**Object Tracking: A Survey**

Alper Yilmaz

Ohio State University

Omar Javed

ObjectVideo, Inc.

and

Mubarak Shah

University of Central Florida

The goal of this article is to review the state-of-the-art tracking methods, classify them into different categories, and identify new trends. Object tracking, in general, is a challenging problem. Difficulties in tracking objects can arise due to abrupt object motion, changing appearance patterns of both the object and the scene, nonrigid object structures, object-to-object and object-to-scene occlusions, and camera motion.

이 기사의 목표는 최첨단 추적 방법을 검토하고 이를 다른 범주로 분류하고 새로운 경향을 파악하는 것입니다. 객체 추적은 일반적으로 어려운 문제입니다. 객체 추적의 어려움은 급격한 객체 이동, 객체 및 장면의 외양 패턴 변경, 비 열역학적 객체 구조, 객체 간 객체 및 객체 간 오클루젼 및 카메라 모션으로 인해 발생할 수 있습니다.

Tracking is usually performed in the context of higher-level applications that require the location and/or shape of the object in every frame. Typically, assumptions are made to constrain the tracking problem in the context of a particular application. In this survey, we categorize the tracking methods on the basis of the object and motion representations used, provide detailed descriptions of representative methods in each category, and examine their pros and cons. Moreover, we discuss the important issues related to tracking including the use of appropriate image features, selection of motion models, and detection of objects.

추적(트랙킹)은 대개 모든 프레임에서 객체의 위치 및 / 또는 모양을 필요로 하는 고급 응용 프로그램의 컨텍스트에서 수행됩니다. 일반적으로 특정 응용 프로그램의 컨텍스트에서 추적 문제를 제한하기 위한 가정이 있습니다. 이 조사에서는 사용 된 객체 및 동작 표현을 기반으로 추적 방법을 분류하고 각 범주의 대표적인 방법에 대한 자세한 설명을 제공하며 장단점을 검사합니다. 또한, 적절한 이미지 특징의 사용, 모션 모델의 선택 및 객체 탐지를 포함하여 추적과 관련된 중요한 문제를 논의합니다.

Categories and Subject Descriptors: I.4.8 [Image Processing and Computer Vision]: Scene Analysis— Tracking

범주 및 주제 설명자 : I.4.8 [이미지 처리 및 컴퓨터 비전] : 장면 분석 - 추적

General Terms: Algorithms

일반 용어 : 알고리즘

Additional Key Words and Phrases: Appearance models, contour evolution, feature selection, **object detection**, object representation, point tracking, shape tracking

추가 키워드 및 구문 : 모양 모델, 등고선 전개, 피쳐 선택, 객체 감지, 객체 표현, 포인트 추적, 형상 추적

ACM Reference Format:

Yilmaz, A., Javed, O., and Shah, M. 2006. Object tracking: A survey. ACM Comput. Surv. 38, 4, Article 13 (Dec. 2006), 45 pages. DOI = 10.1145/1177352.1177355 <http://doi.acm.org/10.1145/1177352.1177355>

----------------------------------------------------------------------------------------------------------------------

This material is based on work funded in part by the US Government. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the US Government.

이 자료는 부분적으로 미국 정부가 자금을 지원한 업무를 기반으로합니다. 본 자료에 언급된 모든 의견, 결과 및 결론이나 권장 사항은 저자의 것이며 의견이 반드시 미국 정부의 견해를 반영하는 것은 아닙니다.

Author’s address: A. Yilmaz, Department of CEEGS, Ohio State University; email: yilmaz.15@osu.edu; O. Javed, ObjectVideo, Inc., Reston, VA 20191; email: ojaved@objectvideo.com; M. Shah, School of EECS, University of Central Florida; email: [shah@cs.ucf.edu](mailto:shah@cs.ucf.edu).

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or direct commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee.

개인적 또는 교실 이용을 위해이 저작물의 일부 또는 전부를 디지털 또는 하드카피 할 수 있는 권한은, 이익을 위해 또는 직접적인 상업적 이익을 위해 사본을 만들거나 배포하지 않고, 첫 페이지 또는 초기 화면 전체 인용문과 함께 디스플레이의 ACM보다 다른 사람들이 소유한 본 저작물의 구성 요소에 대한 저작권은 반드시 준수해야합니다. 신용으로 추상화하는 것이 허용됩니다. 달리 복사하거나, 다시 게시하거나, 서버에 게시하거나, 목록에 재배포하거나, 다른 저작물에서 이 저작물의 구성 요소를 사용하려면 사전 허가 및 / 또는 수수료가 필요합니다.

Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or [permissions@acm.org](mailto:permissions@acm.org).

c 2006 ACM 0360-0300/2006/12-ART13 $5.00 DOI: 10.1145/1177352.1177355 http://doi.acm.org/10.1145/ 1177352.1177355.

사용권한을 요청하려면, 위에 나온곳에 요청하자

**1. INTRODUCTION**

Object tracking is an important task within the field of computer vision. The proliferation of high-powered computers, the availability of high quality and inexpensive video cameras, and the increasing need for automated video analysis has generated a great deal of interest in object tracking algorithms. There are three key steps in video analysis: detection of interesting moving objects, tracking of such objects from frame to frame, and analysis of object tracks to recognize their behavior. Therefore, the use of object tracking is pertinent in the tasks of:

객체 추적은 컴퓨터 비전 분야에서 중요한 작업입니다. 고성능 컴퓨터의 확산, 고품질 및 저렴한 비디오 카메라의 가용성 및 자동화 된 비디오 분석에 대한 필요성 증가로 인해 객체 추적 알고리즘에 많은 관심이 생겼습니다. 비디오 분석에는 **흥미로운 움직이는 물체를 탐지**하고, **프레임간에 프레임을 추적**하며, **물체 추적을 분석하여 동작을 인식**하는 **세 가지 주요 단계**가 있습니다. 따라서 객체 추적의 사용은 다음과 같은 작업에서 적절합니다.

—motion-based recognition, that is, human identification based on gait, automatic object detection, etc;

- 모션 기반 인식, 즉 보행, 자동 물체 탐지 등을 기반으로 한 사람 식별;

—automated surveillance, that is, monitoring a scene to detect suspicious activities or unlikely events;

의심스러운 활동이나 가능성이 없는 사건을 탐지하기 위해 장면을 모니터링하는 자동 감시;

—video indexing, that is, automatic annotation and retrieval of the videos in multimedia databases;

비디오 인덱싱, 즉 멀티미디어 데이터베이스에서 비디오의 자동 주석 및 검색;

—human-computer interaction, that is, gesture recognition, eye gaze tracking for data input to computers, etc.;

컴퓨터 상호 작용, 즉 제스처 인식, 컴퓨터에 입력되는 데이터에 대한 시선 추적 등.

—traffic monitoring, that is, real-time gathering of traffic statistics to direct traffic flow.

트래픽 모니터링을위한 트래픽 통계를 실시간으로 수집합니다.

—vehicle navigation, that is, video-based path planning and obstacle avoidance capabilities.

즉, 비디오 기반 경로 계획 및 장애물 회피 기능을 제공합니다.

In its simplest form, tracking can be defined as the problem of estimating the trajectory of an object in the image plane as it moves around a scene. In other words, a tracker assigns consistent labels to the tracked objects in different frames of a video. Additionally, depending on the tracking domain, a tracker can also provide object-centric information, such as orientation, area, or shape of an object. Tracking objects can be complex due to:

가장 단순한 형태에서, 추적은 장면 주위를 움직일 때 이미지 평면에서 물체의 궤적을 추정하는 문제로 정의할 수 있습니다. 즉, 추적기는 추적된 객체에 비디오의 여러 프레임에 일관된 레이블을 할당합니다. 또한 추적 영역에 따라 추적기는 객체의 방향, 면적 또는 모양과 같은 객체 중심 정보를 제공할 수도 있습니다. 다음과 같은 이유로 추적 객체가 복잡할 수 있습니다.

—loss of information caused by **projection** of the 3D world on a 2D image,

2D 이미지에 대한 3D 세계의 **투영**으로 인한 정보의 손실,

—noise in images,

이미지의 노이즈,

—complex object motion,

complex 오브젝트 모션,

—nonrigid or articulated nature of objects,

객체의 Non-Rigid 또는 Articulated 속성,

(\*\*모션 세그먼테이션을 위한 일반 프레임워크: <http://www.cs.unc.edu/~marc/pubs/YanECCV06.pdf> )

—partial and full object occlusions,

부분적 및 전체 객체 오클루젼,

—complex object shapes,

complex 오브젝트 셰이프

—scene illumination changes, and

scene 일루미네이션 체인지, 그리고

—real-time processing requirements.

실시간 처리 요구 사항

One can simplify tracking by imposing constraints on the motion and/or appearance of objects. For example, almost all tracking algorithms assume that the object motion is smooth with no abrupt changes. One can further constrain the object motion to be of constant velocity or constant acceleration based on a priori information. Prior knowledge about the number and the size of objects, or the object appearance and shape, can also be used to simplify the problem.

객체의 동작 및 / 또는 모양에 제한을 가함으로써 추적을 단순화 할 수 있습니다. 예를 들어, 거의 모든 추적 알고리즘은 오브젝트 모션이 갑작스러운 변화없이 부드럽다고 가정합니다. 또한, 선험적 정보에 기초하여 일정한 속도 또는 일정한 가속도로 대상물의 움직임을 제한할 수있다. 객체의 수와 크기, 객체 모양과 모양에 대한 사전 지식을 사용하여 문제를 단순화 할 수 있습니다.

Numerous approaches for object tracking have been proposed. These primarily differ from each other based on the way they approach the following questions: Which object representation is suitable for tracking? Which image features should be used? How should the motion, appearance, and shape of the object be modeled? The answers to these questions depend on the context/environment in which the tracking is performed and the end use for which the tracking information is being sought. A large number of tracking methods have been proposed which attempt to answer these questions for a variety of scenarios. The goal of this survey is to group tracking methods into broad categories and provide comprehensive descriptions of representative methods in each category. We aspire to give readers, who require a tracker for a certain application, the ability to select the most suitable tracking algorithm for their particular needs. Moreover, we aim to identify new trends and ideas in the tracking community and hope to provide insight for the development of new tracking methods.

객체 추적을 위한 수많은 접근 방식이 제안되었습니다. 이들은 주로 다음 질문에 접근하는 방식에 따라 서로 다릅니다. 추적에 적합한 객체 표현은 무엇입니까? 어떤 이미지 기능을 사용해야합니까? 객체의 동작, 모양 및 모양이 어떻게 모델링되어야 합니까? 이러한 질문에 대한 대답은 추적이 수행되는 상황 / 환경 및 추적 정보가 요구되는 최종 용도에 따라 다릅니다. 다양한 시나리오에 대해 이러한 질문에 답하는 많은 추적 방법이 제안되었습니다. 이 조사의 목표는 추적 방법을 광범위한 범주로 그룹화하고 각 범주의 대표적인 방법에 대한 포괄적인 설명을 제공하는 것입니다. 우리는 특정 응용 프로그램에 대한 추적기가 필요한 독자에게 특정 요구 사항에 가장 적합한 추적 알고리즘을 선택하는 기능을 제공하고자 합니다. 또한 추적 커뮤니티에서 새로운 트렌드와 아이디어를 확인하고 새로운 추적 방법 개발에 대한 통찰력을 제공하고자합니다.

Our survey is focused on methodologies for tracking objects in general and not on trackers tailored for specific objects, for example, person trackers that use human kinematics as the basis of their implementation. There has been substantial work on tracking humans using articulated object models that has been discussed and categorized in the surveys by Aggarwal and Cai [1999], Gavrilla [1999], and Moeslund and Granum [2001]. We will, however, include some works on the articulated object trackers that are also applicable to domains other than articulated objects.

우리의 설문 조사는 일반적으로 객체를 추적하는 방법론에 중점을두고 있으며 특정 객체에 맞는 추적기가 아니라,

예를 들어 인간 운동학을 구현의 기초로 사용하는 사람 추적기입니다.

Aggarwal and Cai [1999], Gavrilla [1999], Moeslund and Granum [2001]의 조사에서 논의되고 범주화된 articulated 객체 모델을 사용하여 인간 추적에 대한 실질적인 연구가 이루어졌다. 그러나 articulated 객체 추적기에는 articulated 객체 이외의 영역에도 적용 할 수 있는 일부 작업이 포함됩니다.

We follow a bottom-up approach in describing the issues that need to be addressed when one sets out to build an object tracker. The first issue is defining a suitable representation of the object. In Section 2, we will describe some common object shape representations, for example, points, primitive geometric shapes and object contours, and appearance representations. The next issue is the selection of image features used as an input for the tracker. In Section 3, we discuss various image features, such as color, motion, edges, etc., which are commonly used in object tracking. Almost all tracking algorithms require detection of the objects either in the first frame or in every frame. Section 4 summarizes the general strategies for detecting the objects in a scene. The suitability of a particular tracking algorithm depends on object appearances, object shapes, number of objects, object and camera motions, and illumination conditions. In Section 5, we categorize and describe the existing tracking methods and explain their strengths and weaknesses in a summary section at the end of each category. In Section 6, important issues relevant to object tracking are discussed. Section 7 presents future directions in tracking research. Finally, concluding remarks are sketched in Section 8.

우리는 객체 추적기를 구축하기 시작할 때 해결해야 할 문제를 설명할 때 상향식 접근 방식을 사용합니다. 첫 번째 문제는 객체의 적절한 표현을 정의하는 것입니다.

2 장에서는 점, 기본 기하학적 모양과 객체 윤곽, 모양 표현과 같은 일반적인 객체 모양 표현을 설명합니다. 다음 문제는 추적기의 입력으로 사용되는 이미지 기능을 선택하는 것입니다.

3 장에서 우리는 객체 추적에 일반적으로 사용되는 색상, 움직임, 모서리 등과 같은 다양한 이미지 특징을 논의한다. 거의 모든 추적 알고리즘은 첫 번째 프레임 또는 모든 프레임에서 개체를 검색해야합니다.

4 장에서는 장면에서 물체를 탐지하기 위한 일반적인 전략을 요약합니다. 특정 추적 알고리즘의 적합성은 객체 모양, 객체 모양, 객체수, 객체 및 카메라 동작 및 조명 조건에 따라 다릅니다.

5 장에서는 기존 추적 방법을 범주화하고 설명하며 각 범주 끝에 요약 섹션의 장단점을 설명합니다.

6 장에서는 객체 추적과 관련된 중요한 문제에 대해 설명합니다.

7 장에서는 연구 추적의 미래 방향을 제시한다.

마지막으로, 결론적인 발언은 8 장에서 설명한다.

**2. OBJECT REPRESENTATION**

In a tracking scenario, an object can be defined as anything that is of interest for further analysis. For instance, boats on the sea, fish inside an aquarium, vehicles on a road, planes in the air, people walking on a road, or bubbles in the water are a set of objects that may be important to track in a specific domain. Objects can be represented by their shapes and appearances. In this section, we will first describe the object shape representations commonly employed for tracking and then address the joint shape and appearance representations.

추적 시나리오에서 개체는 추가 분석에 관심있는 것으로 정의될 수 있습니다. 예를 들어, 바다에 있는 보트, 수족관 안에 있는 물고기, 길에있는 차량, 공중에있는 비행기, 길을 걷는 사람들 또는 물 속에 있는 거품은 특정 도메인에서 추적하는 데 중요한 개체일 수 있습니다. 물체는 모양과 외모로 나타낼 수 있습니다. 이 섹션에서는 추적에 일반적으로 사용되는 객체 모양 표현을 먼저 설명한 다음 접합 모양 및 모양 표현에 대해 설명합니다.

—Points. The object is represented by a point, that is, the centroid (Figure 1(a)) [Veenman et al. 2001] or by a set of points (Figure 1(b)) [Serby et al. 2004]. In general, the point representation is suitable for tracking objects that occupy small regions in an image.(see Section 5.1).

포인트.   
객체는 점, 즉 중심 (그림 1 (a)) [Veenman et al. 2001] 또는 일련의 점 (그림 1 (b)) [Serby et al. 2004]. 일반적으로 포인트 표현은 이미지의 작은 영역을 차지하는 객체를 추적하는 데 적합합니다 (5.1 참조).

—Primitive geometric shapes. Object shape is represented by a rectangle, ellipse (Figure 1(c), (d) [Comaniciu et al. 2003], etc. Object motion for such representations is usually modeled by translation, affine, or projective (homography) transformation (see Section 5.2 for details). Though primitive geometric shapes are more suitable for representing simple rigid objects, they are also used for tracking nonrigid objects.

프리미티브 기하학적 shape.   
객체 모양은 직사각형, 타원 (그림 1 (c), (d) [Comaniciu et al. 2003] 등)으로 표현된다. 그러한 표현에 대한 객체 모션은 보통 번역, affine 또는 projective (homography) 변환에 의해 모델링된다. 자세한 내용은 섹션 5.2) 원시 기하학적 모양은 단순한 rigid 를 표현하는 데 더 적합하지만 비 강체를 추적하는데도 사용됩니다.

—Object silhouette and contour. Contour representation defines the boundary of an object (Figure 1(g), (h). The region inside the contour is called the silhouette of the object (see Figure 1(i) ). Silhouette and contour representations are suitable for tracking complex nonrigid shapes [Yilmaz et al. 2004].

오브젝트 실루엣 및 윤곽. 윤곽 표현은 객체의 경계를 정의합니다 (그림 1 (g), (h)). 윤곽 내부의 영역을 객체의 실루엣이라고합니다 (그림 1 (i) 참조) 실루엣 및 윤곽 표현은 복잡한 nonrigid shape [Yilmaz 그 외 여러분 2004 년].

—Articulated shape models. Articulated objects are composed of body parts that are held together with joints. For example, the human body is an articulated object with torso, legs, hands, head, and feet connected by joints. The relationship between the parts are governed by kinematic motion models, for example, joint angle, etc. In order to represent an articulated object, one can model the constituent parts using cylinders or ellipses as shown in Figure 1(e).

Articulated 모양 모델. 굴절된 물체는 조인트와 함께 유지되는 몸체 부분으로 구성됩니다. 예를 들어, 인체는 관절로 연결된 몸통, 다리, 손, 머리, 발을 가진 관절 된 물체입니다. 부품들 사이의 관계는 관절각과 같은 운동학적 동작 모델에 의해 관리됩니다. 관절 객체를 나타 내기 위해 그림 1 (e)와 같이 원통 또는 타원을 사용하여 구성 부품을 모델링 할 수 있습니다.

—Skeletal models. Object skeleton can be extracted by applying medial axis transform to the object silhouette [Ballard and Brown 1982, Chap. 8]. This model is commonly used as a shape representation for recognizing objects [Ali and Aggarwal 2001]. Skeleton representation can be used to model both articulated and rigid objects (see Figure 1(f).

골격 모델. 객체 골격은 객체 실루엣에 중간 축 변환을 적용하여 추출 할 수있다 [Ballard and Brown 1982, Chap. 8]. 이 모델은 일반적으로 객체를 인식하기위한 모양 표현으로 사용됩니다 [Ali and Aggarwal 2001]. Skeleton representation은 articulated 과 rigid 를 모델링하는 데 사용할 수 있습니다 (그림 1 (f) 참조).

|  |
| --- |
|  |
| Fig. 1. Object representations. (a) Centroid, (b) multiple points, (c) rectangular patch, (d) elliptical patch, (e) part-based multiple patches, (f) object skeleton, (g) complete object contour, (h) control points on object contour, (i) object silhouette.  1. 객체 표현. (a) 중심점, (b) 다중 점, (c) 직사각형 패치, (d) 타원 패치, (e) 부분 기반 다중 패치, (f) 객체 골격, (g) 완전한 객체 윤곽선, 오브젝트 윤곽선, (i) 오브젝트 실루엣. |

There are a number of ways to represent the appearance features of objects. Note that shape representations can also be combined with the appearance representations [Cootes et al. 2001] for tracking. Some common appearance representations in the context of object tracking are:

객체의 외양 특징을 표현하는 데는 여러 가지 방법이 있습니다. 모양 표현은 추적을 위한 모양 표현과 결합 될 수도 있습니다. [Cootes et al. 2001] 객체 추적의 맥락에서 일반적으로 나타나는 표현은 다음과 같습니다.

—Probability densities of object appearance. The probability density estimates of the object appearance can either be parametric, such as Gaussian [Zhu and Yuille 1996] and a mixture of Gaussians [Paragios and Deriche 2002], or nonparametric, such as Parzen windows [Elgammal et al. 2002] and histograms [Comaniciu et al. 2003]. The probability densities of object appearance features (color, texture) can be computed from the image regions specified by the shape models (interior region of an ellipse or a contour).

물체 출현의 확률 밀도. 객체 출현의 확률 밀도 추정치는 Gaussian [Zhu and Yuille 1996] 및 Gaussians [Paragios and Deriche 2002]의 혼합과 같은 매개 변수이거나 Parzen 창 [Elgammal et al. 2002] 및 히스토그램 [Comaniciu et al. 2003]. 객체 출현 피쳐 (색, 텍스처)의 확률 밀도는 형상 모델 (타원 또는 윤곽의 내부 영역)에 의해 지정된 이미지 영역으로부터 계산 될 수 있습니다.

—Templates. Templates are formed using simple geometric shapes or silhouettes [Fieguth and Terzopoulos 1997]. An advantage of a template is that it carries both spatial and appearance information. Templates, however, only encode the object appearance generated from a single view. Thus, they are only suitable for tracking objects whose poses do not vary considerably during the course of tracking.

템플릿. 템플릿은 간단한 기하학적 모양이나 실루엣을 사용하여 형성됩니다 [Fieguth and Terzopoulos 1997]. 템플리트의 장점은 공간 및 모양 정보를 모두 전달한다는 것입니다. 그러나 템플릿은 단일보기에서 생성 된 객체 모양 만 인코딩합니다. 따라서 추적하는 동안 포즈가 크게 변하지 않는 개체를 추적하는 데만 적합합니다.

—Active appearance models. Active appearance models are generated by simultaneously modeling the object shape and appearance [Edwards et al. 1998]. In general, the object shape is defined by a set of landmarks. Similar to the contour-based representation, the landmarks can reside on the object boundary or, alternatively, they can reside inside the object region. For each landmark, an appearance vector is stored which is in the form of color, texture, or gradient magnitude. Active appearance models require a training phase where both the shape and its associated appearance is learned from a set of samples using, for instance, the principal component analysis.

적극적인 외관 모델. 활성 모양 모델은 객체 모양과 모양을 동시에 모델링하여 생성됩니다 [Edwards et al. 1998]. 일반적으로 객체 모양은 일련의 표식으로 정의됩니다. 등고선 기반의 표현과 유사하게, 랜드 마크는 객체 경계 상에 존재할 수 있거나, 대안으로 그들은 객체 영역 내부에 상주 할 수있다. 각각의 표식에 대하여, 색, 텍스쳐 또는 그레디언트 크기의 형태 인 외형 벡터가 저장된다. 능동형 모델은 예를 들어 주 구성 요소 분석을 사용하여 모양 세트와 관련 모양을 학습하는 교육 단계가 필요합니다.

—Multiview appearance models. These models encode different views of an object. One approach to represent the different object views is to generate a subspace from the given views. Subspace approaches, for example, Principal Component Analysis (PCA) and Independent Component Analysis (ICA), have been used for both shape and appearance representation [Mughadam and Pentland 1997; Black and Jepson 1998].

멀티 뷰 출현 모델. 이 모델은 객체의 다른 뷰를 인코딩합니다. 다른 객체 뷰를 표현하는 한 가지 방법은 주어진 뷰에서 부분 공간을 생성하는 것입니다. 예를 들어 PCA (Principal Component Analysis) 및 ICA (Independent Component Analysis)와 같은 부분 공간 접근법이 모양 및 모양 표현에 사용되었습니다 [Mughadam and Pentland 1997; Black and Jepson 1998].

Another approach to learn the different views of an object is by training a set of classifiers, for example, the support vector machines [Avidan 2001] or Bayesian networks [Park and Aggarwal 2004]. One limitation of multiview appearance models is that the appearances in all views are required ahead of time.

객체의 다양한 뷰를 학습하는 또 다른 접근법은 classifiers 를 트레이닝하고, (예를 들어, 지원 벡터 시스템 [Avidan 2001] 또는 베이지안 네트워크 [Park and Aggarwal 2004] )

multiview appearance models의 한 가지 제한 사항은 모든 뷰의 모양이 미리 필요하다는 것입니다.

