 7의 (a), (b)는 본 발명에 따른 추적 방법을 통하여 수행된 것이고, 도 7의 (c), (d)는 MS 알고리즘을 통하여 수행된 결과이다.
- [0098] 테스트 시퀀스 2에서는 MS 알고리즘이 비디오 시퀀스의 많은 위치를 놓친 것을 확인할 수 있다. 또한 감지된 얼룩은 매우 멀리 있고 실측 자료와 다른 것으로 발견되었으며, 크기가 매우 작았다. 이는 그림 7(a)와 (b)에 나타나 있다. MS 알고리즘으로 식별된 얼룩 크기는 8 X 20 픽셀이었고, 인체의 작은 부위만 차지했기 때문에 얼룩은 사람의 실제 위치에 부합하지 않은 것을 알 수 있다.
- [0099] 반면, 본 발명에 따른 추적 방법은 32 X 32 픽셀 이상의 사람을 나타내는 가정에 기초하였기 때문에, 가령 사람의 몸통에 있는 작은 얼룩들에 의한 영향을 무시할 수 있었다. 따라서 32X32 픽셀 임계값 아래의 모든 얼룩이 삭제된다. 도 7의 (c) 및 (d)를 참조하면, 본 발명에 따른 추적 방법은 감지된 위치뿐만 아니라 정확도에 있어서도 MS 알고리즘의 성능을 능가한다는 것을 확인할 수 있다. 검출 속도는 MS 알고리즘의 경우, 평균 거리 13 픽셀 내에서 35.8 %로 나타났으나, 본 발명에 따른 추적 방법은 평균 거리 6 픽셀에서 100 %로 나타났다.
- [0101] 도 8은 도 4의 시퀀스 3에서, 본원 발명의 추적 방법과 MS 알고리즘을 비교한 것이다. 도 8의 (a), (b)는 본 발명에 따른 추적 방법을 통하여 수행된 것이고, 도 8의 (c), (d)는 MS 알고리즘을 통하여 수행된 결과이다.
- [0102] 도 8의 (a) 및 (b)는 MS 알고리즘에 대한 테스트 시퀀스 3을 나타낸다. 이를 참조하면, 감지된 얼룩이 사람의 몸에 위치하지 않고 대부분 사람의 어깨나 팔 위에 위치하며 크기가 작은 것을 확인할 수 있다. 이러한 이미지에서 대부분의 얼룩은 크기와 위치 때문에 감지되지 않는다. 이는 도 8의 (a)에 도시되어 있다. 도 8의 (b)에는 191 프레임 후에 20회 스캔의 시간 간격을 통해 관찰한 결과가 나타나있다.
- [0103] 이러한 MS 알고리즘과 비교하여 본 발명에 따른 추적 방법은 훨씬 개선된 검출 결과를 보여준다. 본 발명에 따른 추적 방법은 처음부터 즉시 탐지를 시작하며, 탐지 대상자의 위치에 대한 실측 자료 측면에서 향상된 탐지 기능을 제공하였다. 본 발명에 따른 추적 방법은 시간 간격 동안 추적을 계속하여 앞서 언급한 20초와 같은 큰 시간 갭을 처리하였다. 정지 상태에서 사람을 감지하지 못한 MS 알고리즘과 비교했을 때, 본 발명은 사람이 정적인 상태를 유지했음에도 불구하고 계속해서 사람을 탐지할 수 있었다. 이러한 탐지 결과는 도 8의 (c) 및 (d)에 나타나 있다. 이를 참조하면, 그래프 어느 지점이든 실측에 가깝게 검출되는 것을 확인할 수 있다.
- [0104] 하기 표 1은 앞서 설명한 시퀀스 1, 2, 3에서 얻은 결과를 비교한 것이다. 여기서 HR은 정확도를 나타내며, MSE는 제공된 오류를 나타낸다.

(표 1)

Algo	1		2		3		Avg	
	MSE	HR	MSE	HR	MSE	HR	MSE	HR
Proposed	7.0	97	6.0	100	5.0	99.8	6.0	99.08
MS	13.0	55	13.0	35.8	9.0	38.7	11.0	48.38
CC	12.0	42.1	10.0	25.3	13.0	67	12.0	48.0
MSFG	12.0	50	12.0	30	10.0	43.6	12.0	45.83
MSPF	13.0	32.1	9.0	42.1	7.0	32	10.0	40.43

그 결과 평균 탐지율이 99.08 %로 나타난 본 발명이 다른 모든 알고리즘보다 높은 것으로 나타났다. 본 발명을 제외한 다른 알고리즘 중에서 가장 높은 정확도는 MS 추적 방법의 48.38 % 방법이었다.