In general, there is a strong relationship between the object representations and the tracking algorithms. Object representations are usually chosen according to the application domain. For tracking objects, which appear very small in an image, point representation is usually appropriate.

For instance, Veenman et al. [2001] use the point representation to track the seeds in a moving dish sequence.

Similarly, Shafique and Shah [2003] use the point representation to track distant birds. For the objects whose shapes can be approximated by rectangles or ellipses, primitive geometric shape representations are more appropriate. Comaniciu et al. [2003] use an elliptical shape representation and employ a color histogram computed from the elliptical region for modeling the appearance. In 1998, Black and Jepson used eigenvectors to represent the appearance. The eigenvectors were generated from rectangular object templates. For tracking objects with complex shapes, for example, humans, a contour or a silhouettebased representation is appropriate. Haritaoglu et al. [2000] use silhouettes for object tracking in a surveillance application.

일반적으로 객체 표현과 추적 알고리즘 간에는 강력한 관계가 있습니다. 객체 표현은 대개 응용 프로그램 도메인에 따라 선택됩니다. 이미지에서 매우 작게 나타나는 추적 객체의 경우 일반적으로 포인트 표현이 적합합니다. 예를 들어, Veenman et al. [2001] 등은 moving dish sequence에서 track the seeds 하는것으로 point representation을 사용합니다 (??)

유사하게, Shafique와 Shah [2003]는 멀리 떨어진 새들을 추적하기 위해 포인트 표현을 사용한다. 직사각형 또는 타원으로 근사 될 수있는 객체의 경우 기본 기하학적 모양 표현이 더 적합합니다. Comaniciu et al. [2003] 타원형 모양 표현을 사용하고 모양을 모델링하기 위해 타원형 영역에서 계산된 색 막대 그래프를 사용합니다. 1998 년 Black과 Jepson은 외형을 나타내기 위해 고유 벡터를 사용했습니다. 고유 벡터는 직사각형 객체 템플릿에서 생성되었습니다. 예를 들어 인간과 같이 복잡한 모양을 가진 객체를 추적하려면 윤곽선이나 실루엣을 기반으로 한 표현이 적합합니다. Haritaoglu et al. [2000] 감시 응용 프로그램에서 객체 추적을 위한 실루엣을 사용합니다.

**3. FEATURE SELECTION FOR TRACKING**

**추적을 위한 기능 선택**

Selecting the right features plays a critical role in tracking. In general, the most desirable property of a visual feature is its uniqueness so that the objects can be easily distinguished in the feature space. Feature selection is closely related to the object representation. For example, color is used as a feature for histogram-based appearance representations, while for contour-based representation, object edges are usually used as features. In general, many tracking algorithms use a combination of these features. The details of common visual features are as follows.

올바른 기능을 선택하는 것은 추적에 중요한 역할을 합니다. 일반적으로 시각적 피쳐의 가장 바람직한 속성은 피쳐 공간에서 개체를 쉽게 구별 할 수 있도록 고유성입니다. 피쳐 선택은 객체 표현과 밀접하게 관련됩니다. 예를 들어, 색상은 히스토그램 기반 모양 표현에 대한 피쳐로 사용되고, 윤곽 기반 표현의 경우 객체 모서리는 일반적으로 피쳐로 사용됩니다. 일반적으로 많은 추적 알고리즘은 이러한 기능의 조합을 사용합니다. 일반적인 시각적 기능의 세부 사항은 다음과 같습니다

—Color. The apparent color of an object is influenced primarily by two physical factors, 1) the spectral power distribution of the illuminant and 2) the surface reflectance properties of the object. In image processing, the RGB (red, green, blue) color space is usually used to represent color. However, the RGB space is not a perceptually uniform color space, that is, the differences between the colors in the RGB space do not correspond to the color differences perceived by humans [Paschos 2001]. Additionally, the RGB dimensions are highly correlated. In contrast, L∗u∗v∗ and L∗a∗b∗ are perceptually uniform color spaces, while HSV (Hue, Saturation, Value) is an approximately uniform color space. However, these color spaces are sensitive to noise [Songet al. 1996]. In summary, there is no last word on which color space is more efficient, therefore a variety of color spaces have been used in tracking.

컬러 - 물체의 겉보기 색상은 주로 1) 광원의 스펙트럼 파워 분포와 2) 물체의 표면 반사 특성의 두 가지 물리적 요인에 의해 영향을받습니다. 이미지 처리에서 RGB (빨강, 초록, 파랑) 색 공간은 대개 색을 나타내기 위해 사용됩니다. 그러나 RGB 공간은 지각적으로 균일한 색 공간이 아니며, 즉 RGB 공간의 색상 차이는 인간이 인식하는 색상 차이와 일치하지 않습니다 [Paschos 2001]. 또한 RGB 치수는 상호 연관성이 높습니다. 대조적으로, L \* u \* v \*와 L \* a \* b \*는 지각적으로 균일한 색상공간이지만 HSV (Hue, Saturation, Value)는 거의 균일한 색상 공간입니다. 그러나 이러한 색상 공간은 노이즈에 민감합니다 [Songet al. 1996]. 요약하면, 색상 공간이 더 효율적이라는 마지막 단어가 없으므로 다양한 색상 공간이 추적에 사용되었습니다.

—Edges. Object boundaries usually generate strong changes in image intensities. Edge detection is used to identify these changes. An important property of edges is that they are less sensitive to illumination changes compared to color features. Algorithms that track the boundary of the objects usually use edges as the representative feature. Because of its simplicity and accuracy, the most popular edge detection approach is the Canny Edge detector [Canny 1986]. An evaluation of the edge detection algorithms is provided by Bowyer et al. [2001].

엣지 - 객체 경계는 일반적으로 이미지 강도에서 강력한 변화를 생성합니다. 가장자리 감지는 이러한 변경을 식별하는 데 사용됩니다. 가장자리의 중요한 특성은 색상 특징에 비해 조명 변화에 덜 민감하다는 것입니다. 객체의 경계를 추적하는 알고리즘은 대개 모서리를 대표 기능으로 사용합니다. 단순성과 정확성 때문에 가장 인기있는 에지 감지 방식은 Canny Edge detector [Canny 1986]입니다. 에지 검출 알고리즘의 평가는 Bowyer et al. [2001].

—Optical Flow. Optical flow is a dense field of displacement vectors which defines the translation of each pixel in a region. It is computed using the brightness constraint, which assumes brightness constancy of corresponding pixels in consecutive frames [Horn and Schunk 1981]. Optical flow is commonly used as a feature in motion-based segmentation and tracking applications. Popular techniques for computing dense optical flow include methods by Horn and Schunck [1981], Lucas and Kanade [1981], Black and Anandan [1996], and Szeliski and Couglan [1997]. For the performance evaluation of the optical flow methods, we refer the interested reader to the survey by Barron et al. [1994].

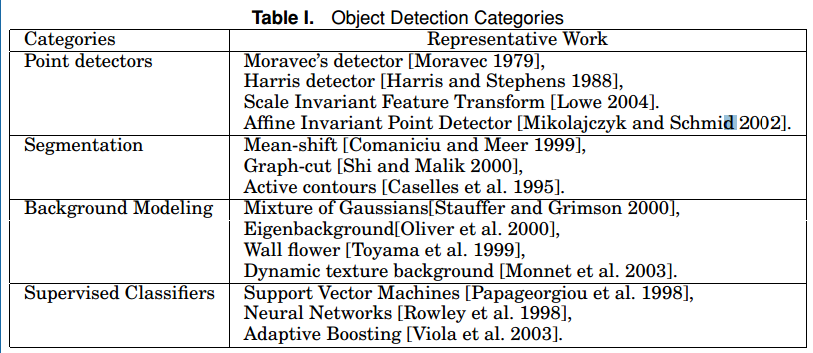
옵티컬 플로우 - 광학 흐름은 영역 내의 각 픽셀의 변환을 정의하는 변위 벡터의 조밀한 필드입니다. 이는 연속 프레임에서 대응하는 픽셀의 밝기 불변성을 가정하는 밝기 제한 조건을 사용하여 계산됩니다 [Horn and Schunk 1981]. 옵티컬 플로우는 일반적으로 동작 기반 세분화 및 추적 애플리케이션의 기능으로 사용됩니다. 조밀 한 옵티컬 플로우 계산을위한 대중적인 기술로는 Horn and Schunck [1981], Lucas and Kanade [1981], Black and Anandan [1996], Szeliski and Couglan [1997] 등이 있습니다. 옵티컬 플로우 방식의 성능 평가를 위해 관심있는 독자를 Barron et al. [1994] 의 서베이에 회부합니다

—Texture. Texture is a measure of the intensity variation of a surface which quantifies properties such as smoothness and regularity. Compared to color, texture requires a processing step to generate the descriptors. There are various texture descriptors: Gray-Level Cooccurrence Matrices (GLCM’s) [Haralick et al. 1973] (a 2D histogram which shows the cooccurrences of intensities in a specified direction and distance), Law’s texture measures [Laws 1980] (twenty-five 2D filters generated from five 1D filters corresponding to level, edge, spot, wave, and ripple), wavelets [Mallat 1989] (orthogonal bank of filters), and steerable pyramids [Greenspan et al. 1994]. Similar to edge features, the texture features are less sensitive to illumination changes compared to color.

텍스쳐 - 텍스처는 매끄럽고 규칙성 같은 속성을 정량화하는 표면의 강도(intensity) 변화를 측정 한것입니다. 색상과 비교할 때 텍스처는 설명자를 생성하는 처리 단계가 필요합니다. 다양한 텍스처 서술자가 있다 : 그레이 레벨 Cooccurrence 매트릭스 (GLCM의) [Haralick 외. 1973] (특정 방향과 거리에서 강도의 공적을 보여주는 2D 히스토그램) Law의 질감 측정 [Laws 1980] (레벨, 에지, 스팟, 웨이브 및 리플에 해당하는 5 개의 1D 필터에서 생성 된 25 개의 2D 필터 ), 웨이브 렛 [Mallat 1989] (필터의 직교 뱅크), 조종 가능한 피라미드 [Greenspan et al. 1994]. 에지 피처와 유사하게 텍스처 피처는 색상에 비해 조명 변화에 덜 민감합니다.

Mostly features are chosen manually by the user depending on the application domain. However, the problem of automatic feature selection has received significant attention in the pattern recognition community. Automatic feature selection methods can be divided into filter methods and wrapper methods [Blum and Langley 1997]. The filter methods try to select the features based on a general criteria, for example, the features should be uncorrelated. The wrapper methods select the features based on the usefulness of the features in a specific problem domain, for example, the classification performance using a subset of features. Principal Component Analysis (PCA) is an example of the filter methods for the feature reduction. PCA involves transformation of a number of (possibly) correlated variables into a (smaller) number of uncorrelated variables called the principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. A wrapper method of selecting the discriminatory features for tracking a particular class of objects is the Adaboost [Tieu and Viola 2004] algorithm. Adaboost is a method for finding a strong classifier based on a combination of moderately inaccurate weak classifiers. Given a large set of features, one classifier can be trained for each feature. Adaboost, as discussed in Sections 4.4, will discover a weighted combination of classifiers (representing features) that maximize the classification performance of the algorithm. The higher the weight of the feature, the more discriminatory it is. One can use the first n highest-weighted features for tracking

대부분의 기능은 응용 프로그램 도메인에 따라 사용자가 수동으로 선택합니다. 그러나, 자동 특징 선택의 문제점은 패턴 인식 커뮤니티에서 주목을 받고있다. 자동 기능 선택 방법은 필터 방법과 래퍼 방법으로 나눌 수 있습니다 [Blum and Langley 1997]. 필터 메소드는 일반적인 기준에 따라 기능을 선택하려고 시도합니다. 예를 들어, 기능은 상관되지 않아야합니다. 래퍼 메서드는 특정 문제 영역에서 기능의 유용성 (예 : 기능의 하위 집합을 사용하는 분류 성능)을 기반으로 기능을 선택합니다. PCA (Principal Component Analysis)는 기능 감소를 위한 필터 방법의 예입니다. PCA는 다수의 (가능하게) 상호 연관된 변수를 주 구성 요소라고하는 비 상관 변수의 (더 작은) 수로 변환하는 것을 포함합니다. 첫 번째 주성분은 가능한 한 많은 데이터의 변동성을 설명하며, 후속 각 구성 요소는 가능한 한 나머지 가변성의 많은 부분을 설명합니다. 특정 클래스의 객체를 추적하기위한 차별 특징을 선택하는 래퍼 메소드는 Adaboost [Tieu and Viola 2004] 알고리즘입니다. Adaboost는 정확하지 않은 약한 분류 기준을 조합하여 강력한 분류 기준을 찾는 방법입니다. 많은 기능 세트가 주어지면 각 기능에 대해 하나의 분류자를 학습 할 수 있습니다. 4.4 절에서 논의 된 Adaboost는 알고리즘의 분류 성능을 최대화하는 분류 자 ​​(특징을 나타내는)의 가중치 조합을 발견합니다. 지형지 물의 무게가 클수록 더 차별적입니다. 첫 번째 n 개의 가장 높은 가중치를 적용한 추적 기능을 사용할 수 있습니다.



Among all features, color is one of the most widely used feature for tracking. Comaniciu et al. [2003] use a color histogram to represent the object appearance. Despite its popularity, most color bands are sensitive to illumination variation. Hence in scenarios where this effect is inevitable, other features are incorporated to model object appearance. Cremers et al. [2003] use optical flow as a feature for contour tracking. Jepson et al. [2003] use steerable filter responses for tracking. Alternatively, a combination of these features is also utilized to improve the tracking performance.

모든 기능 중에서, 색상은 추적에 가장 널리 사용되는 기능 중 하나입니다. Comaniciu et al. [2003] 에서 색상 히스토그램을 사용하여 객체 모양을 나타냅니다. 그 인기에도 불구하고 대부분의 컬러 밴드는 조명 변화에 민감합니다. 따라서 이 효과가 불가피한 시나리오에서는 다른 피쳐가 모델 객체 모양에 통합됩니다. Cremers et al. [2003] 윤곽 추적을 위한 기능으로 옵티컬 플로우를 사용합니다. Jepson et al. [2003] 추적을 위해 조종 가능한 필터 응답을 사용합니다. 대안적으로, 이들 특징들의 조합은 또한 추적 성능을 향상시키는 데 이용된다.

**4. OBJECT DETECTION**

**객체 검출**

Every tracking method requires an object detection mechanism either in every frame or when the object first appears in the video. A common approach for object detection is to use information in a single frame. However, some object detection methods make use of the temporal information computed from a sequence of frames to reduce the number of false detections. This temporal information is usually in the form of frame differencing, which highlights changing regions in consecutive frames. Given the object regions in the image, it is then the tracker’s task to perform object correspondence from one frame to the next to generate the tracks.

모든 추적 방법에는 모든 프레임에서 또는 객체가 비디오에 처음 나타날 때 객체 탐지 ​​메커니즘이 필요합니다. 객체 감지를 위한 일반적인 접근법은 단일 프레임에서 정보를 사용하는 것입니다. 그러나, 일부 객체 검출 방법은 잘못된 검출의 수를 줄이기 위해 일련의 프레임으로부터 계산 된 시간 정보를 이용한다. 이 시간 정보는 일반적으로 연속 프레임의 변화하는 영역을 강조 표시하는 프레임 차등의 형태입니다. 이미지의 오브젝트 영역이 주어지면 한 프레임에서 다음 프레임으로 오브젝트 대응을 수행하여 트랙을 생성하는 것은 추적 프로그램의 태스크입니다.

We tabulate several common object detection methods in Table I. Although the object detection itself requires a survey of its own, here we outline the popular methods in the context of object tracking for the sake of completeness.

우리는 표 1에있는 몇 가지 공통적인 물체 감지 방법을 표로 작성합니다. 물체 감지 자체는 자체 조사를 필요로하지만 여기에서는 완전성을 위해 물체 추적과 관련하여 널리 사용되는 방법을 설명합니다.

**4.1. Point Detectors**

Point detectors are used to find interest points in images which have an expressive texture in their respective localities. Interest points have been long used in the context of motion, stereo, and tracking problems. A desirable quality of an interest point is its invariance to changes in illumination and camera viewpoint. In the literature, commonly used interest point detectors include Moravec’s interest operator [Moravec 1979], Harris interest point detector [Harris and Stephens 1988], KLT detector [Shi and Tomasi 1994], and SIFT detector [Lowe 2004]. For a comparative evaluation of interest point detectors, we refer the reader to the survey by Mikolajczyk and Schmid [2003].

포인트 검출기는 각각의 지역에서 표현적인 질감을 갖는 이미지에서 관심 지점을 찾는데 사용됩니다. 관심 지점은 모션, 스테레오 및 추적 문제와 관련하여 오랫동안 사용되어 왔습니다. 관심 지점의 바람직한 품질은 조명 및 카메라 시점의 변화에 ​​대한 불변성입니다. 문헌에서 흔히 사용되는 관심 지점 검출기에는 Moravec의 관심 연산자 [Moravec 1979], 해리스 관심 지점 검출기 [Harris and Stephens 1988], KLT 검출기 [Shi and Tomasi 1994] 및 SIFT 검출기 [Lowe 2004]가 있습니다. 관심 지점 탐지기의 비교 평가를 위해 독자는 Mikolajczyk 및 Schmid [2003]에 의한 조사를 참조하십시오.

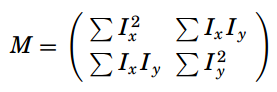
To find interest points, Moravec’s operator computes the variation of the image intensities in a 4 × 4 patch in the horizontal, vertical, diagonal, and antidiagonal directions and selects the minimum of the four variations as representative values for the window. A point is declared interesting if the intensity variation is a local maximum in a 12 × 12 patch.

관심 지점을 찾기 위해 Moravec의 연산자는 수평, 수직, 대각선 및 대각선 방향의 4x4 패치에서 이미지 강도의 변화를 계산하고 4 가지 유사도의 최소값을 창의 대표값으로 선택합니다. 강도 변화가 12 × 12 패치의 최대값이라면 요점이 흥미롭습니다.

|  |
| --- |
|  |
| Fig. 2. Interest points detected by applying (a) the Harris, (b) the KLT, and (c) SIFT operators. |

The Harris detector computes the first order image derivatives, (Ix , I y ), in x and y directions to highlight the directional intensity variations, then a second moment matrix, which encodes this variation, is evaluated for each pixel in a small neighborhood:

Harris 검출기는 방향 세기 변화를 강조하기 위해 x, y 방향으로 1 차 이미지 도함수 (1x, 1y)를 계산 한 다음이 유사도를 인코딩하는 2 차 모멘트 행렬을 작은 이웃의 각 픽셀에 대해 평가합니다.



An interest point is identified using the determinant and the trace of M which measures the variation in a local neighborhood , where k is constant.

관심 포인트(Interest point)는 local neighborhood 에서의 변이를 측정하는 행렬식과 M의 추적을 사용하여 식별된다. 여기서 k는 일정하다.

The interest points are marked by thresholding R after applying nonmaxima suppression (see Figure 2(a) for results).

Interest point은 최대가 아닌 nonmaxima 를 적용한 후 임계값 R로 표시됩니다 (결과에 대해서는 그림 2 (a) 참조).

The source code for Harris detector is available at HarrisSrc.

해리스 탐지기의 소스 코드는 HarrisSrc에서 제공됩니다.

The same moment matrix M given in Equation (1) is used in the interest point detection step of the KLT tracking method. Interest point confidence, R, is computed using the minimum eigenvalue of M, . Interest point candidates are selected by thresholding R.

수학 식 1에서 주어진 동일한 모멘트 행렬 M은 KLT 추적 방법의 interest point detection step에서 사용된다. Interest point 신뢰도(컨피던스) R은 M,의 최소 고유치를 사용하여 계산됩니다.

Among the candidate points, KLT eliminates the candidates that are spatially close to each other (Figure 2(b)). Implementation of the KLT detector is available at KLTSrc.

후보 points중에서 KLT는 공간적으로 가까운 후보를 제거합니다 (그림 2 (b)).

KLTS 검출기의 구현은 KLTSrc에서 가능합니다.

Quantitatively both Harris and KLT emphasize the intensity variations using very similar measures. For instance, R in Harris is related to the characteristic polynomial used for finding the eigenvalues of , while KLT computes the eigenvalues directly. In practice, both of these methods find almost the same interest points. The only difference is the additional KLT criterion that enforces a predefined spatial distance between detected interest points.

양적으로 Harris와 KLT는 매우 유사한 측정법을 사용하여 강도(intensity)변화를 강조(emphasize)합니다.

예를 들어 해리스의 R은 의 고유값을 찾는 데 사용되는 특성 다항식과 관련이 있으며 KLT는 고유 값을 직접 계산합니다.

실제로 이러한 두 가지 방법 모두 거의 동일한 관심 지점을 찾습니다. 유일한 차이점은 탐지 된 관심 지점 사이에 미리 정의된 공간 거리를 적용하는 추가 KLT 기준입니다.

In theory, the M matrix is invariant to both rotation and translation. However, it is not invariant to affine or projective transformations. In order to introduce robust detection of interest points under different transformations, Lowe [2004] introduced the SIFT (Scale Invariant Feature Transform) method which is composed of four steps. First, a scale space is constructed by convolving the image with Gaussian filters at different scales. Convolved images are used to generate difference-of-Gaussians (DoG) images. Candidate interest points are then selected from the minima and maxima of the DoG images across scales. The next step updates the location of each candidate by interpolating the color values using neighboring pixels. In the third step, low contrast candidates as well as the candidates along the edges are eliminated. Finally, remaining interest points are assigned orientations based on the peaks in the histograms of gradient directions in a small neighborhood around a candidate point. SIFT detector generates a greater number of interest points compared to other interest point detectors. This is due to the fact that the interest points at different scales and different resolutions (pyramid) are accumulated. Empirically, it has been shown in Mikolajczyk and Schmid [2003] that SIFT outperforms most point detectors and is more resilient to image deformations. Implementation of the SIFT detector is available at SIFTSrc.

이론적으로, M 행렬은 회전과 변환 모두에 대해 변하지 않습니다. 그러나, 아핀 또는 사영 변환에는 불변하지 않습니다. Lowe [2004]는 서로 다른 변환 하에서 관심 점의 강력한 검출을 도입하기 위해 4 단계로 구성된 SIFT (Scale Invariant Feature Transform) 방법을 도입했다. 첫째, 서로 다른 스케일의 가우시안 필터로 이미지를 컨벌루션하여 스케일 공간을 구성합니다. 컨볼루션 된 이미지는 Gaussian (DoG) 이미지를 생성하는 데 사용됩니다. 후보 관심 지점은 DoG 이미지의 최소 및 최대에서 선택됩니다. 다음 단계는 인접 픽셀을 사용하여 색상 값을 보간하여 각 후보의 위치를 ​​업데이트합니다. 제 3 단계에서, 낮은 콘트라스트 후보들 및 에지를 따라 후보들이 제거된다. 마지막으로, 나머지 관심 지점은 후보 지점 주변의 작은 이웃에서 기울기 방향의 히스토그램의 피크를 기반으로 방향이 지정됩니다. SIFT 검출기는 다른 관심 지점 검출기와 비교하여 많은 수의 관심 지점을 생성합니다. 이것은 다른 척도와 다른 해상도 (피라미드)에서 관심 지점이 누적되기 때문입니다. 경험적으로, Mikolajczyk와 Schmid [2003]에서는 SIFT가 대부분의 점 검출기를 능가하며 이미지 변형에보다 탄력적이라는 것을 보여주었습니다. SIFT 검출기의 구현은 SIFTSrc에서 가능하다.

|  |
| --- |
|  |
| Fig. 3. Mixture of Gaussian modeling for background subtraction. (a) Image from a sequence in which a person is walking across the scene. (b) The mean of the highest-weighted Gaussians at each pixels position. These means represent the most temporally persistent per-pixel color and hence should represent the stationary background. (c) The means of the Gaussian with the second-highest weight; these means represent colors that are observed less frequently. (d) Background subtraction result. The foreground consists of the pixels in the current frame that matched a low-weighted Gaussian.  배경 차감을위한 가우시안 모델링의 혼합. (a) 장면을 가로 질러 걷는 장면의 이미지. (b) 각 픽셀 위치에서 가장 높은 가중치 가우시안의 평균. 이러한 수단은 픽셀 당 가장 일시적으로 지속되는 색상을 나타내므로 고정 된 배경을 나타내야합니다. (c) 두 번째로 큰 가중치를 갖는 가우시안의 평균; 이러한 수단은 덜 자주 관찰되는 색상을 나타냅니다. (d) 배경 뺄셈 결과. 전경은 가중치가 낮은 가우시안과 일치하는 현재 프레임의 픽셀로 구성됩니다. |

**4.2. Background Subtraction**

Object detection can be achieved by building a representation of the scene called the background model and then finding deviations from the model for each incoming frame. Any significant change in an image region from the background model signifies a moving object. The pixels constituting the regions undergoing change are marked for further processing. Usually, a connected component algorithm is applied to obtain connected regions corresponding to the objects. This process is referred to as the background subtraction.

객체 검출은 배경 모델이라고 불리는 장면의 표현을 구축한 다음 각 수신 프레임에 대해 모델로부터 편차를 찾아냄으로써 얻을 수 있습니다. 배경 이미지 모델에서 이미지 영역의 중요한 변화는 움직이는 물체를 의미합니다. 변화를 겪고있는 영역을 구성하는 픽셀은 추가 처리를 위해 표시됩니다. 일반적으로 객체에 해당하는 연결된 영역을 얻기 위해 연결된 컴포넌트 알고리즘이 적용됩니다. 이 프로세스를 백그라운드 뺄셈이라고합니다.

Frame differencing of temporally adjacent frames has been well studied since the late 70s [Jain and Nagel 1979]. However, background subtraction became popular following the work of Wren et al. [1997]. In order to learn gradual changes in time, Wren et al. propose modeling the color of each pixel, I (x, y), of a stationary background with a single 3D (Y, U, and V color space) Gaussian, I (x, y) ∼ N(μ(x, y), (x, y)). The model parameters, the mean μ(x, y) and the covariance (x, y), are learned from the color observations in several consecutive frames. Once the background model is derived, for every pixel (x, y) in the input frame, the likelihood of its color coming from N(μ(x, y), (x, y)) is computed, and the pixels that deviate from the background model are labeled as the foreground pixels. However, a single Gaussian is not a good model for outdoor scenes [Gao et al. 2000] since multiple colors can be observed at a certain location due to repetitive object motion, shadows, or reflectance. A substantial improvement in background modeling is achieved by using multimodal statistical models to describe per-pixel background color. For instance, Stauffer and Grimson [2000] use a mixture of Gaussians to model the pixel color. In this method, a pixel in the current frame is checked against the background model by comparing it with every Gaussian in the model until a matching Gaussian is found. If a match is found, the mean and variance of the matched Gaussian is updated, otherwise a new Gaussian with the mean equal to the current pixel color and some initial variance is introduced into the mixture. Each pixel is classified based on whether the matched distribution represents the background process. Moving regions, which are detected using this approach, along with the background models are shown in Figure 3.

시간적으로 인접한 프레임의 프레임 차이는 70 년대 후반부터 잘 연구되어왔다 [Jain and Nagel 1979]. 그러나 백그라운드 감산은 Wren et al. [1997]. 시간의 점진적인 변화를 배우기 위해서, Wren et al. I (x, y) ~ N (μ (x, y))의 단일 3D (Y, U 및 V 색상 공간) 가우시안을 갖는 정지 된 배경의 각 픽셀 I (x, y) , (x, y)). 모델 매개 변수, 평균 μ (x, y) 및 공분산 (x, y)은 여러 연속 프레임의 색상 관측치로부터 학습됩니다. 일단 배경 모델이 유도되면 입력 프레임의 모든 픽셀 (x, y)에 대해 N (μ (x, y), (x, y))에서 오는 색상의 우도가 계산되고, 배경 모델의 픽셀은 전경 픽셀로 표시됩니다. 그러나 하나의 가우시안은 야외 장면에 적합한 모델이 아닙니다 [Gao et al. 2000]. 반복되는 물체의 움직임, 그림자, 반사 등으로 인해 특정 위치에서 여러 색상을 볼 수 있기 때문입니다. 멀티 모달 통계 모델을 사용하여 픽셀 별 배경색을 설명함으로써 백그라운드 모델링이 크게 향상되었습니다. 예를 들어, Stauffer and Grimson [2000]은 픽셀 색상을 모델링하기 위해 Gaussian의 혼합을 사용합니다. 이 방법에서 현재 프레임의 픽셀은 일치하는 가우시안이 발견 될 때까지 모델의 모든 가우시안과 비교하여 백그라운드 모델과 대조하여 검사됩니다. 일치가 발견되면 일치하는 가우시안의 평균 및 분산이 업데이트되고, 그렇지 않은 경우 현재 픽셀 색상과 평균이 같은 새 가우시안과 일부 초기 분산이 혼합되어 도입됩니다. 각 픽셀은 일치 된 분포가 백그라운드 프로세스를 나타내는 지 여부에 따라 분류됩니다. 이 접근법을 사용하여 감지 된 이동 영역과 백그라운드 모델은 그림 3에 나와 있습니다.

Another approach is to incorporate region-based (spatial) scene information instead of only using color-based information. Elgammal and Davis [2000] use nonparametric kernel density estimation to model the per-pixel background. During the subtraction process, the current pixel is matched not only to the corresponding pixel in the background model, but also to the nearby pixel locations. Thus, this method can handle camera jitter or small movements in the background. Li and Leung [2002] fuse the texture and color features to perform background subtraction over blocks of 5 × 5 pixels. Since texture does not vary greatly with illumination changes, the method is less sensitive to illumination. Toyama et al. [1999] propose a three-tiered algorithm to deal with the background subtraction problem. In addition to the pixel-level subtraction, the authors use the region and the frame-level information. At the pixel level, the authors propose to use Wiener filtering to make probabilistic predictions of the expected background color. At the region level, foreground regions consisting of homogeneous color are filled in. At the frame level, if most of the pixels in a frame exhibit suddenly change, it is assumed that the pixel-based color background models are no longer valid. At this point, either a previously stored pixel-based background model is swapped in, or the model is reinitialized.

또 다른 접근법은 컬러 기반 정보를 사용하는 대신에 영역 기반 (공간적) 장면 정보를 통합하는 것이다. Elgammal과 Davis [2000]는 픽셀 별 배경을 모델링하기 위해 비모수적인 커널 밀도 추정을 사용한다. 감산 프로세스 동안, 현재 픽셀은 배경 모델의 대응 픽셀뿐만 아니라 근처 픽셀 위치에도 매치된다. 따라서이 방법은 카메라 지터 또는 배경에서의 작은 움직임을 처리 할 수 ​​있습니다. Li and Leung [2002]은 텍스처 및 색상 피쳐를 융합하여 5x5 픽셀 블록에 배경 차감을 수행합니다. 조명 변화에 따라 텍스처가 크게 변하지 않기 때문에 조명에 덜 민감합니다. Toyama et al. [1999]는 배경 빼기 문제를 다루기위한 3 단계 알고리즘을 제안한다. 제작자는 픽셀 단위의 뺄셈 외에 영역과 프레임 레벨 정보를 사용합니다. 픽셀 수준에서 저자는 예상되는 배경색을 확률 적으로 예측하기 위해 위너 필터링을 사용하는 방법을 제안합니다. 영역 레벨에서 동질적인 색상으로 구성된 전경 영역이 채워집니다. 프레임 수준에서 프레임의 대부분의 픽셀이 갑자기 변경되는 경우 픽셀 기반 색상 배경 ​​모델이 더 이상 유효하지 않은 것으로 간주됩니다. 이 시점에서 이전에 저장된 픽셀 기반 배경 모델이 스왑되거나 모델이 다시 초기화됩니다.

|  |
| --- |
|  |
| Fig. 4. Eigenspace decomposition-based background subtraction (space is constructed with objects in the FOV of camera): (a) an input image with objects, (b) reconstructed image after projecting input image onto the eigenspace, (c) difference image. Note that the foreground objects are clearly identifiable.  고유 공간 분해 기반의 배경 빼기 (공간은 카메라의 FOV에있는 객체로 구성됩니다.) : (a) 객체가있는 입력 이미지, (b) 입력 이미지를 고유 공간으로 투영 한 후 재구성 된 이미지, (c) 차이 이미지. 전경 물체는 명확하게 식별 할 수 있습니다. |

An alternate approach for background subtraction is to represent the intensity variations of a pixel in an image sequence as discrete states corresponding to the events in the environment. For instance, for tracking cars on a highway, image pixels can be in the background state, the foreground (car) state, or the shadow state. Rittscher et al. [2000] use Hidden Markov Models (HMM) to classify small blocks of an image as belonging to one of these three states. In the context of detecting light on and off events in a room, Stenger et al. [2001] use HMMs for the background subtraction. The advantage of using HMMs is that certain events, which are hard to model correctly using unsupervised background modeling approaches, can be learned using training samples.

배경 차감을위한 또 다른 접근법은 환경에서의 이벤트에 대응하는 개별 상태로서 이미지 시퀀스에서 픽셀의 강도 변화를 나타내는 것이다. 예를 들어, 고속도로에서 자동차를 추적하기 위해 이미지 픽셀은 배경 상태, 전경 (자동차) 상태 또는 그림자 상태 일 수 있습니다. Rittscher et al. [2000] 숨겨진 마르코프 모델 (HMM)을 사용하여 이미지의 작은 블록을이 세 가지 상태 중 하나에 속하는 것으로 분류합니다. 방의 조명 켜기 및 끄기 이벤트를 감지하는 맥락에서 Stenger et al. [2001] 배경 빼기를 위해 HMM을 사용합니다. HMM을 사용하는 이점은 감독되지 않은 백그라운드 모델링 접근법을 사용하여 올바르게 모델링하기 어려운 특정 이벤트를 교육 샘플을 사용하여 학습 할 수 있다는 것입니다.

Instead of modeling the variation of individual pixels, Oliver et al. [2000] propose a holistic approach using the eigenspace decomposition. For k input frames, I i : i = 1 ··· k, of size n× m, a background matrix B of size k ×l is formed by cascading m rows in each frame one after the other, where l = (n× m), and eigenvalue decomposition is applied to the covariance of B, C = BTB. The background is then represented by the most descriptive η eigenvectors, ui, where i <η< k, that encompass all possible illuminations in the field of view (FOV). Thus, this approach is less sensitive to illumination. The foreground objects are detected by projecting the current image to the eigenspace and finding the difference between the reconstructed and actual images. We show detected object regions using the eigenspace approach in Figure 4.

개별 픽셀의 변화를 모델링하는 대신, Oliver et al. [2000]은 고유 공간 분해를 사용하여 전체 론적 접근법을 제안한다. k 개의 입력 프레임들에 대해, 크기 n × m의 i i : i = 1 ... k에 대해, 크기 k × l의 배경 행렬 B는 각 프레임에서 m 개의 행들을 차례로 케스 케이 딩함으로써 형성된다. 여기서 l = (n × m), B의 공분산에 고유치 분해를 적용하면 C = BTB가된다. 그 배경은 시각적 (FOV)에서 가능한 모든 일루미네이션을 포함하는 가장 설명적인 η 고유 벡터, ui로 표현된다. 따라서이 접근법은 조명에 덜 민감합니다. 전경 물체는 현재 이미지를 고유 공간으로 투영하고 재구성 된 이미지와 실제 이미지의 차이점을 찾아내는 것으로 감지됩니다. 그림 4에서 고유 공간 접근법을 사용하여 탐지 된 객체 영역을 보여줍니다.

One limitation of the aforementioned approaches is that they require a static background. This limitation is addressed by Monnet et al. [2003], and Zhong and Sclaroff [2003]. Both of these methods are able to deal with time-varying background (e.g., the waves on the water, moving clouds, and escalators). These methods model the image regions as autoregressive moving average (ARMA) processes which provide a way to learn and predict the motion patterns in a scene. An ARMA process is a time series model that is made up of sums of autoregressive and moving-average components, where an autoregressive process can be described as a weighted sum of its previous values and a white noise error.

앞서 언급 한 접근법의 한 가지 제한 사항은 정적 배경이 필요하다는 것입니다. 이 제한은 Monnet et al. [2003], 그리고 Zhong and Sclaroff [2003]. 이 두 방법 모두 시간에 따라 변하는 배경 (예 : 물의 파도, 구름, 에스컬레이터 이동)을 처리 할 수 ​​있습니다. 이러한 방법은 이미지 영역을 ARMA (autoregressive moving average) 프로세스로 모델링하여 장면의 모션 패턴을 학습하고 예측하는 방법을 제공합니다. ARMA 프로세스는 자기 회귀 프로세스가 이전 값과 화이트 노이즈 오류의 가중 합으로 설명 될 수있는 자동 회귀 및 이동 평균 구성 요소의 합계로 구성된 시계열 모델입니다.

|  |
| --- |
|  |
| Fig. 5. Segmentation of the image shown in (a), using mean-shift segmentation (b) and normalized cuts (c).  평균 변이 분할 (b) 및 정규화 된 컷을 사용하여 (a)에 표시된 이미지의 세분화 |

In summary, most state-of-the-art tracking methods for fixed cameras, for example, Haritaoglu et al. [2000] and Collins et al. [2001] use background subtraction methods to detect regions of interest. This is because recent subtraction methods have the capabilities of modeling the changing illumination, noise, and the periodic motion of the background regions and, therefore, can accurately detect objects in a variety of circumstances. Moreover, these methods are computationally efficient. In practice, background subtraction provides incomplete object regions in many instances, that is, the objects may be spilled into several regions, or there may be holes inside the object since there are no guarantees that the object features will be different from the background features. The most important limitation of background subtraction is the requirement of stationary cameras. Camera motion usually distorts the background models. These methods can be applied to video acquired by mobile cameras by regenerating background models for small temporal windows, for instance, three frames, from scratch [Kanade et al. 1998] or by compensating sensor motion, for instance, creating background mosaics [Rowe and Blake 1996; Irani and Anandan 1998]. However, both of these solutions require assumptions of planar scenes and small motion in successive frames.

요약하면 고정 카메라에 대한 최첨단 추적 방법, 예를 들어, Haritaoglu et al. [2000] 및 Collins et al. [2001] 배경 차감 방법을 사용하여 관심 영역을 탐지합니다. 이는 최근의 뺄셈 방법이 변화하는 조명, 노이즈 및 배경 영역의주기적인 모션을 모델링 할 수있는 능력을 가지며, 따라서 다양한 상황에서 물체를 정확하게 검출 할 수 있기 때문입니다. 또한, 이러한 방법은 계산 상 효율적입니다. 실제로, 배경 차감은 많은 경우에 불완전한 객체 영역을 제공합니다. 즉, 객체가 여러 영역으로 흘러 들어가거나 객체 내부에 구멍이있을 수 있습니다. 객체 피처가 배경 피쳐와 다를 수 있다는 보장이 없기 때문입니다. 배경 빼기의 가장 중요한 한계는 정지 카메라의 요구 사항입니다. 카메라 모션은 대개 배경 모델을 왜곡합니다. 이 방법은 처음부터 세 개의 프레임과 같이 작은 시간 창에 대한 배경 모델을 재생성하여 모바일 카메라에서 얻은 비디오에 적용 할 수 있습니다 [Kanade et al. 1998] 또는 센서 모션 보상, 예를 들어 배경 모자이크 생성 [Rowe and Blake 1996; Irani and Anandan 1998]. 그러나이 두 가지 솔루션은 연속적인 프레임에서 평면 장면과 작은 모션의 가정을 필요로합니다.

**4.3. Segmentation**

The aim of image segmentation algorithms is to partition the image into perceptually similar regions. Every segmentation algorithm addresses two problems, the criteria for a good partition and the method for achieving efficient partitioning [Shi and Malik 2000]. In this section, we will discuss recent segmentation techniques that are relevant to object tracking.

이미지 분할 알고리즘의 목적은 이미지를 지각 적으로 유사한 영역으로 분할하는 것입니다. 모든 세그멘테이션 알고리즘은 두 가지 문제, 즉 좋은 파티션의 기준과 효율적인 파티셔닝을 달성하는 방법을 다룬다 [Shi and Malik 2000]. 이 섹션에서는 객체 추적과 관련된 최근 세분화 기법에 대해 설명합니다.

4.3.1. Mean-Shift Clustering. For the image segmentation problem, Comaniciu and Meer [2002] propose the mean-shift approach to find clusters in the joint spatial+color space, [l, u, v, x, y], where [l, u, v] represents the color and [x, y] represents the spatial location. Given an image, the algorithm is initialized with a large number of hypothesized cluster centers randomly chosen from the data. Then, each cluster center is moved to the mean of the data lying inside the multidimensional ellipsoid centered on the cluster center. The vector defined by the old and the new cluster centers is called the mean-shift vector. The mean-shift vector is computed iteratively until the cluster centers do not change their positions. Note that during the mean-shift iterations, some clusters may get merged. In Figure 5(b), we show the segmentation using the mean-shift approach generated using the source code available at MeanShiftSegmentSrc. Mean-shift clustering is scalable to various other applications such as edge detection, image regularization [Comaniciu and Meer 2002], and tracking [Comaniciu et al. 2003].

Mean-Shift 클러스터링. 이미지 분할 문제에 대해, Comaniciu와 Meer [2002]는 [l, u, v]가 [l, u, v, x, y] 인 공동 공간 + 색상 공간에서 클러스터를 찾기위한 평균 이동 접근법을 제안한다. 색과 [x, y]는 공간적인 위치를 나타낸다. 이미지가 주어지면 알고리즘은 데이터에서 무작위로 선택된 많은 수의 가정 된 클러스터 센터로 초기화됩니다. 그런 다음 각 클러스터 센터는 클러스터 센터를 중심으로 다차원 타원체 내부에있는 데이터의 평균으로 이동합니다. 이전 클러스터 센터와 새 클러스터 센터에서 정의한 벡터를 평균 이동 벡터라고합니다. 평균 이동 벡터는 클러스터 센터가 위치를 변경하지 않을 때까지 반복적으로 계산됩니다. 평균 이동 반복 중에 일부 클러스터가 병합 될 수 있습니다. 그림 5 (b)에서 MeanShiftSegmentSrc에서 사용 가능한 소스 코드를 사용하여 생성 된 평균 이동 접근법을 사용하여 세분화를 보여줍니다. Mean-shift 클러스터링은 엣지 감지, 이미지 정규화 및 추적과 같은 다양한 다른 어플리케이션으로 확장 가능합니다. [Comaniciu et al. 2003].

Mean-shift based segmentation requires fine tuning of various parameters to obtain better segmentation, for instance, selection of the color and spatial kernel bandwidths, and the threshold for the minimum size of the region considerably effect the resulting segmentation.

Mean-shift 기반 세분화는 색상 및 공간 커널 대역폭의 선택과 같이 더 나은 세분화를 얻기 위해 다양한 매개 변수를 미세 조정해야하며, 영역의 최소 크기에 대한 임계 값은 결과 세그먼트 화에 상당히 영향을 미칩니다.

4.3.2. Image Segmentation Using Graph-Cuts. Image segmentation can also be formulated as a graph partitioning problem, where the vertices (pixels), V = {u, v, ...}, of a graph (image), G, are partitioned into N disjoint subgraphs (regions), Ai, N i = 1 Ai = V, Ai ∩ Aj = ∅, i = j, by pruning the weighted edges of the graph. The total weight of the pruned edges between two subgraphs is called a cut. The weight is typically computed by color, brightness, or texture similarity between the nodes. Wu and Leahy [1993] use the minimum cut criterion, where the goal is to find the partitions that minimize a cut. In their approach, the weights are defined based on the color similarity. One limitation of minimum cut is its bias toward oversegmenting the image. This effect is due to the increase in cost of a cut with the number of edges going across the two partitioned segments.

그래프 컷을 이용한 이미지 분할. 이미지 분할은 또한 그래프 분할 문제로 공식화 될 수 있는데, 여기서 그래프 (이미지) G의 정점 (픽셀) V = {u, v, ...}는 N 개의 분리 된 부분 그래프 (영역)로 분할되며, Ai, N i = 1 Ai = V, Ai ∩ Aj = ∅, i = j, 그래프의 가중치를 제거. 두 개의 하위 그래프 사이의 정리 된 가장자리의 총 무게를 절단이라고합니다. 가중치는 일반적으로 노드 간의 색상, 밝기 또는 텍스처 유사성에 의해 계산됩니다. Wu와 Leahy [1993]는 최소 절단 기준을 사용하는데, 목표는 절단을 최소화하는 분할을 찾는 것이다. 그들의 접근 방식에서, 가중치는 색상 유사성을 기반으로 정의됩니다. 최소 절단의 한 가지 한계는 이미지를 초과 세그먼트 화하는 편향입니다. 이 효과는 두 개의 분할 된 세그먼트를 가로 질러가는 엣지의 수와 함께 컷의 비용이 증가하기 때문입니다.

Shi and Malik [2000] propose the normalized cut to overcome the oversegmentation problem. In their approach, the cut not only depends on the sum of edge weights in the cut, but also on the ratio of the total connection weights of nodes in each partition to all nodes of the graph. For image-based segmentation, the weights between the nodes are defined by the product of the color similarity and the spatial proximity. Once the weights between each pair of nodes are computed, a weight matrix W and a diagonal matrix D, where Di,i = N j = 1 Wi, j , are constructed. The segmentation is performed first by computing the eigenvectors and the eigenvalues of the generalized eigensystem (D − W)y = λDy, then the second-smallest eigenvector is used to divide the image into two segments. For each new segment, this process is recursively performed until a threshold is reached. In Figure 5(c), we show the segmentation results obtained by the normalized cuts approach.

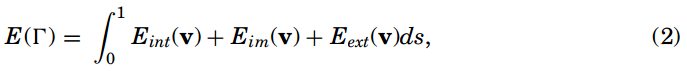
In normalized cuts-based segmentation, the solution to the generalized eigensystem for large images can be expensive in terms of processing and memory requirements. However, this method requires fewer manually selected parameters, compared to meanshift segmentation. Normalized cuts have also been used in the context of tracking object contours [Xu and Ahuja 2002].

Shi와 Malik [2000]은 oversegmentation 문제를 극복하기 위해 정규화 된 cut을 제안한다. 그들의 접근법에서, 컷은 컷의 엣지 가중치의 합뿐만 아니라 각 파티션의 노드의 총 연결 가중치와 그래프의 모든 노드의 비율에도 의존합니다. 이미지 기반 분할의 경우 노드 간 가중치는 색상 유사성과 공간 근접성의 곱으로 정의됩니다. 각 노드 쌍 사이의 가중치가 계산되면, 가중치 행렬 및 대각 행렬 D가 생성되고, Di, i = Nj = 1 Wi, j가 구성된다. 세분화는 먼저 일반화 된 고유 시스템 (D - W) y = λDy의 고유 벡터와 고유 값을 계산하여 수행 한 다음 두 번째로 작은 고유 벡터를 사용하여 이미지를 두 세그먼트로 나눕니다. 새로운 세그먼트마다이 프로세스는 임계 값에 도달 할 때까지 재귀 적으로 수행됩니다. 그림 5 (c)에서, 우리는 정규화 된 컷 접근법에 의해 얻어진 분할 결과를 보여준다.

정규화 된 컷 기반 세분화에서 대형 이미지 용 일반화 된 고유 시스템에 대한 솔루션은 처리 및 메모리 요구 사항면에서 비용이 많이들 수 있습니다. 그러나이 방법은 임시 세그먼트 화 방법과 비교하여 수동으로 선택한 매개 변수를 덜 필요로합니다. 정규화 된 컷은 오브젝트 윤곽 추적과 관련하여 사용되었습니다 [Xu and Ahuja 2002].