도 9는 도 4의 시퀀스 4에서, 본원 발명의 추적 방법과 MS 알고리즘을 비교한 것이다.

시험 시퀀스 번호 4는 시나리오의 갑작스럽고 역동적인 변화 및 프레임 내 데이터의 차이로 인해 복잡한 특성을 가지고 있다. 시퀀스는 관측된 데이터 공백이 있는 1021 개의 프레임으로 구성된다. 이 프레임에서 사람은 약 8 초라는 한번의 긴 점프 구간을 제외하고 한 방향으로 40 픽셀 반경으로 유지되었다. 시간이 건너뛰는 동안 사람의 위치 변화는 원래 위치와 동일한 수평 방향으로 관찰되며, 이는 약 100 픽셀이다.

비디오 총 시간 길이는 76초이며 거의 30초 분량의 데이터가 누락되었다. 누락된 30초는 총 730개의 프레임으로 구성된다. 용접으로 인한 와이어, 기계 및 스파크가 있기 때문에 환경의 복잡성이 나타난다. 사람은 이 환경 안에 가려져 있다.

본 발명에 따른 추적 방법을 다른 세 개의 알고리즘과 비교한다. 사람의 감지된 위치의 궤적은 도 9에서 확인할 수 있다. 도 9의 (a)를 참조하면, 일부가 한계점 이상의 위치에 있는 것으로 관찰되며, 실제 관측과는 거리가 있는 것으로 나타난다. 이는 용접 기계로 인한 불꽃이 거짓으로 감지된 얼룩을 만들기 때문이다.

도 9의 (d)는 시퀀스에서 관찰된 시간 간격을 나타내기 위해 x축의 시간 대비 Y축의 실제 관측 자료를 나타낸 것이다. 도 9의 (c) 및 (b)는 각각 본 발명에 따른 추적 방법과 시간 및 실제 관측 자료에 대한 MS 추적 알고리즘을 사용해 검출된 얼룩 위치의 궤적을 나타낸다.

이러한 그래프의 결과는 비디오 시퀀스 프레임에서 시간 간격을 가진 사람의 위치를 감지하는 측면에서의 성능을 보여주는 것으로, 이를 표 2에 나타내었다.

(표 2)

Algorithm	MSE	HR (%)
proposed	9.0	97.8
Mean Shift	11.0	21.8
Connected Component	14.0	22.2
MSPF	11.0	23.8
MSFG	11.0	31.2

테스트 시퀀스 4는 용접 등과 같은 환경에서 발생하는 교차 및 기타 방해 요인으로 인해 가장 복잡한 시퀀스이지만, 본 발명에 따른 추적 방법이 다른 검출 알고리즘 보다 높은 정확도를 나타냈다. 본 발명을 제외한 다른 알고리즘에서 나타난 가장 높은 정확도는 31.2 %에 불과하여, 97.8 %의 정확도가 나타난 본원 발명과는 현저한 차이가 존재하는 것을 알 수 있었다.

본 발명은 오버헤드 카메라를 통하여 사람들을 추적하는 효과적인 방법을 개시한다. 본 발명은 시간 간격이 크더라도 정지된 사람을 효과적으로 탐지하고 추적할 수 있다.

이상에서 본 발명의 대표적인 실시예들을 상세하게 설명하였으나, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자는 상술한 실시예에 대하여 본 발명의 범주에서 벗어나지 않는 한도 내에서 다양한 변형이 가능함을

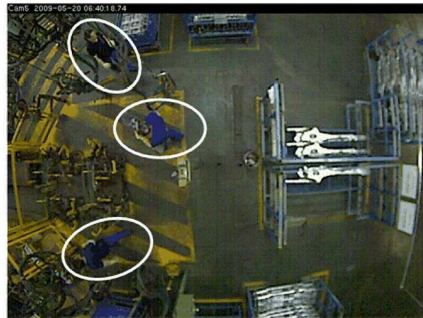
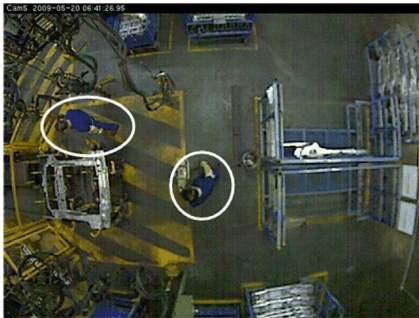
이해할 것이다. 그러므로 본 발명의 권리범위는 설명된 실시예에 국한되어 정해져서는 안되며, 후술하는 특허청구범위뿐만 아니라 이 특허청구범위와 균등한 것들에 의해 정해져야 한다.