4.3.3. Active Contours. In an active contour framework, object segmentation is achieved by evolving a closed contour to the object’s boundary, such that the contour tightly encloses the object region. Evolution of the contour is governed by an energy functional which defines the fitness of the contour to the hypothesized object region. Energy functional for contour evolution has the following common form:

활성 윤곽선. 활성 윤곽 프레임 워크에서 객체 분할은 닫힌 윤곽을 객체의 경계까지 전개하여 윤곽이 객체 영역을 단단히 둘러싸도록하여 수행됩니다. 윤곽의 진화는 가설 영역에 대한 윤곽의 적합성을 정의하는 에너지 기능에 의해 결정됩니다. 등고선 진화를위한 에너지 기능은 다음과 같은 일반적인 형태를 가지고 있습니다 :



where s is the arc-length of the contour , Eint includes regularization constraints, Eim includes appearance-based energy, and Eext specifies additional constraints. Eint usually includes a curvature term, first-order (∇v) or second-order (∇2v) continuity terms to find the shortest contour. Image-based energy, Eim, can be computed locally or globally. Local information is usually in the form of an image gradient and is evaluated around the contour [Kass et al. 1988; Caselles et al. 1995]. In contrast, global features are computed inside and outside of the object region. Global features include color [Zhu and Yuille 1996; Yilmaz et al. 2004; Ronfard 1994] and texture [Paragios and Deriche 2002].

여기서 s는 윤곽선의 호 길이이고 Eint는 정규화 제약 조건을 포함하고 Eim은 모양 기반 에너지를 포함하며 Eext는 추가 제약 조건을 지정합니다. Eint는 보통 최단 윤곽을 찾기 위해 곡률 항, 1 차 (∇v) 또는 2 차 (∇2v) 연속 항을 포함합니다. 이미지 기반 에너지 Eim은 국지적으로 또는 전 세계적으로 계산 될 수 있습니다. 지역 정보는 일반적으로 이미지 그래디언트의 형태로되어 있으며 컨투어를 중심으로 평가됩니다 [Kass et al. 1988; Caselles et al. 1995]. 대조적으로 전역 기능은 객체 영역의 내부와 외부에서 계산됩니다. 글로벌 기능에는 색상 [Zhu and Yuille 1996; Yilmaz et al. 2004; Ronfard 1994] 그리고 질감 [Paragios and Deriche 2002]을 참고한다.

Different researchers have used different energy terms in Equation (2). In 1995, Caselles et al. exclude the Eext and use only the image gradient as the image energy Eim = g(|∇ I |), where g is the sigmoid function. Compared to the gradient, a function of the gradient defines the object contour as a geodesic curve in the Riemannian space [Caselles et al. 1995]. However, image gradients provide very local information and are sensitive to local minima. In order to overcome this problem, researchers introduced region-based image energy terms. In 1996, Zhu and Yuille proposed using region information instead of the image gradient. However, the use of regional terms in the energy functional does not result in good localization of the object contour. Recently, methods that combine both region-based and gradient-based image energy have become popular. Paragios and Deriche [2002] propose using a convex combination of the gradient and region-based energies, Eimage = λEboundary + (1 − λ)Eregion. In particular, the authors model the appearance in Eregion by mixture of Gaussians. Contour evolution is first performed globally, then locally by varying the α from 0 to 1 at each iteration.

다른 연구자들은 식 (2)에서 다른 에너지 항을 사용했다. 1995 년 Caselles et al. Eext를 제외하고 이미지 에너지 Eim = g (| ∇I |)으로 이미지 그래디언트만을 사용하십시오. 여기서 g는 시그 모이 드 함수입니다. 그라디언트와 비교할 때, 그라디언트의 함수는 리머 니안 공간에서 측량 곡선으로 객체 윤곽을 정의합니다 [Caselles et al. 1995]. 그러나 이미지 그라디언트는 매우 로컬 정보를 제공하며 로컬 미니 마에 민감합니다. 이 문제를 극복하기 위해 연구자들은 영역 기반 이미지 에너지 용어를 도입했습니다. 1996 년 Zhu와 Yuille는 이미지 그라디언트 대신 지역 정보를 사용하도록 제안했습니다. 그러나 에너지 함수에서 지역 용어를 사용한다고해서 객체 윤곽선이 잘 위치 파악되지는 않습니다. 최근에는 영역 기반 및 그라디언트 기반 이미지 에너지를 결합하는 방법이 널리 보급되고 있습니다. Paragios와 Deriche [2002]는 기울기와 영역 기반 에너지의 볼록한 조합 인 Eimage = λEboundary + (1 - λ) Eregion을 사용하여 제안했다. 특히, 저자는 가우시안의 혼합에 의해 Eregion의 외관을 모델링합니다. 윤곽선 진화는 먼저 전역 적으로 수행 된 다음 각 반복마다 α를 0에서 1로 변경하여 로컬로 수행됩니다.

performed globally, then locally by varying the α from 0 to 1 at each iteration. An important issue in contour-based methods is the contour initialization. In image gradient-based approaches, a contour is typically placed outside the object region and shrunk until the object boundary is encountered [Kass et al. 1988; Caselles et al. 1995]. This constraint is relaxed in region-based methods such that the contour can be initialized either inside or outside the object so that the contour can either expand or shrink, respectively, to fit the object boundary. However, these approaches require prior object or background knowledge [Paragios and Deriche 2002]. Using multiple frames or a reference frame, initialization can be performed without building region priors. For instance, in Paragios and Deriche [2000], the authors use background subtraction to initialize the contour.

전역 적으로 수행 된 다음 각 반복마다 α를 0에서 1로 변경하여 로컬로 수행합니다. 윤곽 기반 방법에서 중요한 문제는 윤곽 초기화입니다. 이미지 그래디언트 기반 접근법에서 윤곽선은 일반적으로 객체 영역 외부에 배치되고 객체 경계에 도달 할 때까지 축소됩니다 [Kass et al. 1988; Caselles et al. 1995]. 이 제약 조건은 영역 기반 방법에서 완화되어 객체의 내부 또는 외부에서 등고선을 초기화하여 객체 경계에 맞게 윤곽선을 각각 확장하거나 축소 할 수 있습니다. 그러나 이러한 접근법은 이전의 대상이나 배경 지식을 필요로한다 [Paragios and Deriche 2002]. 여러 프레임 또는 참조 프레임을 사용하여 영역 사전 작성없이 초기화를 수행 할 수 있습니다. 예를 들어, Paragios와 Deriche [2000]에서 저자는 배경 감산을 사용하여 등고선을 초기화합니다.

Besides the selection of the energy functional and the initialization, another important issue is selecting the right contour representation. Object contour, , can be represented either explicitly (control points, v) or implicitly (level sets, φ). In the explicit representation, the relation between the control points are defined by spline equations. In the level sets representation, the contour is represented on a spatial grid which encodes the signed distances of the grids from the contour with opposite signs for the object and the background regions. The contour is implicitly defined as the zero crossings in the level set grid. The evolution of the contour is governed by changing the grid values according to the energy computed using Equation (2), evaluated at each grid position. The changes in the grid values result in new zero crossings, hence, a new contour positions (more details are given in Section 5.3). The source code for generic level sets, which can be used for various applications by specifying the contour evolution speed, for instance, segmentation, tracking, heat flow etc., is available at LevelSetSrc. The most important advantage of implicit representation over the explicit representation is its flexibility in allowing topology changes (split and merge).

에너지 기능과 초기화의 선택 외에도 또 다른 중요한 문제는 올바른 등고선 표현을 선택하는 것입니다. 객체 윤곽선은 명시 적으로 (제어점, v) 또는 암시 적으로 (레벨 집합, φ) 나타낼 수 있습니다. 명시 적 표현에서 제어점 간의 관계는 스플라인 방정식으로 정의됩니다. 레벨 세트 표현에서, 컨투어는 객체 및 배경 영역에 대해 반대 기호가있는 컨투어에서 그리드의 부호가있는 거리를 인코딩하는 공간 격자에 표시됩니다. 윤곽선은 레벨 집합 격자의 제로 크로싱으로 암시 적으로 정의됩니다. 윤곽선의 진화는 식 (2)를 사용하여 계산 된 에너지에 따라 그리드 값을 변경하여 각 그리드 위치에서 평가합니다. 그리드 값이 변경되면 새로운 제로 크로싱이 생기므로 새 윤곽 위치가 생깁니다 (자세한 내용은 5.3 절 참조). LevelSetSrc에서 세분화, 추적, 열 흐름 등과 같은 형상 발전 속도를 지정하여 다양한 응용 프로그램에 사용할 수있는 일반 레벨 세트의 소스 코드를 사용할 수 있습니다. 명시 적 표현에 비해 암시 적 표현의 가장 중요한 이점은 토폴로지 변경 (분할 및 병합) 허용의 유연성입니다.

**4.4. Supervised Learning**

Object detection can be performed by learning different object views automatically from a set of examples by means of a supervised learning mechanism. Learning of different object views waives the requirement of storing a complete set of templates. Given a set of learning examples, supervised learning methods generate a function that maps inputs to desired outputs. A standard formulation of supervised learning is the classification problem where the learner approximates the behavior of a function by generating an output in the form of either a continuous value, which is called regression, or a class label, which is called classification. In context of object detection, the learning examples are composed of pairs of object features and an associated object class where both of these quantities are manually defined.

객체 감지는 감독 학습 메커니즘을 사용하여 일련의 예제에서 자동으로 다른 객체보기를 학습하여 수행 할 수 있습니다. 다른 객체 뷰를 학습하면 완전한 템플릿 집합을 저장해야 할 필요가 없습니다. 일련의 학습 사례가 주어지면 감독 학습 방법은 입력을 원하는 출력에 매핑하는 함수를 생성합니다. 감독 학습의 표준 공식은 학습자가 회귀라고하는 연속 값 또는 분류라고하는 클래스 레이블의 형태로 출력을 생성하여 함수의 동작을 근사하는 분류 문제입니다. 물체 감지의 맥락에서 학습 예제는 물체 피쳐 쌍과 물체 클래스 쌍으로 구성됩니다.이 두 물 리량은 수동으로 정의됩니다.

Selection of features plays an important role in the performance of the classification, hence, it is important to use a set of features that discriminate one class from the other. In addition to the features discussed in Section 3, it is also possible to use other features such as object area, object orientation, and object appearance in the form of a density function, for example, histogram. Once the features are selected, different appearances of an object can be learned by choosing a supervised learning approach. These learning approaches include, but are not limited to, neural networks [Rowley et al. 1998], adaptive boosting [Viola et al. 2003], decision trees [Grewe and Kak 1995], and support vector machines [Papageorgiou et al. 1998]. These learning methods compute a hypersurface that separates one object class from the other in a high dimensional space.

피쳐의 선택은 분류의 수행에 중요한 역할을하므로, 한 클래스를 다른 클래스와 구별하는 일련의 피쳐를 사용하는 것이 중요합니다. 3 장에서 논의 된 기능 이외에, 히스토그램과 같은 밀도 함수의 형태로 객체 영역, 객체 방향 및 객체 모양과 같은 다른 기능을 사용할 수도 있습니다. 피쳐가 선택되면 감독 학습 방식을 선택하여 객체의 다양한 모습을 학습 할 수 있습니다. 이러한 학습 방법으로는 신경 네트워크 [Rowley et al. 1998], adaptive boosting [Viola et al. 2003], 의사 결정 나무 [Grewe and Kak 1995], 지원 벡터 머신 [Papageorgiou et al. 1998]. 이러한 학습 방법은 고 차원 공간에서 하나의 객체 클래스와 다른 객체 클래스를 분리하는 초 표면을 계산합니다.

Supervised learning methods usually require a large collection of samples from each object class. Additionally, this collection must be manually labeled. A possible approach to reducing the amount of manually labeled data is to accompany cotraining with supervised learning [Blum and Mitchell 1998]. The main idea behind cotraining is to train two classifiers using a small set of labeled data where the features used for each classifier are independent. After training is achieved, each classifier is used to assign unlabeled data to the training set of the other classifier. It was shown that, starting from a small set of labeled data with two sets of statistically independent features, cotraining can provide a very accurate classification rule [Blum and Mitchell 1998]. Cotraining has been successfully used to reduce the amount of manual interaction required for training in the context of adaboost [Levin et al. 2003] and support vector machines [Kockelkorn et al. 2003]. Following we will discuss the adaptive boosting and the support vector machines due to their applicability to object tracking.

감독 학습 방법은 대개 각 객체 클래스의 샘플 모음을 필요로합니다. 또한이 컬렉션을 수동으로 레이블해야합니다. 수작업으로 분류 된 데이터의 양을 줄이는 접근법은 감독 학습 (supervised learning)을 통한 공동 작업 (cotraining)을 수반하는 것이다 (Blum and Mitchell 1998). cotraining의 주된 아이디어는 각 분류 자에 사용 된 특징이 독립적 인 작은 레이블 데이터 세트를 사용하여 두 개의 분류자를 훈련시키는 것입니다. 트레이닝이 이루어진 후에는 각 분류기를 사용하여 분류되지 않은 데이터를 다른 분류기의 트레이닝 세트에 할당합니다. 통계적으로 독립적 인 두 세트의 표지 된 데이터 세트에서 시작하여, cotraining은 매우 정확한 분류 규칙을 제공 할 수 있음이 밝혀졌습니다 [Blum and Mitchell 1998]. Cotraining은 adaboost의 맥락에서 훈련에 필요한 수동 상호 작용의 양을 줄이는데 성공적으로 사용되었습니다 [Levin et al. 2003] 및 지원 벡터 머신 [Kockelkorn et al. 2003]. 다음은 객체 추적에 대한 적용 가능성으로 인해 적응 형 증폭 및 지원 벡터 머신에 대해 설명합니다.

4.4.1. Adaptive Boosting. Boosting is an iterative method of finding a very accurate classifier by combining many base classifiers, each of which may only be moderately accurate [Freund and Schapire 1995]. In the training phase of the Adaboost algorithm, the first step is to construct an initial distribution of weights over the training set. The boosting mechanism then selects a base classifier that gives the least error, where the error is proportional to the weights of the misclassified data. Next, the weights associated with the data misclassified by the selected base classifier are increased. Thus the algorithm encourages the selection of another classifier that performs better on the misclassified data in the next iteration. For interested readers tutorials on boosting are available at <http://www.boosting.org>.

In the context of object detection, weak classifiers can be simple operators such as a set of thresholds, applied to the object features extracted from the image. In 2003, Viola et al. used the Adaboost framework to detect pedestrians. In their approach, perceptrons were chosen as the weak classifiers which are trained on image features extracted by a combination of spatial and temporal operators. The operators for feature extraction are in the form of simple rectangular filters, and are shown in Figure 6. The operators in the temporal domain are in the form of frame differencing which encode some form of motion information. Frame differencing, when used as an operator in the temporal domain, reduces the number of false detections by enforcing object detection in the regions where the motion occurs.

적응 부스트. Boosting은 많은 기본 분류기를 결합하여 매우 정확한 분류기를 찾는 반복적 인 방법으로, 각 분류기는 비교적 정확할 수 있습니다 [Freund and Schapire 1995]. Adaboost 알고리즘의 트레이닝 단계에서 첫 번째 단계는 트레이닝 집합에 대한 가중치의 초기 분포를 만드는 것입니다. 부스팅 메카니즘은 에러가 오 분류 된 데이터의 가중치에 비례하는 최소 에러를 제공하는 기본 분류자를 선택한다. 다음으로, 선택된 기본 분류기에 의해 잘못 분류 된 데이터와 관련된 가중치가 증가합니다. 따라서 알고리즘은 다음 반복에서 잘못 분류 된 데이터에서 더 잘 수행되는 다른 분류자를 선택하도록 장려합니다. 흥미있는 독자를 위해 강화에 자습서는 http://www.boosting.org에 유효하다.

물체 감지와 관련하여 약 분류기는 이미지에서 추출한 객체 피처에 적용된 임계 값 세트와 같은 간단한 연산자 일 수 있습니다. 2003 년 Viola et al. 보행자를 탐지하기 위해 Adaboost 프레임 워크를 사용했습니다. 그들의 접근 방식에서, 퍼셉트론은 공간적 및 시간적 연산자의 조합에 의해 추출 된 이미지 특징에 대해 훈련 된 약한 분류기로 선택되었다. 특징 추출을위한 연산자는 간단한 직사각형 필터의 형태이며, 그림 6에 나와 있습니다. 시간 영역의 연산자는 몇 가지 형태의 동작 정보를 인코딩하는 프레임 차별화 형식입니다. 프레임 차이는 시간 영역에서 연산자로 사용될 때 모션이 발생하는 영역에서 객체 감지를 시행하여 오 탐지 횟수를 줄입니다.

4.4.2. Support Vector Machines. As a classifier, Support Vector Machines (SVM) are used to cluster data into two classes by finding the maximum marginal hyperplane that separates one class from the other [Boser et al. 1992]. The margin of the hyperplane, which is maximized, is defined by the distance between the hyperplane and the closest data points. The data points that lie on the boundary of the margin of the hyperplane are called the support vectors. In the context of object detection, these classes correspond to the object class (positive samples) and the nonobject class (negative samples). From manually generated training examples labeled as object and nonobject, computation of the hyperplane from among an infinite number of possible hyperplanes is carried out by means of quadratic programming.

지원 벡터 머신. 분류 자로서 SVM (Support Vector Machines)은 하나의 클래스를 다른 클래스와 구분하는 최대 한계 초평면을 찾음으로써 데이터를 두 클래스로 클러스터링하는 데 사용됩니다 [Boser et al. 1992]. 최대화되는 초평면의 여백은 초평면과 가장 근접한 데이터 점 사이의 거리로 정의됩니다. 초평면의 여백 경계에있는 데이터 점을지지 벡터라고합니다. 객체 탐지와 관련하여 이러한 클래스는 객체 클래스 (양수 샘플)와 비 객체 클래스 (음수 샘플)에 해당합니다. 객체 및 객체가 아닌 것으로 수동으로 생성 된 트레이닝 예제에서 무한 수의 가능한 초평면 중 초평면 계산이 2 차 프로그래밍을 통해 수행됩니다.

|  |
| --- |
|  |
| Fig. 6. A set of rectangular filters used by Viola et al. [2003] to extract features used in the Adaboost framework. Each filter is composed of three regions: white, light gray, and dark gray, with associated weights 0, −1, and 1 respectively. In order to compute the feature in a window, these filters are convolved with the image.  Viola 등이 사용하는 직사각형 필터 세트. [2003] Adaboost 프레임 워크에서 사용되는 기능을 추출합니다. 각 필터는 흰색, 밝은 회색 및 어두운 회색의 세 가지 영역으로 구성되며 관련 가중치는 각각 0, -1 및 1입니다. 창에서 피쳐를 계산하기 위해 이러한 필터가 이미지와 컨볼루션 됩니다. |

Despite being a linear classifier, SVM can also be used as a nonlinear classifier by applying the kernel trick to the input feature vector extracted from the input. Application of the kernel trick to a set of data that is not linearly separable, transforms the data to a higher dimensional space which is likely to be separable. The kernels used for kernel trick are polynomial kernels or radial basis functions, for example, Gaussian kernel and two-layer perceptron, for instance, a sigmoid function. However, the selection of the right kernel for the problem at hand is not easy. Once a kernel is chosen, one has to test the classification performance for a set of parameters which may not work as well when new observations are introduced to the sample set.  
SVM은 선형 분류 자 ​​임에도 불구하고 입력에서 추출한 입력 특징 벡터에 커널 트릭을 적용하여 비선형 분류 자로 사용할 수도 있습니다. 선형 적으로 분리 할 수없는 데이터 세트에 커널 트릭을 적용하면 데이터를 분리 가능한 고차원 공간으로 변환합니다. 커널 트릭에 사용되는 커널은 다항식 커널 또는 방사형 기저 함수 (예 : 가우시안 커널 및 시그마 모달 함수와 같은 2 층 퍼셉트론)입니다. 그러나 문제의 올바른 커널 선택은 쉽지 않습니다. 일단 커널이 선택되면 새로운 관찰이 샘플 세트에 도입 될 때 잘 작동하지 않을 수있는 매개 변수 집합에 대한 분류 성능을 테스트해야합니다.

In the context of object detection, Papageorgiou et al. [1998] use SVM for detecting pedestrians and faces in images. The features used to discriminate between the classes are extracted by applying Haar wavelets to the sets of positive and negative training examples. In order to reduce the search space, temporal information is utilized by computing the optical flow field in the image. Particularly, the discontinuities in the optical flow field are used to initiate the search for possible objects resulting in a decreased number of false positives.

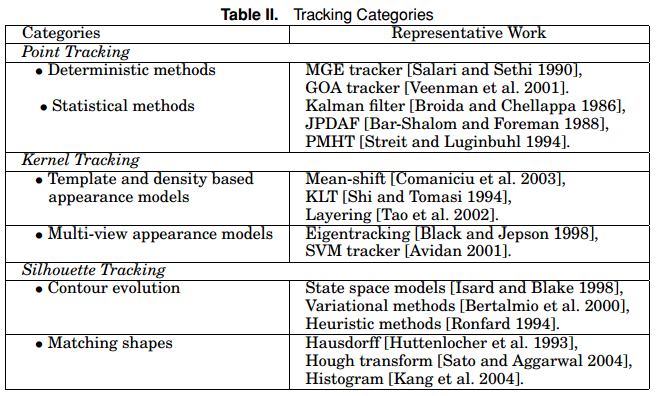
물체 감지와 관련하여, Papageorgiou et al. [1998] 이미지에서 보행자와 얼굴을 탐지하기 위해 SVM을 사용한다. Haar 웨이블릿을 양수 및 음수 트레이닝 예제 세트에 적용하여 클래스를 구분하는 데 사용되는 기능을 추출합니다. 검색 공간을 줄이기 위해 시간 정보는 이미지의 옵티컬 플로우 필드를 계산하여 활용됩니다. 특히, 옵티컬 플로우 필드의 불연속성은 가능한 객체에 대한 검색을 시작하는 데 사용되어 가양 성 (false positive)의 수가 감소합니다.

**5. OBJECT TRACKING**

The aim of an object tracker is to generate the trajectory of an object over time by locating its position in every frame of the video. Object tracker may also provide the complete region in the image that is occupied by the object at every time instant. The tasks of detecting the object and establishing correspondence between the object instances across frames can either be performed separately or jointly. In the first case, possible object regions in every frame are obtained by means of an object detection algorithm, and then the tracker corresponds objects across frames. In the latter case, the object region and correspondence is jointly estimated by iteratively updating object location and region information obtained from previous frames. In either tracking approach, the objects are represented using the shape and/or appearance models described in Section 2. The model selected to represent object shape limits the type of motion or deformation it can undergo. For example, if an object is represented as a point, then only a translational model can be used. In the case where a geometric shape representation like an ellipse is used for the object, parametric motion models like affine or projective transformations are appropriate. These representations can approximate the motion of rigid objects in the scene. For a nonrigid object, silhouette or contour is the most descriptive representation and both parametric and nonparametric models can be used to specify their motion.

물체 추적기의 목적은 비디오의 모든 프레임에서 물체의 위치를 ​​파악함으로써 시간의 흐름에 따라 물체의 궤도를 생성하는 것입니다. 물체 추적기는 매 시간마다 물체가 차지하는 이미지의 전체 영역을 제공 할 수도 있습니다. 오브젝트를 검출하고 프레임을 가로 질러 오브젝트 인스턴스들 간의 대응 관계를 설정하는 태스크는 개별적으로 또는 공동으로 수행 될 수 있습니다. 첫 번째 경우, 모든 프레임의 가능한 오브젝트 영역은 오브젝트 검출 알고리즘을 통해 얻은 다음, 트래커는 여러 프레임의 오브젝트에 해당합니다. 후자의 경우, 객체 영역 및 대응은 이전 프레임으로부터 획득 된 객체 위치 및 영역 정보를 반복적으로 업데이트함으로써 공동으로 추정된다. 두 추적 방법 중 하나에서 객체는 2 장에서 설명한 모양 및 / 또는 모양 모델을 사용하여 표현됩니다. 객체 모양을 나타 내기 위해 선택된 모형은 그것이받을 수있는 동작 또는 변형의 유형을 제한합니다. 예를 들어, 객체가 점으로 표현되는 경우 변환 모델 만 사용할 수 있습니다. 타원과 같은 기하학적 도형 표현이 객체에 사용되는 경우, 아핀 또는 투영 변환과 같은 파라 메트릭 모션 모델이 적절합니다. 이러한 표현은 장면에서 강체의 움직임을 근사시킬 수 있습니다. 가장 지독하지 않은 오브젝트의 경우 실루엣 또는 윤곽이 가장 서술적인 표현이며 파라 메트릭 및 비모수 모델 모두 모션을 지정하는 데 사용될 수 있습니다.

|  |
| --- |
|  |
| Fig. 7. Taxonomy of tracking methods. |



|  |
| --- |
|  |
| Fig. 8. (a) Different tracking approaches. Multipoint correspondence, (b) parametric transformation of a rectangular patch, (c, d) Two examples of contour evolution. |

In view of the aforementioned discussion, we provide a taxonomy of tracking methods in Figure 7. Representative work for each category is tabulated in Table II. We now briefly introduce the main tracking categories, followed by a detailed section on each category.