도면

도면1

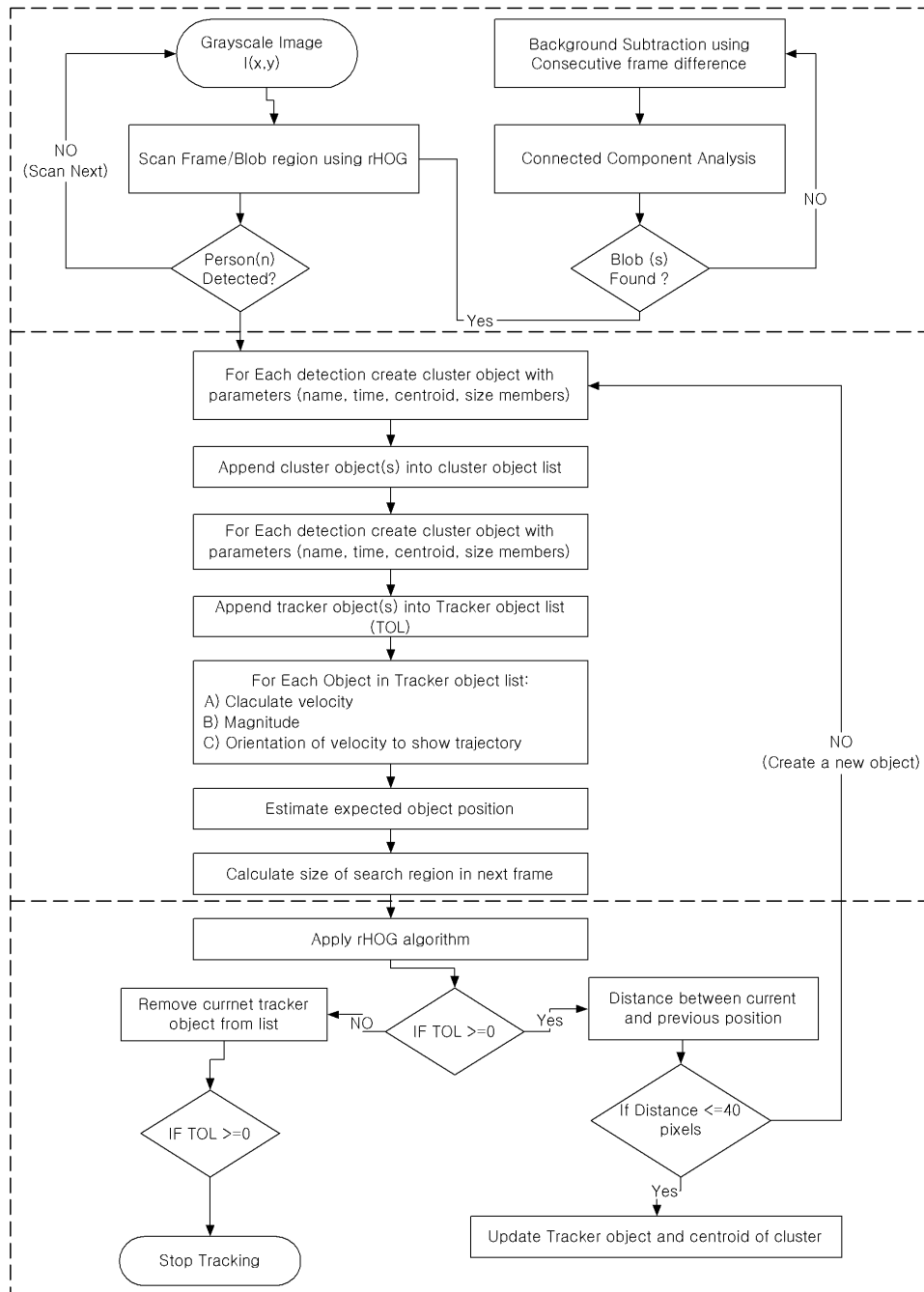


(a)