앞서 논의한 관점에서 볼 때, 우리는 그림 7에서 추적 방법의 분류법을 제공합니다. 각 카테고리의 대표적인 작업을 표 2에 도표로 나타냅니다. 이제 주요 추적 카테고리를 간략하게 소개 한 다음 각 카테고리에 대한 자세한 섹션을 소개합니다.

—Point Tracking. Objects detected in consecutive frames are represented by points, and the association of the points is based on the previous object state which can include object position and motion. This approach requires an external mechanism to detect the objects in every frame. An example of object correspondence is shown in Figure 8(a).

포인트 추적. 연속적인 프레임들에서 검출 된 객체들은 점들에 의해 표현되고, 점들의 연관은 객체 위치 및 동작을 포함 할 수있는 이전의 객체 상태에 기초한다. 이 접근법은 매 프레임마다 객체를 탐지하는 외부 메커니즘을 필요로합니다. 객체 대응의 예가 그림 8 (a)에 나와 있습니다.

—Kernel Tracking. Kernel refers to the object shape and appearance. For example, the kernel can be a rectangular template or an elliptical shape with an associated histogram. Objects are tracked by computing the motion of the kernel in consecutive frames (Figure 8(b)). This motion is usually in the form of a parametric transformation such as translation, rotation, and affine.

커널 추적. 커널은 객체 모양과 모양을 나타냅니다. 예를 들어, 커널은 연관된 막대 그래프가있는 직사각형 템플릿이나 타원형이 될 수 있습니다. 객체는 연속적인 프레임에서 커널의 움직임을 계산하여 추적됩니다 (그림 8 (b)). 이 동작은 대개 평행 이동, 회전 및 아핀과 같은 매개 변수 변환 형식입니다.

—Silhouette Tracking. Tracking is performed by estimating the object region in each frame. Silhouette tracking methods use the information encoded inside the object region. This information can be in the form of appearance density and shape models which are usually in the form of edge maps. Given the object models, silhouettes are tracked by either shape matching or contour evolution (see Figure 8(c), (d)). Both of these methods can essentially be considered as object segmentation applied in the temporal domain using the priors generated from the previous frames.

실루엣 추적. 트래킹은 각 프레임에서 대상 영역을 추정하여 수행됩니다. 실루엣 추적 방법은 객체 영역 내부에 인코딩 된 정보를 사용합니다. 이 정보는 일반적으로 에지 맵의 형태 인 모양 밀도 및 모양 모델의 형태 일 수 있습니다. 객체 모델이 주어지면, 실루엣은 형상 매칭 또는 등고선 진화에 의해 추적됩니다 (그림 8 (c), (d) 참조). 이 두 가지 방법은 본질적으로 이전 프레임에서 생성 된 사전을 사용하여 시간 영역에서 적용된 객체 세분화로 간주 될 수 있습니다.

**5.1. Point Tracking**

Tracking can be formulated as the correspondence of detected objects represented by points across frames. Point correspondence is a complicated problem-specially in the presence of occlusions, misdetections, entries, and exits of objects. Overall, point correspondence methods can be divided into two broad categories, namely, deterministic and statistical methods. The deterministic methods use qualitative motion heuristics [Veenman et al. 2001] to constrain the correspondence problem. On the other hand, probabilistic methods explicitly take the object measurement and take uncertainties into account to establish correspondence.

추적은 프레임을 가로 질러 포인트로 표시되는 탐지 된 객체의 일치로 공식화 될 수 있습니다. 점 대응은 특히 폐색, 오 탐지, 항목 및 객체 종료가있는 복잡한 문제입니다. 전반적으로, 포인트 대응 방법은 크게 결정 론적 방법과 통계적 방법의 두 가지 범주로 나눌 수 있습니다. 결정 론적 방법은 정 성적 운동 휴리스틱을 사용한다 [Veenman et al. 2001]. 다른 한편, 확률 론적 방법은 명시 적으로 물체 측정을 ​​취하고 통신을 확립하기 위해 불확실성을 고려합니다.

5.1.1. Deterministic Methods for Correspondence. Deterministic methods for point correspondence define a cost of associating each object in frame t − 1 to a single object in frame t using a set of motion constraints. Minimization of the correspondence cost is formulated as a combinatorial optimization problem. A solution, which consists of oneto-one correspondences (Figure 9(b)) among all possible associations (Figure 9(a)), can be obtained by optimal assignment methods, for example, Hungarian algorithm, [Kuhn 1955] or greedy search methods. The correspondence cost is usually defined by using a combination of the following constraints.

통신에 대한 결정 론적 방법. 점 대응을위한 결정 론적 방법은 모션 제약 조건 집합을 사용하여 프레임 t - 1의 각 객체를 프레임 t의 단일 객체와 연관시키는 비용을 정의합니다. 대응 비용의 최소화는 조합 최적화 문제로 공식화된다. 예를 들어, 헝가리 알고리즘, [Kuhn 1955] 또는 탐욕적인 검색과 같은 최적의 할당 방법을 통해 모든 가능한 연관들 (그림 9 (a)) 중 하나의 대응 (그림 9 (b))으로 구성된 솔루션을 얻을 수 있습니다 행동 양식. 통신 비용은 일반적으로 다음 제약 조건의 조합을 사용하여 정의됩니다.

—Proximity assumes the location of the object would not change notably from one frame to other (see Figure 10(a)).

근접성은 객체의 위치가 한 프레임에서 다른 프레임으로 크게 변하지 않는다고 가정합니다 (그림 10 (a) 참조).

—Maximum velocity defines an upper bound on the object velocity and limits the possible correspondences to the circular neighborhood around the object (see Figure 10(b)).

최대 속도는 물체 속도의 상한을 정의하고 가능한 대응을 물체 주변의 원형 이웃으로 제한합니다 (그림 10 (b) 참조).

—Small velocity change (smooth motion) assumes the direction and speed of the object does not change drastically (see Figure 10(c)).

작은 속도 변화 (부드러운 움직임)는 물체의 방향과 속도가 크게 변하지 않는다고 가정합니다 (그림 10 (c) 참조).

—Common motion constrains the velocity of objects in a small neighborhood to be similar (see Figure 10(d)). This constraint is suitable for objects represented by multiple points.

공통 운동은 작은 이웃에있는 물체의 속도를 비슷하게 제한합니다 (그림 10 (d) 참조). 이 제한은 여러 점으로 표시된 객체에 적합합니다.

—Rigidity assumes that objects in the 3D world are rigid, therefore, the distance between any two points on the actual object will remain unchanged (see Figure 10(e)).

강성은 3D 세계의 오브젝트가 고정되어 있다고 가정하므로 실제 오브젝트의 두 점 사이의 거리는 변경되지 않습니다 (그림 10 (e) 참조).

—Proximal uniformity is a combination of the proximity and the small, velocity change constraints.

근사 균일 성은 근접 도와 작은 속도 변화 제약의 조합입니다.

|  |
| --- |
|  |
| Fig. 9. Point correspondence. (a) All possible associations of a point (object) in frame t − 1 with points (objects) in frame t, (b) unique set of associations plotted with bold lines, (c) multiframe correspondences.  포인트 대응. (a) 프레임 t - 1의 점 (객체)과 프레임 t의 점 (객체)과의 모든 가능한 연관, (b) 굵은 선으로 그려진 고유 한 연관성 집합, (c) 다중 프레임 대응 성. |
|  |
| Fig. 10. Different motion constraints. (a) proximity, (b) maximum velocity (r denotes radius), (c) small velocity-change, (d) common motion, (e) rigidity constraints. denotes object position at frame t − 2, ◦ denotes object position at frame t − 1, and finally × denotes object position at frame t.  다른 모션 제약. (a) 근접, (b) 최대 속도 (r은 반경), (c) 작은 속도 변화, (d) 공통 운동, (e) 강성 제약. 는 프레임 t - 2에서의 물체 위치를 나타내며, ◦는 프레임 t - 1에서의 물체 위치를 나타내며, 마지막으로 x는 프레임 t에서의 물체 위치를 나타냅니다. |

We should, however, note that these constraints are not specific to the deterministic methods, and they can also be used in the context of point tracking using statistical methods.

그러나 이러한 제약 조건은 결정 론적 방법에만 국한되지 않으며 통계 방법을 사용하는 포인트 추적과 관련하여 사용될 수도 있습니다.

Here we present a sample of different methods proposed in the literature in this category. Sethi and Jain [1987] solve the correspondence by a greedy approach based on the proximity and rigidity constraints. Their algorithm considers two consecutive frames and is initialized by the nearest neighbor criterion. The correspondences are exchanged iteratively to minimize the cost. A modified version of the same algorithm which computes the correspondences in the backward direction (from the last frame to the first frame) in addition to the forward direction is also analyzed. This method cannot handle occlusions, entries, or exits. Salari and Sethi [1990] handle these problems, by first establishing correspondence for the detected points and then extending the tracking of the missing objects by adding a number of hypothetical points. Rangarajan and Shah [1991] propose a greedy approach, which is constrained by proximal uniformity. Initial correspondences are obtained by computing optical flow in the first two frames. The method does not address entry and exit of objects. If the number of detected points decrease, occlusion or misdetection is assumed. Occlusion is handled by establishing the correspondence for the detected objects in the current frame. For the remaining objects, position is predicted based on a constant velocity assumption. In the work by Intille et al. [1997], which uses a slightly modified version of Rangarajan and Shah [1991] for matching object centroids, the objects are detected by using background subtraction. The authors explicitly handle the change in the number of objects by examining specific regions in the image, for example, a door, to detect entries/exits before computing the correspondence.

여기에서는 이 카테고리의 문헌에서 제안 된 여러 가지 방법의 샘플을 제시합니다. Sethi와 Jain [1987]은 근접성과 강성 제약에 기초한 탐욕적인 접근에 의해 서신을 풀어 낸다. 그들의 알고리즘은 두 개의 연속 된 프레임을 고려하고 가장 가까운 이웃 기준에 의해 초기화됩니다. 통신은 비용을 최소화하기 위해 반복적으로 교환됩니다. 순방향뿐만 아니라 역방향 (최종 프레임에서 첫 번째 프레임까지)의 대응을 계산하는 동일한 알고리즘의 수정 된 버전도 분석됩니다. 이 메서드는 폐색, 항목 또는 종료를 처리 할 수 ​​없습니다. Salari와 Sethi [1990]는 먼저 검출 된 점들에 대한 일치 성을 확립 한 다음 여러 가상의 점들을 추가하여 누락 된 물체들의 추적을 확장함으로써 이러한 문제들을 다룬다. Rangarajan과 Shah [1991]는 근위 균일성에 의해 제약을받는 탐욕스러운 접근법을 제안한다. 초기 대응은 처음 두 프레임에서 옵티컬 플로우를 계산하여 얻어집니다. 이 메서드는 객체의 시작과 종료를 처리하지 않습니다. 검출 된 포인트의 수가 감소하면, 폐색 또는 오 검출이 가정된다. 폐색은 현재 프레임에서 탐지 된 객체에 대한 일치 성을 설정하여 처리됩니다. 나머지 물체에 대해, 위치는 일정한 속도 가정에 기초하여 예측된다. Intille et al. [1997]은 Rangarajan과 Shah [1991]의 약간 수정 된 버전을 사용하여 객체 중심을 매칭하며, 객체는 배경 빼기를 사용하여 탐지된다. 저자는 서신을 계산하기 전에 입구 / 출구를 감지하기 위해 이미지의 특정 영역 (예 : 문)을 검사하여 객체 수의 변경을 명시 적으로 처리합니다.

|  |
| --- |
|  |
| Fig. 11. Results of two point correspondence algorithms. (a) Tracking using the algorithm proposed by Veenman et al. [2001] in the rotating dish sequence color segmentation was used to detect black dots on a white dish (c 2001 IEEE). (b) Tracking birds using the algorithm proposed by Shafique and Shah [2003]; birds are detected using background subtraction (c 2003 IEEE).  2 점 일치 알고리즘의 결과. (a) Veenman et al. [2001] 회전 접시 순서에서 색 분할은 흰 접시에 검은 점을 감지하는 데 사용되었습니다 (c 2001 IEEE). (b) Shafique and Shah [2003]에 의해 제안 된 알고리즘을 사용하여 조류를 추적; 새들은 background subtraction (c 2003 IEEE)을 사용하여 탐지됩니다. |

Veenman et al. [2001] extend the work of Sethi and Jain [1987], and Rangarajan and Shah [1991] by introducing the common motion constraint for correspondence. The common motion constraint provides a strong constraint for coherent tracking of points that lie on the same object; however, it is not suitable for points lying on isolated objects moving in different directions. The algorithm is initialized by generating the initial tracks using a two-pass algorithm, and the cost function is minimized by Hungarian assignment algorithm in two consecutive frames. This approach can handle occlusion and misdetection errors, however, it is assumed that the number of objects are the same throughout the sequence, that is, no object entries or exits. See Figure 11(a) for tracking results.

Veenman et al. [2001] Sethi와 Jain [1987], Rangarajan and Shah [1991]의 연구를 확장하여 서신에 공통적 인 운동 제한을 도입 하였다. 일반적인 모션 제약 조건은 동일한 객체에있는 점들의 일관된 추적에 강력한 제약 조건을 제공합니다. 그러나 다른 방향으로 움직이는 고립 된 물체 위에 놓여있는 점에는 적합하지 않습니다. 알고리즘은 two-pass 알고리즘을 사용하여 초기 트랙을 생성함으로써 초기화되고 비용 함수는 두 개의 연속 프레임에서 헝가리 할당 알고리즘에 의해 최소화됩니다. 이 접근법은 오 클루 전 및 오 검출 오류를 처리 할 수 ​​있지만 오브젝트의 수가 시퀀스 전체에서 동일하다고 가정합니다 (오브젝트 입력 또는 종료가 없음). 결과 추적에 대해서는 그림 11 (a)를 참조하십시오.

Shafique and Shah [2003] propose a multiframe approach to preserve temporal coherency of the speed and position (Figure 9(c)). They formulate the correspondence as a graph theoretic problem. Multiple frame correspondence relates to finding the best unique path Pi = {x0, ... , xk} for each point (the superscript represents the frame number). For misdetected or occluded objects, the path will consist of missing positions in corresponding frames. The directed graph, which is generated using the points in k frames, is converted to a bipartite graph by splitting each node (object) into two (+ and −) nodes and representing directed edges as undirected edges from + to − nodes. The correspondence is then established by a greedy algorithm. They use a window of frames during point correspondence to handle occlusions whose durations are shorter than the temporal window used to perform matching. See Figure 11(b) for results on this algorithm for the tracking of birds.

Shafique와 Shah [2003]는 속도와 위치의 시간적 일관성을 보존하기 위해 다중 프레임 접근법을 제안한다 (그림 9 (c)). 그들은 서신을 그래프 이론 문제로 공식화합니다. 다중 프레임 대응은 각 포인트에 대해 최상의 고유 경로 Pi = {x0, ..., xk}를 찾는 것과 관련이 있습니다 (위 첨자는 프레임 번호를 나타냄). 잘못 검출되거나 폐색 된 물체의 경우 경로는 해당 프레임의 누락 된 위치로 구성됩니다. k 프레임의 점을 사용하여 생성 된 유향 그래프는 각 노드 (객체)를 두 개의 (+ 및 -) 노드로 분할하고 방향이 지정된 에지를 + 노드에서 + 노드로 나타내는 무차별 에지로 나타내는 방식으로 이분 그래프로 변환됩니다. 그 서신은 욕심 많은 알고리즘에 의해 설정됩니다. 이들은 지점 대응 중에 프레임 창을 사용하여 지속 시간이 일치를 수행하는 데 사용 된 시간 창보다 짧게하는 폐색을 처리합니다. 조류 추적을위한 알고리즘에 대한 결과는 그림 11 (b)를 참조하십시오.

5.1.2. Statistical Methods for Correspondence. Measurements obtained from video sensors invariably contain noise. Moreover, the object motions can undergo random perturbations, for instance, maneuvering vehicles. Statistical correspondence methods solve these tracking problems by taking the measurement and the model uncertainties into account during object state estimation. The statistical correspondence methods use the state space approach to model the object properties such as position, velocity, and acceleration. Measurements usually consist of the object position in the image, which is obtained by a detection mechanism. Followings, we will discuss the state estimation methods in the context of point tracking, however, it should be noted that these methods can be used in general to estimate the state of any time varying system. For example, these methods have extensively been used for tracking contours [Isard and Blake 1998], activity recognition [Vaswani et al. 2003], object identification [Zhou et al. 2003], and structure from motion [Matthies et al. 1989].

Consider a moving object in the scene. The information representing the object, for example, location, is defined by a sequence of states X t : t = 1, 2, .... The change in state over time is governed by the dynamic equation,

통신에 대한 통계 방법. 비디오 센서에서 얻은 측정 값에는 항상 노이즈가 포함되어 있습니다. 또한, 물체의 움직임은 예를 들어 차량을 조작하는 것과 같은 임의의 섭동을 겪을 수 있습니다. 통계적 일치 방법은 개체 상태 추정 중에 측정 및 모델 불확실성을 고려하여 이러한 추적 문제를 해결합니다. 통계적 일치 방법은 상태 공간 접근법을 사용하여 위치, 속도 및 가속도와 같은 객체 속성을 모델링합니다. 측정은 대개 감지 메커니즘으로 얻은 이미지의 대상 위치로 구성됩니다. 다음으로, 우리는 점 추적의 맥락에서 상태 추정 방법을 논의 할 것이지만,이 방법은 일반적으로 시간 변화 시스템의 상태를 평가하는 데 사용될 수 있다는 점에 주목해야한다. 예를 들어, 이러한 방법은 윤곽 추적에 광범위하게 사용되었다 [Isard and Blake 1998], 활동 인식 [Vaswani et al. 2003], 객체 식별 [Zhou et al. 2003], 운동으로부터의 구조 [Matthies et al. 1989].

장면에서 움직이는 물체를 생각해보십시오. 물체를 나타내는 정보, 예를 들어 위치는 일련의 상태 X t : t = 1, 2, ...에 의해 정의된다. 시간에 따른 상태의 변화는 동적 방정식에 의해 지배되며,



where Wt : t = 1, 2, ... is white noise. The relationship between the measurement and the state is specified by the measurement equation Zt = ht (X t , Nt ), where Nt is the white noise and is independent of Wt . The objective of tracking is to estimate the state X t given all the measurements up to that moment or, equivalently, to construct the probability density function p(X t |Z1,...,t ). A theoretically optimal solution is provided by a recursive Bayesian filter which solves the problem in two steps. The prediction step uses a dynamic equation and the already computed pdf of the state at time t − 1 to derive the prior pdf of the current state, that is, p(X t |Z1,...,t−1). Then, the correction step employs the likelihood function p(Zt |X t ) of the current measurement to compute the posterior pdf p(X t |Z1,...,t ). In the case where the measurements only arise due to the presence of a single object in the scene, the state can be simply estimated by the two steps as defined. On the other hand, if there are multiple objects in the scene, measurements need to be associated with the corresponding object states. We now discuss the two cases.

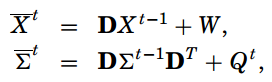
여기서 Wt : t = 1, 2, ...는 백색 잡음이다. 측정과 상태 사이의 관계는 측정 공식 Zt = ht (Xt, Nt)에 의해 지정됩니다. 여기서 Nt는 백색 잡음이며 Wt와는 독립적입니다. 추적의 목적은 그 순간까지의 모든 측정치가 주어 졌을 때 상태 X t를 추정하거나 등가 적으로 확률 밀도 함수 p (X t | Z 1, ..., t)를 구성하는 것입니다. 이론적으로 최적의 솔루션은 2 단계로 문제를 해결하는 재귀 베이지안 필터에 의해 제공됩니다. 예측 단계에서는 현재 상태 즉, p (Xt | Z1, ..., t-1)의 이전 pdf를 유도하기 위해 시간 t - 1에서 동적 방정식과 이미 계산 된 상태의 pdf를 사용합니다. 그런 다음 보정 단계에서는 현재 측정치의 우도 함수 p (Zt | Xt)를 사용하여 사후 pdf p (Xt | Z1, ..., t)를 계산합니다. 장면에서 단일 객체의 존재로 인해 측정 만 발생하는 경우 정의 된대로 두 단계로 상태를 간단히 추정 할 수 있습니다. 반면에 장면에 여러 객체가있는 경우 측정은 해당 객체 상태와 연관되어야합니다. 우리는 이제 두 가지 경우에 대해 논의합니다.

5.1.2.1. Single Object State Estimation. For the single object case, if f t and ht are linear functions and the initial state X 1 and noise have a Gaussian distribution, then the optimal state estimate is given by the Kalman Filter. In the general case, that is, object state is not assumed to be a Gaussian, state estimation can be performed using particle filters [Tanizaki 1987].

단일 객체 상태 추정. 단일 객체의 경우, f t와 ht가 선형 함수이고 초기 상태 X 1과 잡음이 가우시안 분포를 갖는다면, 최적 상태 추정은 칼만 필터에 의해 주어진다. 일반적으로 객체 상태가 가우시안으로 가정되지 않는 경우, 상태 추정은 입자 필터를 사용하여 수행 될 수있다 [Tanizaki 1987].

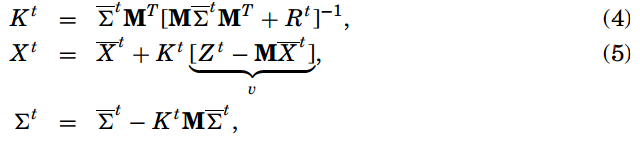
—Kalman Filters. A Kalman filter is used to estimate the state of a linear system where the state is assumed to be distributed by a Gaussian. Kalman filtering is composed of two steps, prediction and correction. The prediction step uses the state model to predict the new state of the variables:

칼만 필터. 칼만 필터 (Kalman filter)는 상태가 가우스에 의해 분포되는 것으로 가정되는 선형 시스템의 상태를 추정하는 데 사용됩니다. 칼만 필터링은 예측과 수정의 두 단계로 구성됩니다. 예측 단계에서는 상태 모델을 사용하여 변수의 새 상태를 예측합니다.



where X t and t are the state and the covariance predictions at time t. D is the state transition matrix which defines the relation between the state variables at time t and t − 1. Q is the covariance of the noise W. Similarly, the correction step uses the current observations Zt to update the object’s state:

여기서 X t와 t는 시간 t에서의 상태와 공분산 예측이다. D는 시간 t와 t - 1에서 상태 변수 간의 관계를 정의하는 상태 전이 행렬이다. Q는 잡음 W의 공분산이다. 유사하게, 보정 단계는 현재 관측치 Zt를 사용하여 객체의 상태를 갱신한다 :



where v is called the innovation, M is the measurement matrix, K is the Kalman gain, which is the Riccati Equation (4) used for propagation of the state models. Note that the updated state, X t is still distributed by a Gaussian. In case the functions f t and ht are nonlinear, they can be linearized using the Taylor series expansion to obtain the extended Kalman filter [Bar-Shalom and Foreman 1988]. Similar to the Kalman filter, the extended Kalman filter assumes that the state is distributed by a Gaussian.

여기서 v는 혁신이라고 불리며, M은 측정 행렬이고, K는 칼만 게인이며, 상태 모델의 전파에 사용되는 Riccati 방정식 (4)입니다. 업데이트 된 상태 X t는 여전히 가우스에 의해 분산됩니다. 함수 f t와 ht가 비선형 인 경우 Taylor 계열 확장을 사용하여 선형화하여 확장 된 칼만 필터를 얻을 수 있습니다 [Bar-Shalom and Foreman 1988]. 칼만 필터와 유사하게, 확장 된 칼만 필터는 상태가 가우시안에 의해 분산되어 있다고 가정한다.

The Kalman filter has been extensively used in the vision community for tracking. Broida and Chellappa [1986] used the Kalman filter to track points in noisy images. In stereo camera-based object tracking, Beymer and Konolige [1999] use the Kalman filter for predicting the object’s position and speed in x − z dimensions. Rosales and Sclaroff [1999] use the extended Kalman filter to estimate 3D trajectory of an object from 2D motion. A Matlab toolbox for Kalman filtering is available at KalmanSrc.