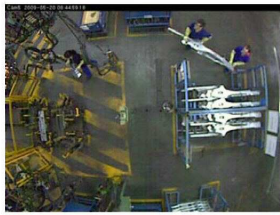


(b)

도면2



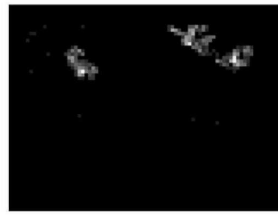
도면3



(a)



(b)



(c)

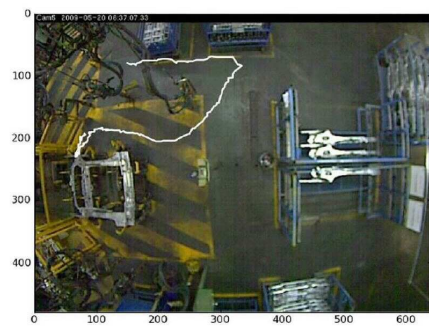


(d)

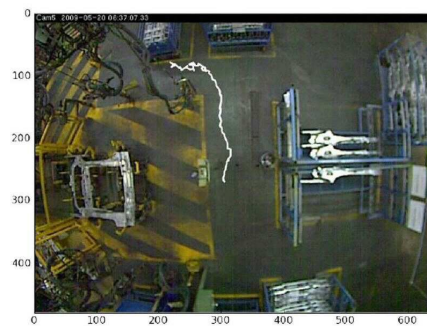


(e)

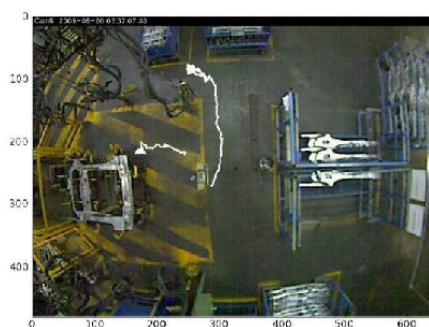
도면4



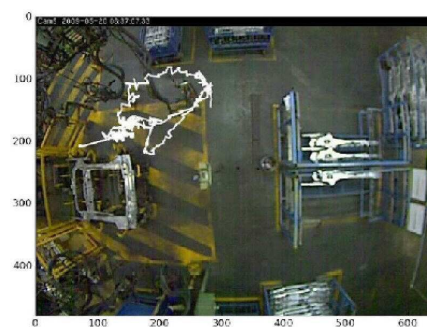
(a)



(b)

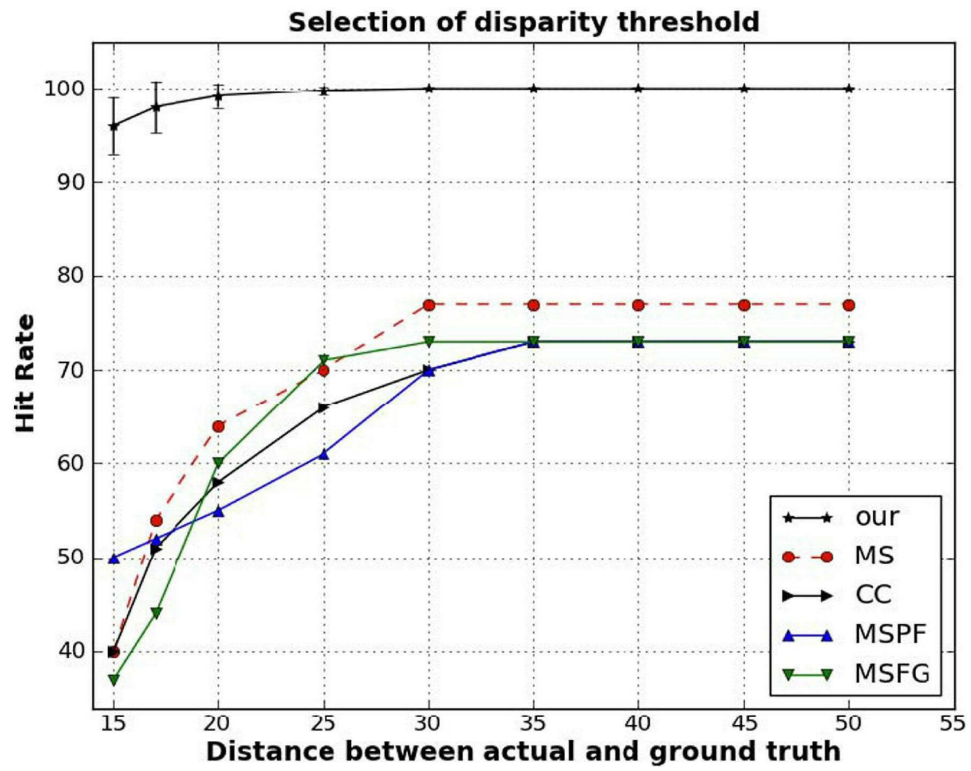


(c)

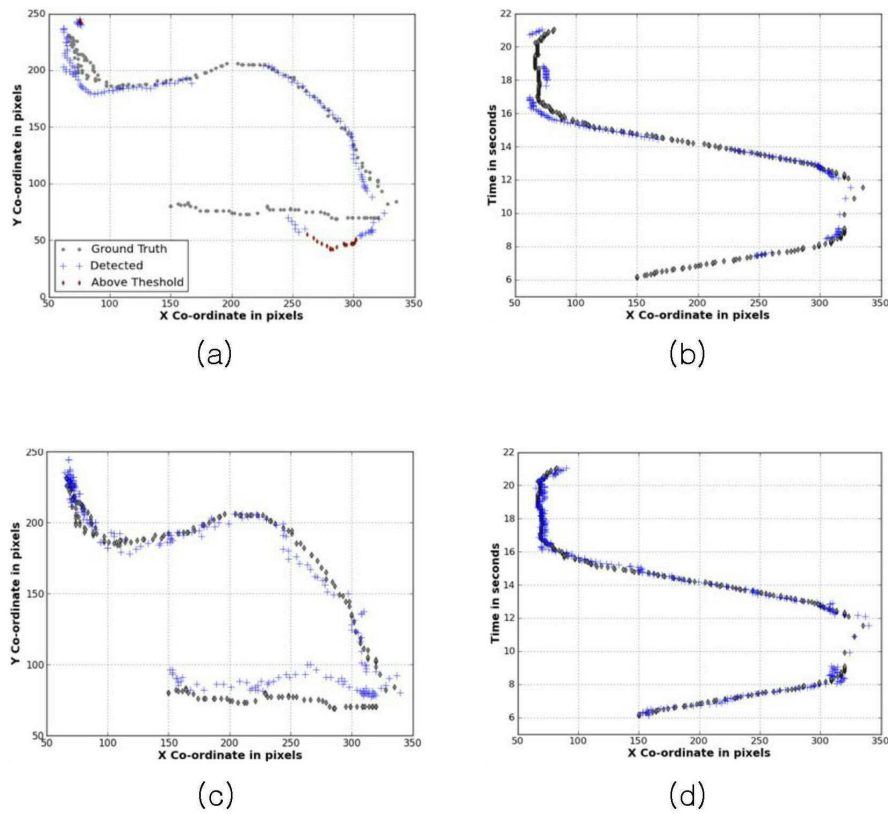


(d)

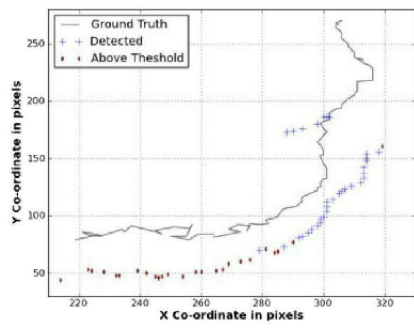
도면5



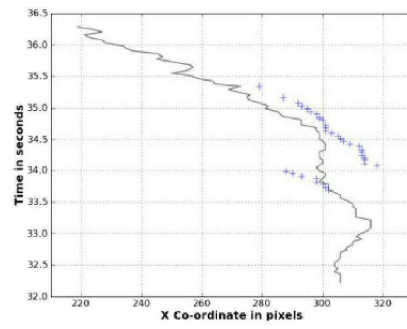
도면6



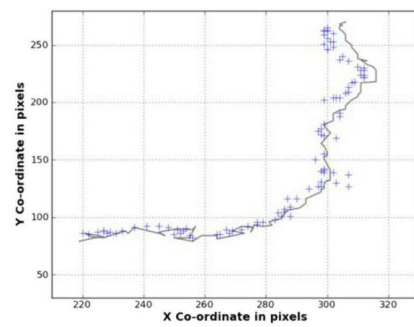
도면7



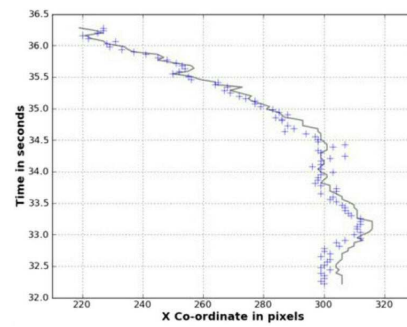
(a)