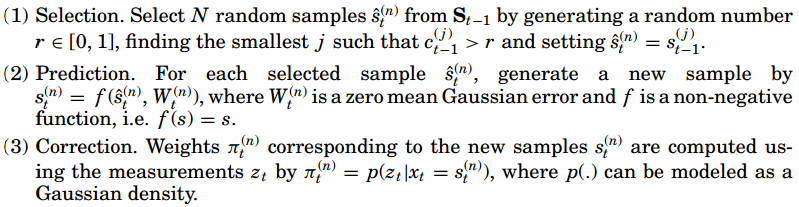
Kalman 필터는 추적을 위해 비전 커뮤니티에서 광범위하게 사용되었습니다. Broida와 Chellappa [1986]는 칼만 필터를 사용하여 잡음이 많은 이미지의 포인트를 추적했다. 스테레오 카메라 기반 객체 추적에서 Beymer와 Konolige [1999]는 x - z 차원에서 객체의 위치와 속도를 예측하기 위해 칼만 필터를 사용합니다. Rosales와 Sclaroff [1999]는 확장 된 칼만 필터를 사용하여 2D 모션으로부터 대상의 3D 궤적을 추정한다. KalmanSrc에서 Kalman 필터링을위한 Matlab 도구 상자를 사용할 수 있습니다.

—Particle Filters. One limitation of the Kalman filter is the assumption that the state variables are normally distributed (Gaussian). Thus, the Kalman filter will give poor estimations of state variables that do not follow Gaussian distribution. This limitation can be overcome by using particle filtering [Tanizaki 1987]. In particle filtering, the conditional state density p(X t|Zt) at time t is represented by a set of samples {s(n) t : n = 1, ... , N} (particles) with weights π(n) t (sampling probability). The weights define the importance of a sample, that is, its observation frequency [Isard and Blake 1998].

파티클 필터. 칼만 필터의 한 가지 한계는 상태 변수가 정규 분포 (가우스)라는 가정이다. 따라서, 칼만 필터는 가우시안 분포를 따르지 않는 상태 변수에 대한 잘못된 평가를 제공합니다. 이 제한은 입자 필터링을 사용함으로써 극복 될 수있다 [Tanizaki 1987]. 입자 필터링에서, 시간 t에서의 조건부 밀도 p (Xt | Zt)는 가중치 π (n (n) t : ) t (샘플링 확률). 가중치는 표본의 중요성, 즉 관측 빈도를 정의합니다 [Isard and Blake 1998].

To decrease computational complexity, for each tuple , a cumulative weight c(n) is also stored, where c(N) = 1. The new samples at time t are drawn from  at the previous time t − 1 step based on different sampling schemes [MacKay 1998]. The most common sampling scheme is importance sampling which can be stated as follows.

계산 복잡도를 줄이기 위해 각 튜플 에 대해 c (N) = 1 인 누적 가중치 c (n)도 저장됩니다. 시간 t에서의 새로운 샘플들은 다른 샘플링 방식에 기초하여 이전 시간 t - 1 단계에서 로부터 유도된다. 가장 일반적인 표본 추출 체계는 중요도 표본 추출로 다음과 같이 기술 할 수 있습니다.



Using the new samples St, one can estimate the new object position by .

새로운 샘플 St를 사용하여, 에 의한 새로운 오브젝트 위치를 추정 할 수 있다.

Particle filter-based trackers can be initialized by either using the first measurements, or by training the system using sample sequences.

입자 필터 기반 추적기는 첫 번째 측정 값 를 사용하거나 샘플 시퀀스를 사용하여 시스템을 학습하여 초기화 할 수 있습니다.

In addition to keeping track of the best particles, an additional resampling is usually employed to eliminate samples with very low weights. Note that the posterior density does not have to be a Gaussian. Particle filters recently became popular in computer vision. They are especially used for object detection and tracking. A Matlab toolbox for tracking using particle filtering is available at ParticleFltSrc. Note that the Kalman filter and particle filter described assume a single measurement at each time instant, that is, the state of a single object is estimated. Tracking multiple objects requires a joint solution of data association and state estimation problems.

최상의 입자를 추적하는 것 외에도 매우 낮은 무게의 시료를 제거하기 위해 추가 재 샘플링이 일반적으로 사용됩니다. posterior density는 Gaussian 일 필요는 없다. 입자 필터는 최근 컴퓨터 비전에서 널리 사용되었습니다. 특히 객체 감지 및 추적에 사용됩니다. 입자 필터링을 사용하여 추적하기위한 Matlab 도구 상자는 ParticleFltSrc에서 사용할 수 있습니다. 기술 된 칼만 필터 및 입자 필터는 매 시간마다 단일 측정을 가정한다. 즉, 단일 물체의 상태가 추정된다. 여러 개체를 추적하려면 데이터 연결 및 상태 추정 문제에 대한 공동 솔루션이 필요합니다.

5.1.2.2. Multiobject Data Association and State Estimation. When tracking multiple objects using Kalman or particle filters, one needs to deterministically associate the most likely measurement for a particular object to that object’s state, that is, the correspondence problem needs to be solved before these filters can be applied. The simplest method to perform correspondence is to use the nearest neighbor approach. However, if the objects are close to each other, then there is always a chance that the correspondence is incorrect. An incorrectly associated measurement can cause the filter to fail to converge. There exist several statistical data association techniques to tackle this problem. A detailed review of these techniques can be found in the book by Fortmann and Bar-Shalom [1988] or in the survey by Cox [1993]. Joint Probability Data Association Filtering (JPDAF) and Multiple Hypothesis Tracking (MHT) are two widely used techniques for data association. We give a brief description of these techniques in the following. —Joint Probability Data Association Filter. Let a track be defined as a sequence of measurements that are assumed to originate from the same object. Suppose we have N tracks and, at time t, Z(t) = z1(t), ··· zmt (t) are the m measurements. We need to assign these measurements to the existing tracks. Let η be a set of assignments. It is assumed that the number of tracks will remain constant over time. Let vi,l be the innovation (see the discussion on the Kalman Filter) associated with the track l due to the measurement zi. The JPDAF associates all measurements with each track. The combined weighted innovation is given by

Multiobject Data Association과 State Estimation. Kalman 또는 입자 필터를 사용하여 여러 객체를 추적 할 때 특정 객체에 대한 가장 가능성있는 측정을 객체의 상태와 결정적으로 연관시켜야합니다. 즉, 대응 문제를 해결해야 이러한 필터를 적용 할 수 있습니다. 서신을 수행하는 가장 간단한 방법은 가장 가까운 이웃 접근 방식을 사용하는 것입니다. 그러나 객체가 서로 가깝다면 항상 일치가 잘못 될 가능성이 있습니다. 잘못 연결된 측정으로 인해 필터가 수렴하지 못할 수 있습니다. 이 문제를 해결하기위한 몇 가지 통계 데이터 연관 기술이 존재합니다. 이러한 기술에 대한 자세한 검토는 Fortmann과 Bar-Shalom [1988]의 책자 또는 Cox [1993]의 설문서에서 찾을 수 있습니다. JPDAF (Joint Probability Data Association Filtering) 및 MHT (Multiple Hypothesis Tracking)는 데이터 연관을 위해 널리 사용되는 두 가지 기술입니다. 이 기술에 대한 간단한 설명을 다음과 같이 제공합니다. - 공동 확률 데이터 연관 필터. 하나의 트랙을 동일한 객체에서 발생한 것으로 간주되는 일련의 측정으로 정의합니다. N 개의 트랙이 있고 시간 t에서 Z (t) = z1 (t), ... zmt (t)가 m 개의 측정 값이라고 가정합니다. 우리는 이러한 측정 값을 기존 트랙에 할당해야합니다. η를 할당 집합이라하자. 시간이 지남에 따라 트랙의 수가 일정하게 유지된다고 가정합니다. vi, l을 측정 zi로 인한 트랙 l과 연관된 혁신 (칼만 필터에 대한 논의 참조)이라고하자. JPDAF는 모든 측정치를 각 트랙과 관련시킨다. 통합 가중 혁신은



where is the posterior probability that the measurement i originated from the object associated with track l and is given as:

여기서 는 측정치 i가 트랙 l과 연관된 객체에서 기인 한 사후 확률이며 다음과 같이 주어진다 :



where  is the indicator variable, with i = 1, ... , mk and l = 1, ... , N. It is equal to one if the measurement zi(k) is associated with track l, otherwise it is zero. The weighted innovation given in Equation (6) can be plugged into the Kalman filter update Equations (5) for each track l.

여기서 는 지시자 변수이고, i = 1, ..., mk 및 l = 1, ..., N이다. 측정 값 zi (k)가 트랙 l과 연관된다면 1과 같고, 그렇지 않으면 0이다. . 식 (6)에서 주어진 가중 된 혁신은 각 트랙 (l)에 대한 칼만 필터 업데이트 식 (5)에 꽂을 수있다.

JPDAF is used by Chang and Aggarwal [1991] to perform 3D structure reconstruction from a video sequence. Rasmussen and Hager [2001] use a constrained JPDAF filter to track regions. The major limitation of the JPDAF algorithm is its inability to handle new objects entering the field of view (FOV) or already tracked objects exiting the FOV. Since the JPDAF algorithm performs data association of a fixed number of objects tracked over two frames, serious errors can arise if there is a change in the number of objects. The MHT algorithm, which is explained next, does not have this shortcoming.

JPDAF는 비디오 시퀀스에서 3D 구조 재구성을 수행하기 위해 Chang과 Aggarwal [1991]에 의해 사용됩니다. Rasmussen과 Hager [2001]는 제한된 JPDAF 필터를 사용하여 영역을 추적합니다. JPDAF 알고리즘의 주된 한계는 FOV (FOV)로 들어오는 새 객체 또는 FOV를 종료하는 이미 추적 된 객체를 처리 할 수 ​​없다는 것입니다. JPDAF 알고리즘은 두 프레임에 걸쳐 추적되는 고정 된 수의 객체의 데이터 연관을 수행하기 때문에 객체의 수가 변경되면 심각한 오류가 발생할 수 있습니다. 다음에 설명하는 MHT 알고리즘에는 이러한 단점이 없습니다.

—Multiple Hypothesis Tracking (MHT ). If motion correspondence is established using only two frames, there is always a finite chance of an incorrect correspondence. Better tracking results can be obtained if the correspondence decision is deferred until several frames have been examined. The MHT algorithm maintains several correspondence hypotheses for each object at each time frame [Reid 1979]. The final track of the object is the most likely set of correspondences over the time period of its observation. The algorithm has the ability to create new tracks for objects entering the FOV and terminate tracks for objects exiting the FOV. It can also handle occlusions, that is, continuation of a track even if some of the measurements from an object are missing.

다중 가설 추적 (MHT). 두 프레임만을 사용하여 동작 대응이 설정되면 항상 잘못된 대응이 가능할 수 있습니다. 통신 결정이 여러 프레임이 검사 될 때까지 연기되면 더 나은 추적 결과를 얻을 수 있습니다. MHT 알고리즘은 매 시간 프레임마다 각 객체에 대한 여러 대응 문언을 유지한다 [Reid 1979]. 객체의 최종 트랙은 관찰 기간 동안의 가장 가능성있는 대응 세트입니다. 이 알고리즘은 FOV에 들어가는 객체에 대한 새 트랙을 생성하고 FOV에서 나가는 객체에 대한 트랙을 종료하는 기능이 있습니다. 또한 객체의 측정 값 중 일부가 누락 되어도 오클루젼, 즉 트랙의 연속을 처리 할 수 ​​있습니다.

MHT is an iterative algorithm. An iteration begins with a set of current track hypotheses. Each hypothesis is a collection of disjoint tracks. For each hypothesis, a prediction of each object’s position in the next frame is made. The predictions are then compared with actual measurements by evaluating a distance measure. A set of correspondences (associations) are established for each hypothesis based on the distance measure which introduces new hypotheses for the next iteration. Each new hypothesis represents a new set of tracks based on the current measurements. Note that each measurement can belong to a new object entering the FOV, a previously tracked object, or a spurious measurement. Moreover, a measurement may not be assigned to an object because the object may have exited the FOV, or a measurement corresponding to an object may not be obtained. The latter happens because either the object is occluded or it is not detected due to noise.

MHT는 반복 알고리즘입니다. 반복은 현재 트랙 가설의 집합으로 시작됩니다. 각 가설은 분리 된 트랙의 모음입니다. 각각의 가설에 대해, 다음 프레임에서 각각의 대상의 위치의 예측이 행해진 다. 그런 다음 거리 측정을 평가하여 예측을 실제 측정과 비교합니다. 일련의 대응 (연관)이 다음 반복에 대한 새로운 가설을 도입하는 거리 측정치에 기초하여 각 가설에 대해 설정된다. 각각의 새로운 가설은 현재의 측정에 기초한 새로운 트랙 세트를 나타낸다. 각 측정은 FOV에 들어가는 새 객체, 이전에 추적 된 객체 또는 가짜 측정에 속할 수 있습니다. 또한, 물체가 FOV를 빠져 나가거나 물체에 해당하는 측정 값을 얻을 수 없기 때문에 물체에 측정이 할당되지 않을 수 있습니다. 후자는 물체가 가려 지거나 노이즈로 인해 감지되지 않기 때문에 발생합니다.

Note that MHT makes associations in a deterministic sense and exhaustively enumerates all possible associations. To reduce the computational load, Streit and Luginbuhl [1994] propose a probabilistic MHT (PMHT) in which the associations are considered to be statistically independent random variables and thus there is no requirement for exhaustive enumeration of associations. Recently, particle filters that handle multiple measurements to track multiple objects have been proposed by Hue et al. [2002]. In their method, data association is handled in a similar way as in PMHT, however, the state estimation is achieved through particle filters.

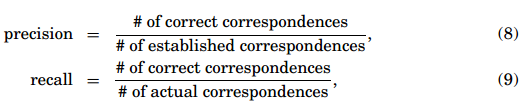
The MHT algorithm is computationally exponential both in memory and time. To overcome this limitation, Cox and Hingorani [1996] use Murty’s [1968] algorithm to determine k-best hypotheses in polynomial time for tracking interest points. Cham and Rehg [1999] use the multiple hypothesis framework to track the complete human body.

MHT는 결정적인 의미로 연관을 만들고 모든 가능한 연관을 철저히 나열합니다. 계산량을 줄이기 위해 Streit and Luginbuhl [1994]은 연관성이 통계적으로 독립적 인 확률 변수로 간주되는 확률 론적 MHT (PMHT)를 제안하므로 연관성의 철저한 계산을 요구하지 않는다. 최근, 여러 객체를 추적하기 위해 여러 측정을 처리하는 입자 필터가 Hue et al. [2002]. 그들의 방법에서 데이터 연관성은 PMHT와 비슷한 방식으로 처리되지만, 상태 추정은 입자 필터를 통해 이루어집니다.

MHT 알고리즘은 메모리와 시간 모두에서 계산적으로 지수 함수입니다. 이 한계를 극복하기 위해 Cox와 Hingorani [1996]는 관심 지점을 추적하기 위해 다항식 시간에 k-best 가설을 결정하기 위해 Murty [1968] 알고리즘을 사용합니다. Cham과 Rehg [1999]는 다중 가설 프레임 워크를 사용하여 완전한 인체를 추적합니다.

5.1.3. Discussion. Point tracking methods can be evaluated on the basis of whether they generate correct point trajectories. Given a ground truth, the performance can be evaluated by computing precision and recall measures. In the context of point tracking, precision and recall measures can be defined as:

토론. 포인트 추적 방법은 올바른 포인트 궤도를 생성하는지 여부에 따라 평가할 수 있습니다. 근거 진실을 감안할 때, 성능은 정확도 및 리콜 측정을 계산하여 평가할 수 있습니다. 포인트 추적과 관련하여 정밀도 및 리콜 측정은 다음과 같이 정의 할 수 있습니다.



where actual correspondences denote the correspondences available in the ground truth. Additionally, a qualitative comparison of object trackers can be made based on their ability to

실제적인 대응은 지상 진실에서 이용 가능한 대응을 나타낸다. 또한 객체 추적기의 질적 비교는

—deal with entries of new objects and exits of existing objects,

새로운 객체의 입력과 기존 객체의 종료가있는 경우

—handle the missing observations (occlusion), and

실종 관측 (폐색), 그리고

—provide an optimal solution to the cost function minimization problem used for establishing correspondence.

통신을 설정하기 위해 사용되는 비용 함수 최소화 문제에 대한 최적 솔루션을 제공합니다.

In Table III, we provide a qualitative comparison based on these properties.

One important issue in the context of point trackers is the handling of missing or noisy observations. To address these problems, deterministic point trackers often use a combination of motion-based constraints addressed in Section 5.1.1, that is, common motion [Veenman et al. 2001] or proximal uniformity [Rangarajan and Shah 1991]. Statistical point tracking methods explicitly handle noise by taking model uncertainties into consideration. These uncertainties are usually assumed to be in the form of normally distributed noise. However, the assumption that measurements are normally distributed around their predicted position may not hold. Moreover, in many cases, the noise parameters are not known. In the case of valid assumptions on distributions and noise, Kalman filters [Bar-Shalom and Foreman 1988] and MHT [Reid 1979] give optimal solutions. Another possible approach to handling noise and missing observations is to enforce constraints that define the object’s 3D structure. For instance, multibody factorization methods can be used for handling noisy observations by enforcing the object points to fit into the 3D object shape. This is addressed for the nonrigid object by Bregler et al. [Torresani and Bregler 2002; Bregler et al. 2000] where the authors first define a set of shape bases from a set of reliable tracks which has minimum or no appearance error on the points trajectory. Computed shape basis then serves as a constraint on the remaining point trajectories that are labeled as unreliable.

표 3에서 우리는 이러한 특성에 기초한 질적 비교를 제공한다.

포인트 추적기와 관련된 중요한 문제 중 하나는 누락되거나 시끄러운 관측을 처리하는 것입니다. 이러한 문제를 해결하기 위해 결정 론적 포인트 추적기는 5.1.1 절에서 다루는 모션 기반 제약, 즉 일반적인 모션의 조합을 사용합니다 [Veenman et al. 2001] 또는 근위 균일 성 [Rangarajan and Shah 1991]. 통계적 포인트 추적 방법은 모델 불확실성을 고려하여 노이즈를 명시 적으로 처리합니다. 이러한 불확도는 일반적으로 정규 분포 잡음의 형태로 가정된다. 그러나 측정치가 예상 위치 주변에 정상적으로 분포된다고 가정하지 못할 수도 있습니다. 더욱이, 많은 경우에, 잡음 파라미터는 알려지지 않았다. 분포와 잡음에 대한 유효한 가정의 경우, 칼만 필터 [Bar-Shalom and Foreman 1988]와 MHT [Reid 1979]는 최적 해답을 제시한다. 노이즈 및 누락 된 관측을 처리하는 또 다른 가능한 접근법은 객체의 3D 구조를 정의하는 제약 조건을 적용하는 것입니다. 예를 들어, 3D 객체 모양에 맞도록 객체 점을 적용하여 잡음이 많은 관측을 처리하는 데 다중 보자력 분해 방법을 사용할 수 있습니다. 이것은 Bregler et al. [Torresani and Bregler 2002; Bregler et al. 2000] 여기서 저자는 처음에 점 궤적에 최소한의 출현 오류를 갖는 신뢰할 수있는 궤적으로부터 쉐이프 기반의 집합을 정의한다. 계산 된 쉐이프 기반은 신뢰할 수없는 것으로 표시된 나머지 점 궤도에 대한 제약 조건으로 사용됩니다.

|  |
| --- |
|  |
| Table III. Qualitative Comparison of Point Trackers (#: number of objects, M: multiple objects, S single object. Symbols √ and × denote whether the tracker can or cannot handle occlusions, object entries object exits, and provide the optimal solution.)  포인트 추적기의 정 성적 비교 (# : 객체 수, M : 다중 객체, S 단일 객체) √ 및 × 기호는 추적기가 폐색을 처리 할 수 있는지 여부를 나타내며 객체 입력 개체가 종료되고 최적의 솔루션을 제공합니다. |

Point trackers are suitable for tracking very small objects which can be represented by a single point (single point representation). Multiple points are needed to track larger objects. In the context of tracking objects using multiple points, automatic clustering of points that belong to the same object is an important problem. This is due to the need to distinguish between multiple objects and, between objects and background. Motion-based clustering or segmentation approaches [Vidal and Ma 2004; Black and Anandan 1996; Wang and Adelson 1994] usually assume that the points being tracked lie on rigid bodies in order to simplify the segmentation problem.

점 추적기는 단일 점 (단일 점 표현)으로 표현할 수있는 매우 작은 객체를 추적하는 데 적합합니다. 큰 개체를 추적하려면 여러 개의 점이 필요합니다. 여러 점을 사용하여 객체를 추적하는 맥락에서 동일한 객체에 속하는 점의 자동 클러스터링은 중요한 문제입니다. 이것은 여러 개체와 개체와 배경을 구별 할 필요가 있기 때문입니다. 모션 기반 클러스터링 또는 세분화 접근 [Vidal and Ma 2004; 블랙 앤 아난드 1996; Wang and Adelson 1994]는 보통 세그멘테이션 문제를 단순화하기 위해 추적되는 점이 강체에 있다고 가정한다.

**5.2. Kernel Tracking**

Kernel tracking is typically performed by computing the motion of the object, which is represented by a primitive object region, from one frame to the next. The object motion is generally in the form of parametric motion (translation, conformal, affine, etc.) or the dense flow field computed in subsequent frames. These algorithms differ in terms of the appearance representation used, the number of objects tracked, and the method used to estimate the object motion. We divide these tracking methods into two subcategories based on the appearance representation used, namely, templates and density-based appearance models, and multiview appearance models.

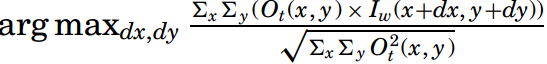
커널 추적은 일반적으로 한 프레임에서 다음 프레임으로 원시 객체 영역으로 표현되는 객체의 동작을 계산하여 수행됩니다. 오브젝트 모션은 일반적으로 파라 메트릭 모션 (translation, conformal, affine 등) 또는 후속 프레임에서 계산되는 고밀도 플로우 필드의 형태입니다. 이러한 알고리즘은 사용 된 모양 표현, 추적 된 개체 수 및 개체 동작을 추정하는 데 사용되는 방법이 다릅니다. 이 추적 방법은 템플릿 및 밀도 기반 모양 모델 및 다중보기 모양 모델과 같이 사용 된 모양 표현을 기반으로 두 가지 하위 범주로 나뉩니다.

5.2.1. Tracking Using Template and Density-Based Appearance Models. Templates and density-based appearance models (see Section 2) have been widely used because of the their relative simplicity and low computational cost. We divide the trackers in this category into two subcategories based on whether the objects are tracked individually or jointly.

템플릿 및 밀도 기반 모양 모델을 사용한 추적. 템플릿 및 밀도 기반 모양 모델 (2 절 참조)은 상대적으로 단순하고 계산 비용이 적기 때문에 널리 사용되었습니다. 이 카테고리의 추적 프로그램은 개체가 개별적으로 또는 공동으로 추적되는지 여부에 따라 두 개의 하위 범주로 나뉩니다.

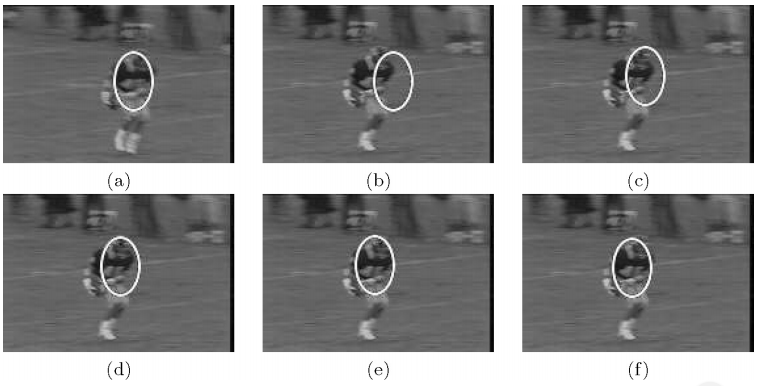
5.2.1.1. Tracking single objects. The most common approach in this category is template matching. Template matching is a brute force method of searching the image, Iw, for a region similar to the object template, Ot defined in the previous frame. The position of the template in the current image is computed by a similarity measure, for example, cross correlation:

단일 개체를 추적합니다. 이 카테고리의 가장 일반적인 접근법은 템플릿 매칭입니다. 템플릿 매칭은 이전 프레임에서 정의 된 객체 템플릿과 유사한 영역에 대해 이미지 Iw를 검색하는 무차별 한 방법입니다. 현재 이미지에서 템플릿의 위치는 유사성 척도, 예를 들어 상호 상관에 의해 계산됩니다.



where (d x, d y) specify the candidate template position. Usually image intensity or color features are used to form the templates. Since image intensity is very sensitive to illumination changes, image gradients [Birchfield 1998] can also be used as features. A limitation of template matching is its high computation cost due to the brute force search. To reduce the computational cost, researchers usually limit the object search to the vicinity of its previous position. Also, more efficient algorithms for template matching have been proposed [Schweitzer et al. 2002].

여기서 (d x, d y)는 후보 템플릿 위치를 지정합니다. 일반적으로 이미지 강도 또는 색상 피쳐가 템플릿을 형성하는 데 사용됩니다. 이미지 강도는 조명 변화에 매우 민감하기 때문에 이미지 그라데이션 [Birchfield 1998]을 피쳐로 사용할 수 있습니다. 템플리트 매칭의 한계는 무차별 대입 (brute force search)으로 인한 높은 계산 비용입니다. 계산 비용을 줄이기 위해 연구자들은 일반적으로 객체 검색을 이전 위치 부근으로 제한합니다. 또한, 템플릿 정합을위한보다 효율적인 알고리즘이 제안되었다 [Schweitzer et al. 2002].



Note that instead of templates, other object representations can be used for tracking, for instance, color histograms or mixture models can be computed by using the appearance of pixels inside the rectangular or ellipsoidal regions. Fieguth and Terzopoulos [1997] generate object models by finding the mean color of the pixels inside the rectangular object region. To reduce computational complexity, they search the object in eight neighboring locations. The similarity between the object model, M, and the hypothesized position, H, is computed by evaluating the ratio between the color means computed from M and H. The position which provides the highest ratio is selected as the current object location.

Comaniciu and Meer [2003] use a weighted histogram computed from a circular region to represent the object. Instead of performing a brute force search for locating the object, they use the mean-shift procedure (Section 4.3).

템플릿 대신에 다른 객체 표현을 사용하여 추적 할 수 있습니다. 예를 들어 색상 히스토그램 또는 직사각형 또는 타원형 영역 내부의 픽셀 모양을 사용하여 혼합 모델을 계산할 수 있습니다. Fieguth and Terzopoulos [1997]는 직사각형 객체 영역 내부의 픽셀의 평균 색을 찾아 객체 모델을 생성한다. 계산의 복잡성을 줄이기 위해 이웃 한 8 개 위치에서 객체를 검색합니다. 객체 모델 M과 가정 된 위치 H 사이의 유사성은 M과 H에서 계산 된 색상 평균 간의 비율을 계산하여 계산됩니다. 가장 높은 비율을 제공하는 위치가 현재 객체 위치로 선택됩니다.

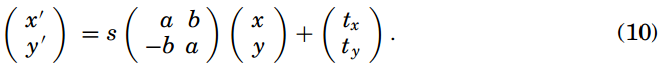
Comaniciu and Meer [2003]는 원형 영역에서 계산 된 가중치 히스토그램을 사용하여 객체를 나타냅니다. 객체를 찾기 위해 무차별 강제 검색을 수행하는 대신 평균 변환 절차 (4.3 절)를 사용합니다.

The mean-shift tracker maximizes the appearance similarity iteratively by comparing the histograms of the object, Q, and the window around the hypothesized object location, P. Histogram similarity is defined in terms of the Bhattacharya coefficient, , where b is the number of bins. At each iteration, the mean-shift vector is computed such that the histogram similarity is increased.

평균 이동 추적기는 객체 Q의 히스토그램과 가정 된 객체 위치 P 주위의 창을 비교하여 모양 유사성을 반복적으로 최대화합니다. 히스토그램 유사성은 Bhattacharya 계수 으로 정의됩니다. 여기서 b는 쓰레기통. 각각의 반복에서, 히스토그램 유사성이 증가하도록 평균 - 이동 벡터가 계산된다.

This process is repeated until convergence is achieved, which usually takes five to six iterations. For histogram generation, the authors use a weighting scheme defined by a spatial kernel which gives higher weights to the pixels closer to the object center. Comaniciu [2002] extended the mean-shift tracking approach used a joint spatial-color histogram (Section 4.3) instead of just a color histogram. An example of mean-shift tracking is given in Figure 12. An obvious advantage of the meanshift tracker over the standard template matching is the elimination of a brute force search, and the computation of the translation of the object patch in a small number of iterations. However, mean-shift tracking requires that a portion of the object is inside the circular region upon initialization (part of the object has to be inside the white ellipse in Figure 12(b)). Implementation of the mean-shift tracker is available in OpenCV as CAMSHIFT at MeanShiftSegmentSrc. Jepson et al. [2003] propose an object tracker that tracks an object as a threecomponent mixture, consisting of the stable appearance features, transient features and noise process. The stable component identifies the most reliable appearance for motion estimation, that is, the regions of the object whose appearance does not quickly change over time. The transient component identifies the quickly changing pixels. The noise component handles the outliers in the object appearance that arise due to noise. An online version of the EM algorithm is used to learn the parameters of this threecomponent mixture. The authors use the phase of the steerable filter responses as features for appearance representation. The object shape is represented by an ellipse. The motion of the object is computed in terms of warping the tracked region from one frame to the next one. The warping transformation consists of translation, (tx , t y ), rotation (a, b), and scale, s, parameters:

이 과정은 수렴 (convergence)이 달성 될 때까지 반복되는데, 보통 수 차례 5 ~ 6 회의 반복이 필요합니다. 히스토그램 생성을 위해 저자는 객체 중심에 가까운 픽셀에 더 높은 가중치를 부여하는 공간 커널에 의해 정의 된 가중치 체계를 사용합니다. Comaniciu [2002]는 단순한 색상 히스토그램 대신 공동 공간 - 색상 히스토그램 (4.3 절)을 사용하여 평균 이동 추적 접근법을 확장했다. mean-shift 추적의 예가 그림 12에 나와 있습니다. 표준 템플릿 매칭보다 meanshift 추적 프로그램의 명백한 이점은 무차별 검색 (brute force search)을 제거하고 적은 반복 횟수로 대상 패치의 번역을 계산하는 것입니다 . 그러나 평균 이동 추적은 초기화시 원형 영역 내부에 객체의 일부가 있어야합니다 (객체의 일부는 그림 12 (b)의 흰색 타원 내부에 있어야합니다). mean-shift 추적기의 구현은 OpenCV에서 MeanShiftSegmentSrc의 CAMSHIFT로 사용할 수 있습니다. Jepson et al. [2003]은 안정된 외관 특징, 일시적인 특징 및 잡음 과정으로 구성된 3 성분 혼합물로서 대상을 추적하는 물체 추적 장치를 제안한다. 안정된 구성 요소는 동작 추정을위한 가장 신뢰할 수있는 모양, 즉 시간이 지남에 따라 외관이 빠르게 변하지 않는 대상의 영역을 식별합니다. transient 구성 요소는 빠르게 변화하는 픽셀을 식별합니다. 노이즈 구성 요소는 노이즈로 인해 발생하는 개체 외양의 아웃 라이어를 처리합니다. EM 알고리즘의 온라인 버전은이 threecomponent 혼합물의 매개 변수를 학습하는 데 사용됩니다. 저자는 조종 가능한 필터 응답의 위상을 외형 표현의 피쳐로 사용합니다. 객체 모양은 타원으로 표시됩니다. 대상의 모션은 한 프레임에서 다음 프레임으로 추적 된 영역을 뒤틀어서 계산됩니다. 변형 변환은 변환, (tx, ty), 회전 (a, b) 및 축척, 매개 변수로 구성됩니다.

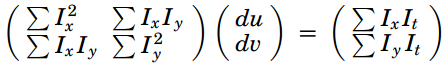


A weighted combination of the stable and transient components is used to determine the warping parameters. The advantage of learning stable and transient features is that one can give more weight to stable features for tracking, for example, if the face of a person who is talking is being tracked, then the forehead or nose region can give a better match to the face in the next frame as opposed to the mouth of the person (see Figure 13).

안정 및 과도 성분의 가중 조합을 사용하여 뒤틀림 매개 변수를 결정합니다. 안정적이고 일시적인 기능을 학습하는 장점은 안정된 기능에 더 많은 무게를 줄 수 있다는 것입니다. 예를 들어, 말하고있는 사람의 얼굴을 추적하면 이마 또는 코 영역이 더 나은 일치를 제공 할 수 있습니다. 사람의 입과는 대조적으로 다음 프레임에서 얼굴을 보게됩니다 (그림 13 참조).

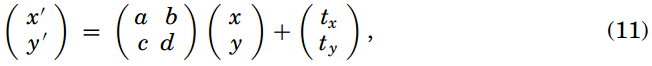
Another approach to track a region defined by a primitive shape is to compute its translation by use of an optical flow method. Optical flow methods are used for generating dense flow fields by computing the flow vector of each pixel under the brightness constancy constraint, I (x, y, t) − I (x + dx, y + dy, t + dt) = 0 [Horn and Schunk 1981]. This computation is always carried out in the neighborhood of the pixel either algebraically [Lucas and Kanade. 1981] or geometrically [Schunk 1986]. Extending optical flow methods to compute the translation of a rectangular region is trivial. In 1994, Shi and Tomasi proposed the KLT tracker which iteratively computes the translation (du, dv) of a region (e.g., 25 × 25 patch) centered on an interest point (for interest point detection, see Section 4.1):

프리미티브 모양으로 정의 된 영역을 추적하는 또 다른 방법은 옵티컬 플로 방식을 사용하여 변환을 계산하는 것입니다. 옵티컬 플로우 방법은 휘도 불변성 제약 하에서 각 픽셀의 플로우 벡터를 계산함으로써 조밀 한 유동장을 생성하는데 사용된다. I (x, y, t) - I (x + dx, y + dy, t + dt) = 0 [ Horn and Schunk 1981]. 이 계산은 대수적으로 [Lucas and Kanade] 픽셀 근처에서 수행됩니다. 1981] 또는 기하학적으로 [Schunk 1986]. 직사각형 영역의 평행 이동을 계산하기 위해 옵티컬 플로우 메서드를 확장하는 것은 쉽지 않습니다. 1994 년 Shi와 Tomasi는 관심 지점을 중심으로 한 지역 (예 : 25 × 25 패치)의 번역 (du, dv)을 반복 계산하는 KLT 추적 프로그램을 제안했습니다 (관심 지점 감지의 경우 4.1 참조).



This equation is similar in construction to the optical flow method proposed by Lucas and Kanade [1981]. Once the new location of the interest point is obtained, the KLT tracker evaluates the quality of the tracked patch by computing the affine transformation

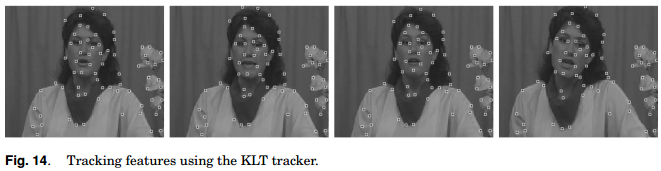
이 방정식은 **루카스 (Lucas)와 카나데 (Kanade)가 제안한 옵티컬 플로우 (optical flow) 방법**과 구조면에서 유사하다 [1981]. 일단 관심 지점의 새로운 위치가 획득되면, KLT 추적기는 아핀 변환을 계산함으로써 추적 된 패치의 품질을 평가한다



between the corresponding patches in consecutive frames. If the sum of square difference between the current patch and the projected patch is small, they continue tracking the feature, otherwise the feature is eliminated. The implementation of the KLT tracker is available at KLTSrc . The results obtained by the KLT tracker are shown in Figure 14.

연속 프레임의 해당 패치 사이. 현재 패치와 투영 된 패치의 제곱합 차가 작 으면 계속해서 해당 피쳐를 추적합니다. 그렇지 않으면 피쳐가 제거됩니다. KLTS 추적기의 구현은 KLTSrc에서 제공됩니다. KLT 추적기로 얻은 결과는 그림 14와 같습니다.

|  |
| --- |
|  |
| Fig. 13. Results of the robust online tracking method by Jepson et al. [2003] (a) The target region in different frames. (b) The mixing probability of the stable component. Note that the probabilities around the mouth and eyebrow regions change, while they remain the same in the other regions (c 2003 IEEE).  Jepson et al.에 의한 강력한 온라인 추적 방법의 결과 [2003] (a) 다른 프레임의 대상 영역. (b) 안정 성분의 혼합 확률. 입과 눈썹 주변의 확률은 변화하지만 다른 지역 (c 2003 IEEE)에서는 동일하게 유지됩니다. |



5.2.1.2. Tracking Multiple Objects. Modeling objects individually does not take into account the interaction between multiple objects and between objects and background during the course of tracking. An example interaction between objects can be one object partially or completely occluding the other. The tracking methods given in the following model the complete image, that is, the background and all moving objects are explicitly tracked.

여러 개체 추적. 개체를 개별적으로 모델링하는 것은 추적하는 동안 여러 개체 간의 상호 작용과 개체와 배경 간의 상호 작용을 고려하지 않습니다. 객체들 사이의 상호 작용의 예는 하나의 객체가 부분적으로 또는 완전히 다른 객체를 가릴 수 있습니다. 다음 추적 모델은 전체 이미지, 즉 배경 및 모든 움직이는 객체를 명시 적으로 추적합니다.

Tao et al. [2002] propose an object tracking method based on modeling the whole image, I t , as a set of layers. This representation includes a single background layer and one layer for each object. Each layer consists of shape priors (ellipse), , motion model (translation and rotation), , and layer appearance, A, (intensity modeled using a single Gaussian).

Tao et al. [2002]는 전체 이미지를 계층 집합으로 모델링하는 것에 기반한 객체 추적 방법을 제안한다. 이 표현에는 하나의 배경 레이어와 각 객체에 대한 하나의 레이어가 포함됩니다. 각 레이어는 모양 사전 (타원), 동작 모델 (병진 및 회전), 레이어 모양, A (단일 가우스를 사용하여 모델화 된 강도)로 구성됩니다.

Layering is performed by first compensating the background motion modeled by projective motion such that the object’s motion can be estimated from the compensated image using 2D parametric motion. Then, each pixel’s probability of belonging to a layer (object), pl , is computed based on the object’s previous motion and shape characteristics. Any pixel far from a layer is assigned a uniform background probability, pb. Later, the object’s appearance (intensity, color) probability pa is coupled with pl to obtain the final layer estimate.

레이어링은 투영 모션에 의해 모델링 된 배경 모션을 먼저 보상하여 2D 파라 메트릭 모션을 사용하여 보상 된 이미지에서 객체의 모션을 추정 할 수 있도록함으로써 수행됩니다. 그런 다음, 각 픽셀의 레이어 (객체)에 속하는 확률, p1은 객체의 이전 동작 및 모양 특성을 기반으로 계산됩니다. 레이어에서 멀리 떨어져있는 모든 픽셀에는 균일 한 배경 확률 pb가 할당됩니다. 나중에, 객체의 외양 (강도, 색상) 확률 pa는 pl과 결합되어 최종 층 추정치를 얻는다.

The model parameters (t, t, At) that maximize observing a layer at time t are estimated iteratively using an expectation maximization algorithm. However, due to the difficulty in simultaneous estimation of the parameters, the authors individually estimate one set, while fixing the others. For instance, they first estimate layer ownership using the intensity for each pixel, then they estimate the motion (rotation and translation) using appearance probabilities, and finally update layer ownership using this motion. The unknowns for each object are iteratively estimated until the layer ownership probabilities are maximized. Isard and MacCormick [2001] propose joint modeling of the background and foreground regions for tracking.

시간 t에서 계층을 관찰하는 것을 최대화하는 모델 파라미터 (t, t, At)는 기대 최대화 알고리즘을 사용하여 반복적으로 추정된다. 그러나 매개 변수의 동시 평가가 어렵 기 때문에 저자는 개별적으로 하나의 세트를 추정하면서 다른 세트는 고정시킨다. 예를 들어, 먼저 각 픽셀의 강도를 사용하여 레이어 소유권을 추정 한 다음 모양 확률을 사용하여 모션 (회전 및 평행 이동)을 추정하고 마지막으로이 모션을 사용하여 레이어 소유권을 업데이트합니다. 각 객체에 대한 미지수는 레이어 소유권 확률이 최대화 될 때까지 반복적으로 추정됩니다. Isard와 MacCormick [2001]은 추적을위한 배경 및 전경 영역의 공동 모델링을 제안한다.

The background appearance is represented by a mixture of Gaussians. Appearance of all foreground objects is also modeled by mixture of Gaussians. The shapes of objects are modeled as cylinders. They assume the ground plane is known, thus the 3D object positions can be computed. Tracking is achieved by using particle filters where the state vector includes the 3D position, shape and the velocity of all objects in the scene. They propose a modified prediction and correction scheme for particle filtering which can increase or decrease the size of the state vector to include or remove objects. The method can also tolerate occlusion between objects. However, the maximum number of objects in the scene is required to be predefined. Another limitation of the approach is the use of the same appearance model for all foreground objects, and it requires training to model the foreground regions.

배경 모양은 가우시안의 혼합으로 표현됩니다. 모든 전경 물체의 모양은 또한 가우시안의 혼합에 의해 모델링됩니다. 객체의 모양은 원통형으로 모델링됩니다. 그들은 지상 평면이 알려지고, 따라서 3D 오브젝트 위치가 계산 될 수 있다고 가정합니다. 추적은 상태 벡터가 장면의 모든 객체의 3D 위치, 모양 및 속도를 포함하는 입자 필터를 사용하여 수행됩니다. 이들은 물체를 포함하거나 제거하기 위해 상태 벡터의 크기를 늘리거나 줄일 수있는 입자 필터링을위한 수정 된 예측 및 수정 체계를 제안합니다. 이 방법은 객체 간의 교합을 허용 할 수도 있습니다. 그러나 장면의 최대 개체 수는 미리 정의해야합니다. 접근법의 또 다른 한계는 모든 전경 객체에 동일한 외형 모델을 사용하고 전경 영역을 모델링하는 데 필요한 교육입니다.

5.2.2. Tracking Using Multiview Appearance Models. In the previous tracking methods, the appearance models, that is, histograms, templates etc., are usually generated online. Thus these models represent the information gathered about the object from the most recent observations. The objects may appear different from different views, and if the object view changes dramatically during tracking, the appearance model may no longer be valid, and the object track might be lost. To overcome this problem, different views of the object can be learned offline and used for tracking.

다중 뷰 모양 모델을 사용한 추적. 이전의 추적 방법에서는 모양 모델, 즉 히스토그램, 템플릿 등이 일반적으로 온라인으로 생성됩니다. 따라서이 모델들은 가장 최근의 관측으로부터 물체에 관해 수집 된 정보를 나타냅니다. 개체가 다른보기와 다를 수 있으며 추적 중에 개체보기가 크게 변경되면 모양 모델이 더 이상 유효하지 않고 개체 트랙이 손실 될 수 있습니다. 이 문제를 극복하기 위해 개체의 다른 뷰를 오프라인에서 학습하고 추적에 사용할 수 있습니다.

In 1998, Black and Jepson proposed a subspace-based approach, that is, eigenspace, to compute the affine transformation from the current image of the object to the image reconstructed using eigenvectors. First, a subspace representation of the appearance of an object is built using Principal Component Analysis (PCA), then the transformation from the image to the eigenspace is computed by minimizing the so-called subspace constancy equation which evaluates the difference between the image reconstructed using the eigenvectors and the input image. Minimization is performed in two steps: finding subspace coefficients and computing affine parameters. In the first step, the affine parameters are fixed, and the subspace coefficients are computed. In the second step, using the new subspace coefficients, affine parameters are computed. Based on this, tracking is performed by estimating the affine parameters iteratively until the difference between the input image and the projected image is minimized. Note that the use of eigenspace for similarity computation is a useful alternative to standard template matching techniques such as SSD and normalized correlation. The eigenspace-based similarity computation is equivalent to matching with a linear combination of eigen templates. This allows for distortions in the templates, for example, distortion caused by illumination changes in images.

1998 년 Black과 Jepson은 객체의 현재 이미지에서 고유 벡터를 사용하여 재구성 된 이미지까지의 아핀 변환을 계산하기 위해 부분 공간 기반 접근법, 즉 고유 공간을 제안했습니다. 먼저 주체 성분 분석 (Principal Component Analysis, PCA)을 사용하여 객체의 외형을 부분 공간으로 표현한 다음 이미지에서 고유 공간으로의 변환을 사용하여 재구성 된 이미지와 고유 벡터들 및 입력 이미지. 최소화는 부분 공간 계수를 찾고 아핀 매개 변수를 계산하는 두 단계로 수행됩니다. 제 1 단계에서, 아핀 파라미터는 고정되고, 부분 공간 계수가 계산된다. 제 2 단계에서, 새로운 부분 공간 계수들을 사용하여, 아핀 파라미터들이 계산된다. 이를 토대로, 입력 영상과 투사 된 영상 사이의 차이가 최소화 될 때까지 아핀 파라미터를 반복적으로 추정함으로써 추적이 수행된다. 유사도 계산을위한 고유 공간의 사용은 SSD 및 정규화 상관 관계와 같은 표준 템플릿 매칭 기술에 대한 유용한 대안이라는 점에 유의하십시오. 고유 공간 기반 유사성 계산은 고유 템플리트의 선형 조합과 일치시키는 것과 같습니다. 이렇게하면 템플릿의 왜곡이 생길 수 있습니다 (예 : 이미지의 조명 변화로 인한 왜곡).

In a similar vein, Avidan [2001] used a Support Vector Machine (SVM) classifier for tracking. SVM is a general classification scheme that, given a set of positive and negative training examples, finds the best separating hyperplane between the two classes [1998]. During testing, the SVM gives a score to the test data indicating the degree of membership of the test data to the positive class. For SVM-based trackers, the positive examples consist of the images of the object to be tracked, and the negative examples consist of all things that are not to be tracked. Generally, negative examples consist of background regions that could be confused with the object. Avidan’s tracking method, instead of minimizing the intensity difference of a template from the image regions, maximizes the SVM classification score over image regions in order to estimate the position of the object. One advantage of this approach is that knowledge about background objects (negative examples that are not to be tracked) is explicitly incorporated in the tracker.

비슷한 맥락에서 Avidan [2001]은 SVM (Support Vector Machine) 분류기를 사용하여 추적을 수행했습니다. SVM은 양성 및 음성 훈련 세트를 고려해 볼 때 두 클래스 간의 가장 좋은 분리 초평면을 발견하는 일반적인 분류 체계이다 [1998]. 테스트하는 동안 SVM은 테스트 데이터에 긍정 등급에 대한 테스트 데이터의 등급을 나타내는 점수를 제공합니다. SVM 기반 추적기의 경우 긍정적 인 예는 추적 할 개체의 이미지로 구성되며 부정적인 예제는 추적되지 않는 모든 것들로 구성됩니다. 일반적으로 음수 예제는 객체와 혼동 될 수있는 배경 영역으로 구성됩니다. 이미지 영역에서 템플리트의 강도 차이를 최소화하는 대신 Avidan의 추적 방법은 이미지 영역에 대한 SVM 분류 스코어를 최대화하여 객체의 위치를 ​​추정합니다. 이 접근법의 한 가지 장점은 배경 개체 (추적되지 않는 부정적인 예)에 대한 지식이 추적기에 명시 적으로 통합된다는 것입니다.