(b)

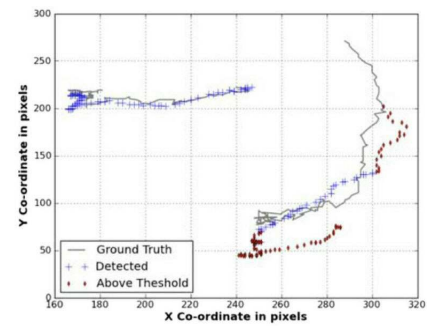


(c)

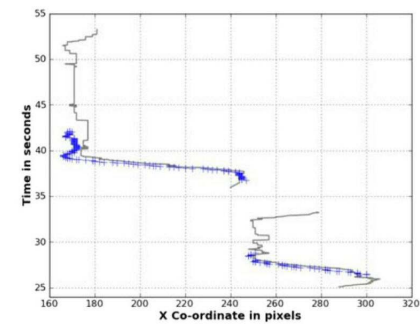


(d)

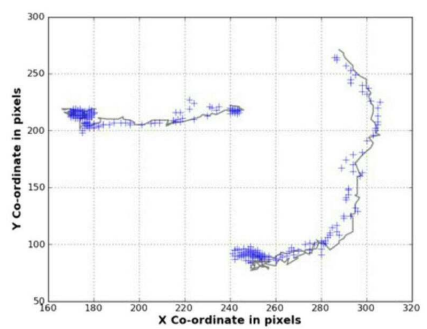
도면8



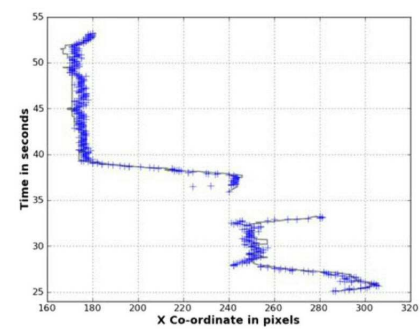
(a)



(b)

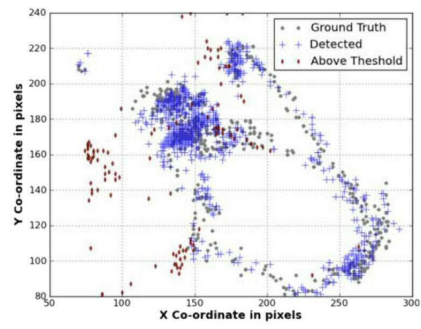


(c)

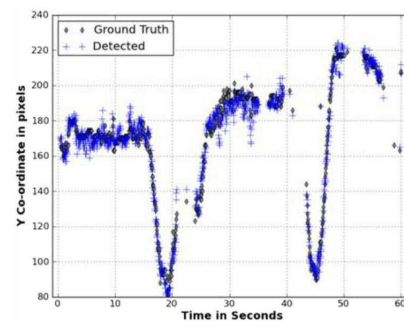


(d)

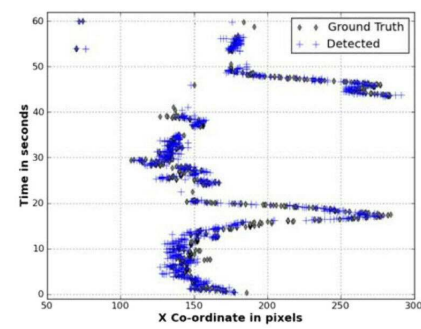
도면9



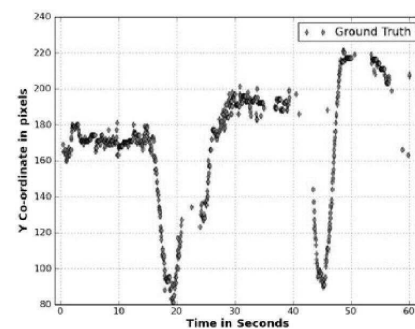
(a)



(b)



(c)



(d)