5.2.3. Discussion. The main goal of the trackers in this category is to estimate the object motion. With the region-based object representation, computed motion implicitly defines the object region as well as the object orientation in the next frame since, for each point of the object in the current frame, its location in the next frame can be determined using the estimated motion model. Depending on the context in which these trackers are being used, only one of these three properties might be more important. For instance, in the case of analyzing the object behavior based on the object trajectory, only the motion is adequate. However, to identify an object, the region it encompasses is also important. In order to evaluate the performance of the trackers in this category, one can define measures based on what is expected from the tracker. In the case when the tracker is expected to provide only object motion, the evaluation can be performed by computing a distance measure between the estimated and actual motion parameters. An example of a distance measure can be the angular distance,

토론. 이 카테고리에서 추적자의 주요 목표는 객체 모션을 추정하는 것입니다. 영역 기반 객체 표현을 사용하면, 계산 된 모션은 현재 프레임의 객체의 각 포인트에 대해 다음 프레임에서의 위치가 예측 된 값을 사용하여 결정될 수 있기 때문에 다음 프레임의 객체 방향뿐만 아니라 객체 영역도 암시 적으로 정의합니다 모션 모델. 이러한 추적기가 사용되는 컨텍스트에 따라 이러한 세 가지 속성 중 하나만 더 중요 할 수 있습니다. 예를 들어, 물체 궤적을 기반으로 물체 거동을 분석하는 경우, 동작만으로 충분합니다. 그러나 객체를 식별하기 위해 객체가 포함되는 영역도 중요합니다. 이 카테고리에서 트래커의 실적을 평가하기 위해 트래커에서 예상되는 것을 기반으로 측정을 정의 할 수 있습니다. 트래커가 물체 움직임만을 제공 할 것으로 예상되는 경우, 평가는 추정 된 모션 파라미터와 실제 모션 파라미터 사이의 거리 측정치를 계산함으로써 수행 될 수있다. 거리 측정의 예는 각도 거리,



between the motion vectors, A and B. For applications when the tracker is required to provide the correct object region in addition to its trajectory, the tracker’s performance can be evaluated by computing the precision and the recall measures. Both the precision and the recall measure are defined in terms of the intersection of the hypothesized and correct object region. In particular, precision is the ratio of the intersection to the hypothesized regions. The recall is the ratio of the intersection to the ground truth. A qualitative comparison of kernel trackers can be obtained based on

추적기가 궤적에 추가하여 올바른 객체 영역을 제공해야 할 때 응용 프로그램의 경우 추적기의 성능은 정밀도와 리콜 측정 값을 계산하여 평가할 수 있습니다. 정밀도와 리콜 측정은 가정 된 객체 영역과 올바른 객체 영역의 교차점으로 정의됩니다. 특히 정밀도는 가설 영역과 교차하는 비율입니다. 리콜은 교차점과 지상 진리의 비율입니다. 커널 추적기의 질적 비교는 다음을 기반으로 얻을 수 있습니다.

—tracking single or multiple objects,

—ability to handle occlusion,

—requirement of training,

—type of motion model, and

—requirement of a manual initialization.

- 하나 또는 여러 객체 추적,

- 교합을 다루는 능력,

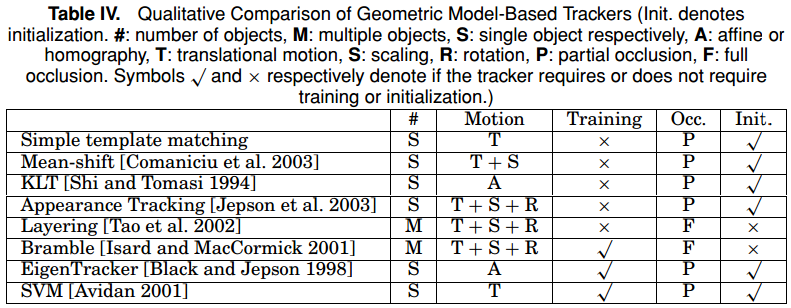
- 훈련의 요구,

- 유형의 모션 모델,

- 수동 초기화의 요구.

In Table IV, we provide the qualitative comparison of the methods discussed in this section.

표 4에서 이 절에서 논의 된 방법의 질적 비교를 제공합니다.



Use of primitive geometric shapes to represent objects is very common due to the realtime applicability of the state-of-the-art methods. Because of the rigidity constraint, tracking methods in this category compute parametric motion of the object. This motion is usually in the form of translation, conformal affine, affine, or projective. Motion of the object can be estimated by maximizing the object appearance similarity between the previous and current frame. The estimation process can be in the form of a brute force search, or by using gradient ascent (descent)-based maximization (minimization) process. Object trackers, based on the gradient ascent (descent) approach, require that some part of the object is at least visible inside the chosen shape whose location is defined by the previous object position. To eliminate such requirements, a possible approach is to use Kalman filtering or particle filtering discussed in the context of point trackers to predict the location of the object in the next frame. Given the object state defined in terms of velocity and acceleration of the object centroid these filters will estimate the position of the object centroid such that the likelihood of observing part of the object inside the kernel is increased [Comaniciu et al. 2003]. This requirement can also be met by performing global motion compensation, assuming that the objects are farther from the camera and camera motion can be estimated by affine or projective transformation [Yilmaz et al. 2003].

객체를 표현하기위한 원시 기하학적 모양의 사용은 최첨단 방법의 실시간 적용 가능성 때문에 매우 일반적입니다. 강성 제약으로 인해이 카테고리의 추적 방법은 객체의 파라 메트릭 모션을 계산합니다. 이 동작은 대개 번역, 등각 사인, 아핀 또는 투영의 형태로 이루어집니다. 객체의 움직임은 이전 프레임과 현재 프레임 사이의 객체 출현 유사성을 최대화함으로써 추정 될 수있다. 추정 프로세스는 무차별 대항력 탐색 (brute force search) 또는 그래디언트 상승 (하강) 기반 최대화 (최소화) 프로세스의 형태 일 수 있습니다. 그래디언트 상승 (하강) 접근법을 기반으로 한 객체 추적기는 객체의 일부가 이전 객체 위치에 의해 정의 된 위치를 선택한 모양 내부에서 적어도 볼 수 있어야합니다. 이러한 요구 사항을 제거하기 위해 가능한 방법은 포인트 추적기에서 설명한 칼만 필터링 또는 입자 필터링을 사용하여 다음 프레임에서 객체의 위치를 ​​예측하는 것입니다. 객체 중력의 속도와 가속도로 정의 된 객체 상태가 주어지면 이러한 필터는 객체 중심의 위치를 ​​추정하여 커널 내부의 객체 일부를 관찰 할 가능성이 증가합니다 [Comaniciu et al. 2003]. 이 요구 조건은 물체가 카메라로부터 멀어지고, 카메라 움직임이 아핀 또는 투영 변환에 의해 추정 될 수 있다고 가정하여 전역 움직임 보상을 수행함으로써 충족 될 수있다 [Yilmaz et al. 2003].

One of the limitations of primitive geometric shapes for object representation is that parts of the objects may be left outside of the defined shape while parts of the background may reside inside it. This phenomena can be observed for both the rigid objects (when the object pose changes) and nonrigid objects (when local motion results in changes in object appearance). In such cases, the object motion estimated by maximizing model similarity may not be correct. To overcome this limitation, one approach is to force the kernel to reside inside the object rather than encapsulating the complete shape. Another approach is to model the object appearance by probability density functions of color/texture and assign weights to the pixels residing inside the primitive shape based on the conditional probability of observed color/texture.

오브젝트 표현의 기본 기하학적 모양의 한계 중 하나는 오브젝트의 일부가 정의 된 모양의 바깥에 남아있을 수 있고 백그라운드의 일부가 내부에있을 수 있다는 것입니다. 이 현상은 강체 (포즈가 바뀔 때)와 강건하지 않은 (강체의 모양이 바뀌면) 관찰 될 수 있습니다. 이러한 경우 모델 유사성을 최대화하여 추정 한 물체 움직임이 올바르지 않을 수 있습니다. 이 제한을 극복하기위한 한 가지 방법은 완전한 모양을 캡슐화하는 대신 커널을 오브젝트 내부에 놓는 것입니다. 다른 접근법은 컬러 / 텍스쳐의 확률 밀도 함수에 의해 객체 외관을 모델링하고 관찰 된 컬러 / 텍스처의 조건부 확률에 기초하여 프리미티브 형상 내에 존재하는 픽셀에 가중치를 할당하는 것이다.

**5.3. Silhouette Tracking**

**실루엣 추적**

Objects may have complex shapes, for example, hands, head, and shoulders (see Figure 15(a)) that cannot be well described by simple geometric shapes. Silhouettebased methods provide an accurate shape description for these objects. The goal of a silhouette-based object tracker is to find the object region in each frame by means of an object model generated using the previous frames. This model can be in the form of a color histogram, object edges or the object contour. We divide silhouette trackers into two categories, namely, shape matching and contour tracking. Shape matching approaches search for the object silhouette in the current frame. Contour tracking approaches, on the other hand, evolve an initial contour to its new position in the current frame by either using the state space models or direct minimization of some energy functional.

객체는 복잡한 기하학적 모양을 가질 수 있습니다 (예 : 손, 머리, 어깨) (그림 15 (a) 참조). 실루엣 기반 메소드는 이러한 오브젝트에 대한 정확한 모양 설명을 제공합니다. 실루엣 기반 객체 추적기의 목표는 이전 프레임을 사용하여 생성 된 객체 모델을 사용하여 각 프레임에서 객체 영역을 찾는 것입니다. 이 모델은 색상 막대 그래프, 객체 모서리 또는 객체 윤곽의 형태 일 수 있습니다. 우리는 실루엣 추적기를 모양 매칭과 윤곽 추적이라는 두 가지 범주로 나눕니다. 모양 매칭 접근법은 현재 프레임에서 객체 실루엣을 검색합니다. 반면에 등고선 추적 접근법은 상태 공간 모델을 사용하거나 일부 에너지 기능을 직접 최소화하여 현재 윤곽선에서 새 윤곽선을 초기 윤곽선으로 전개합니다.

|  |
| --- |
|  |
| Fig. 15. (a) Edge observations along the contour normals (c 1998 Kluwer). (b) Level set contour representation, each grid position encodes the Euclidean distance between a grid point and the point on the contour; gray levels represent the values of the grid.  등고선 법선을 따르는 에지 관측 (C 1998 Kluwer). (b) 레벨 집합 윤곽 표현, 각 그리드 위치는 그리드 점과 윤곽선상의 점 사이의 유클리드 거리를 인코딩합니다. 회색 레벨은 그리드의 값을 나타냅니다. |

5.3.1. Shape Matching. Shape matching can be performed similar to tracking based on template matching (Section 5.2), where an object silhouette and its associated model is searched in the current frame. The search is performed by computing the similarity of the object with the model generated from the hypothesized object silhouette based on previous frame. In this approach, the silhouette is assumed to only translate from the current frame to the next, therefore nonrigid object motion is not explicitly handled. The object model, which is usually in the form of an edge map, is reinitialized to handle appearance changes in every frame after the object is located. This update is required to overcome tracking problems related to viewpoint and lighting condition changes as well as nonrigid object motion. In 1993, Huttenlocher et al. performed shape matching using an edge-based representation. The authors use the Hausdorff distance to construct a correlation surface from which the minimum is selected as the new object position. The Hausdorff metric is a mathematical measure for comparing two sets of points A = {a1, a2, ··· , an} and B = {b1, b2, ··· , bm} in terms of the least similar members [Hausdorff 1962]:

모양 매칭. 모양 매칭은 템플릿 매칭 (5.2 절)에 기반한 추적과 유사하게 수행 될 수 있는데, 여기서 객체 실루엣과 관련 모델은 현재 프레임에서 검색됩니다. 탐색은 이전 프레임을 기반으로 가설 된 객체 실루엣으로부터 생성 된 모델과 객체의 유사성을 계산함으로써 수행된다. 이 접근 방식에서 실루엣은 현재 프레임에서 다음 프레임으로 변환되는 것으로 간주되므로 비 압축 오브젝트 모션이 명시 적으로 처리되지 않습니다. 일반적으로 에지 맵의 형태 인 객체 모델은 객체가 배치 된 후 모든 프레임에서 모양 변경을 처리하도록 다시 초기화됩니다. 이 업데이트는 견고성이 낮은 물체의 움직임뿐만 아니라 시점 및 조명 조건 변경과 관련된 추적 문제를 극복하기 위해 필요합니다. 1993 년 Huttenlocher et al. 가장자리 기반 표현을 사용하여 모양 일치를 수행했습니다. 저자는 Hausdorff 거리를 사용하여 최소값이 새로운 객체 위치로 선택되는 상관 표면을 생성합니다. Hausdorff 메트릭은 가장 유사한 요소가없는 점에서 두 세트의 점 A = {a1, a2, ..., an}과 B = {b1, b2, ..., bm}을 비교하기위한 수학적 측정법입니다 [Hausdorff 1962 ] :

H(A, B) = max{h(A, B), h(b, A)}, (12)

where is the norm of choice.

여기서 는 선택의 표준입니다.

In the context of matching using an edge-based model, Hausdorff distance measures the most mismatched edges. Due to this, the method emphasize parts of the edge map that are not drastically affected by object motion. For instance, in the case of a walking person, the head and the torso do not change their shape much, whereas the motion of the arms and legs will result in drastic shape change, such that removing the edges corresponding to arms and legs will improve the tracking performance. In a similar vein, Li et al. [2001] propose using the Hausdorff distance for verification of the trajectories and pose estimation problem. Tracking is achieved by evaluating the optical flow vector computed inside the hypothesized silhouette such that the average flow provides the new object position. For verification, an edge-based model is kept for each tracked object for all possible object poses. The authors then apply distance transform to the edges in the hypothesized object edges to speed up the computation of L2 norm in the Hausdorff distance computation.

에지 기반 모델을 사용하여 일치하는 맥락에서 Hausdorff 거리는 가장 일치하지 않는 가장자리를 측정합니다. 이로 인해,이 방법은 오브젝트 모션에 의해 크게 영향을받지 않는 에지 맵의 부분을 강조한다. 예를 들어 걷는 사람의 경우 머리와 몸통의 모양이 많이 변하지 않는 반면 팔과 다리의 움직임은 급격한 모양 변화로 이어 지므로 팔과 다리에 해당하는 가장자리를 제거하는 것이 향상됩니다 추적 성능. 비슷한 맥락에서, Li et al. [2001]는 궤도의 검증과 추정 문제의 포즈를 위해 Hausdorff distance를 제안한다. 트래킹은 가설 된 실루엣 내부에서 계산 된 옵티컬 플로우 벡터를 평가하여 평균 플로우가 새로운 오브젝트 위치를 제공하도록하여 이루어진다. 검증을 위해 가능한 모든 오브젝트 포즈에 대해 각 추적 된 오브젝트에 대해 에지 기반 모델이 유지됩니다. 저자들은 Hausdorff 거리 계산에서 L2 놈의 계산을 빠르게하기 위해 가정 된 객체 가장자리에서 가장자리로 거리 변환을 적용합니다.

Another approach to match shapes is to find corresponding silhouettes detected in two consecutive frames. Establishing silhouette correspondence, or in short silhouette matching, can be considered similar to point matching discussed in Section 5.1. However, the main difference between silhouette matching and point matching is the object representations and the object models used. In particular, silhouette matching uses the complete object region in, contrast to using points. In addition, silhouette matching makes use of an objects appearance features, whereas point matching uses only motion and position-based features. Silhouette detection is usually carried out by background subtraction (see Section 4.2 for discussion). Once the object silhouettes are extracted, matching is performed by computing some distance between the object models associated with each silhouette. Object models are usually in the form of density functions (color or edge histograms), silhouette boundary (closed or open object contour), object edges or a combination of these models. In 2004, Kang et al. used histograms of color and edges as the object models. In contrast to traditional histograms, they proposed generating histograms from concentric circles with various radii centered on a set of control points on a reference circle. The reference circle is chosen as the smallest circle encapsulating the object silhouette. Use of concentric circles implicitly encodes the spatial information which in regular histogram is only possible when the spatial (x, y) coordinates are included in the observation vector [Comaniciu and Meer 2002]. Resulting color and edge histograms are rotation, translation, and scale invariant, hence, provide the same matching score for objects transformed by conformal affine transform. The matching score can be computed using several distance measures including cross-correlation, Bhattacharya distance, and Kullback-Leibler divergence. Among these three measures, the authors conclude that the Bhattacharya distance and the Kullback-Leibler divergence perform similarly, and both peform better than the correlation-based measure. To match silhouettes in consecutive frames, Haritaoglu et al. [2000] model the object appearance by the edge information obtained inside the object silhouette. In particular, the edge model is used to refine the translation of the object using the constant velocity assumption. This refinement is carried out by performing binary correlation between the object edge in the consecutive frames.

오브젝트 실루엣이 추출되면 각 실루엣과 연관된 오브젝트 모델 사이의 거리를 계산하여 일치를 수행합니다. 객체 모델은 대개 밀도 함수 (색상 또는 가장자리 히스토그램), 실루엣 경계 (닫힌 객체 또는 열린 객체 등고선), 객체 모서리 또는 이러한 모델의 조합 형식입니다. 2004 년 Kang et al. 객체 모델로 색상 및 가장자리의 히스토그램을 사용했습니다. 전통적인 히스토그램과 달리, 그들은 참조 원상의 제어점 세트를 중심으로 다양한 반지름으로 동심원에서 히스토그램을 생성 할 것을 제안했습니다. 기준 원은 물체 실루엣을 캡슐화하는 가장 작은 원으로 선택됩니다. 동심원을 사용하면 공간 벡터 (x, y)가 관찰 벡터에 포함될 때만 정규 히스토그램에서 가능한 공간 정보가 암시 적으로 인코딩됩니다 [Comaniciu and Meer 2002]. 결과 컬러 및 에지 히스토그램은 회전, 평행 이동 및 스케일 불변이므로 컨 포멀 아핀 변환에 의해 변형 된 객체에 대해 동일한 매칭 스코어를 제공합니다. 일치 점수는 상호 상관, Bhattacharya 거리 및 Kullback-Leibler 분기를 포함한 여러 거리 측정을 사용하여 계산할 수 있습니다. 이 세 가지 척도 중 Bhattacharya distance와 Kullback-Leibler divergence는 유사하게 수행되며, 둘 다 상관 관계 기반 척도보다 우수하다고 결론 지었다. 연속 프레임의 실루엣을 일치시키기 위해 Haritaoglu et al. [2000] 물체의 내부에서 얻어진 에지 정보로 물체의 모습을 모델링한다. 특히, 에지 모델은 일정한 속도 가정을 사용하여 물체의 이동을 다듬는 데 사용됩니다. 이 세밀화는 연속적인 프레임에서 객체 엣지간에 이진 상관을 수행함으로써 수행됩니다.

In contrast to looking for possible silhouette matches in consecutive frames, tracking silhouettes can be performed by computing the flow vectors for each pixel inside the silhouette such that the flow that is dominant over the entire silhouette is used to generate the silhouette trajectory. Following this observation, Sato and Aggarwal [2004] proposed to generating object tracks by applying Hough transform in the velocity space to the object silhouettes in consecutive frames. Binary object silhouettes are detected using background subtraction (see Section 4.2). Then, from a spatio-temporal window around each moving region pixel, a velocity Hough transform is applied to compute voting matrices for the vertical flow v and the horizontal flow u. These voting matrices provides the so-called Temporal Spatio-Velocity (TSV) image in 4D (x, y, u, v) per frame. TSV image encodes the dominant motion of a moving region pixel and its likelihood in terms of number of votes such that a thresholding operation will provide regions with similar motion patterns. In contrast to appearance-based matching of silhouettes, TSV provides a motion-based matching of the object silhouettes and is less sensitive to appearance variations, due to different object views (e.g., front and back of the object may look different).

연속적인 프레임에서 가능한 실루엣 매칭을 찾는 것과는 대조적으로, 전체 실루엣에 대해 지배적 인 흐름이 실루엣 궤도를 생성하는 데 사용되도록 실루엣 내부의 각 픽셀에 대한 흐름 벡터를 계산함으로써 추적 실루엣을 수행 할 수 있습니다. 이러한 관찰에 따라, Sato와 Aggarwal [2004]는 속도 공간에서 Hough 변환을 연속 프레임의 객체 실루엣에 적용하여 객체 트랙을 생성 할 것을 제안했습니다. 이진 객체 실루엣은 백그라운드 뺄셈을 사용하여 탐지됩니다 (4.2 절 참조). 그런 다음, 각 움직이는 영역 픽셀 주위의 시공간 창에서 수직 방향 흐름 v와 수평 방향 흐름 u에 대한 투표 행렬을 계산하기 위해 속도 Hough 변환이 적용됩니다. 이러한 투표 행렬은 프레임 당 4D (x, y, u, v)의 소위 Temporal Spatio-Velocity (TSV) 이미지를 제공합니다. TSV 이미지는 이동 영역 픽셀의 우세 모션과 임계 값 연산이 유사한 모션 패턴을 가진 영역을 제공 할 수있는 투표 수에 대한 우도를 인코딩합니다. 모양 기반의 실루엣 매칭과 달리 TSV는 객체 실루엣의 동작 기반 매칭을 제공하고 다른 객체 뷰 (예 : 객체의 앞면과 뒷면이 다르게 보일 수 있음)로 인해 외관 변화에 덜 민감합니다.

5.3.2. Contour Tracking. Contour tracking methods, in contrast to shape matching methods. iteratively evolve an initial contour in the previous frame to its new position in the current frame. This contour evolution requires that some part of the object in the current frame overlap with the object region in the previous frame. Tracking by evolving a contour can be performed using two different approaches. The first approach uses state space models to model the contour shape and motion. The second approach directly evolves the contour by minimizing the contour energy using direct minimization techniques such as gradient descent.

등고선 추적. 형상 매칭 방법과 대조되는 형상 추적 방법. 이전 프레임의 초기 윤곽을 현재 프레임의 새 위치로 반복적으로 전개합니다. 이 윤곽 진화는 현재 프레임의 객체 일부가 이전 프레임의 객체 영역과 중첩되어야합니다. 윤곽을 전개하여 추적하는 것은 두 가지 접근 방식을 사용하여 수행 할 수 있습니다. 첫 번째 방법은 상태 공간 모델을 사용하여 윤곽선 모양과 모션을 모델링합니다. 두 번째 방법은 그래디언트 디센트와 같은 직접 최소화 기법을 사용하여 윤곽 에너지를 최소화하여 윤곽을 직접 전개합니다.

5.3.2.1. Tracking Using State Space Models. The object’s state is defined in terms of the shape and the motion parameters of the contour. The state is updated at each time instant such that the contour’s a posteriori probability is maximized. The posterior probability depends on the prior state and the current likelihood which is usually de- fined in terms of the distance of the contour from observed edges. Terzopoulos and Szeliski [1992] define the object state by the dynamics of the control points. The dynamics of the control points are modeled in terms of a spring model, which moves the control points based on the spring stiffness parameters. The new state (spring parameters) of the contour is predicted using the Kalman filter. The correction step uses the image observations which are defined in terms of the image gradients. In 1998, Isard and Blake defined the object state in terms of spline shape parameters and affine motion parameters. The measurements consist of image edges computed in the normal direction to the contour (see Figure 15(a)). The state is updated using a particle filter. In order to obtain initial samples for the filter, they compute the state variables from the contours extracted in consecutive frames during a training phase. During the testing phase, the current state variables are estimated through particle filtering based on the edge observations along normal lines at the control points on the contour. In 2000, MacCormick and Blake extended the particle filter-based object tracker in Isard and Blake [1998] to track multiple objects by including the exclusion principle for handling occlusion. The exclusion principle integrates into the sampling step of the particle filtering framework such that, for two objects, if a feature lies in the observation space of both objects, then it contributes more to the samples of the object which is occluding the other object. Since the exclusion principle is only defined between two objects, this approach can track at most two objects undergoing occlusion at any time instant.

상태 공간 모델을 사용한 추적. 객체의 상태는 윤곽선의 모양과 동작 매개 변수로 정의됩니다. 상태는 매 순간마다 업데이트되어 등고선의 사후 확률이 최대가됩니다. 사후 확률은 이전 상태와 관찰 된 모서리에서 등고선까지의 거리로 정의되는 현재 우도에 따라 달라집니다. Terzopoulos와 Szeliski [1992]는 제어점의 역학에 의해 객체 상태를 정의합니다. 제어점의 동역학은 스프링 강성 매개 변수를 기반으로 제어점을 이동시키는 스프링 모델로 모델링됩니다. 윤곽의 새로운 상태 (스프링 파라미터)는 칼만 필터를 사용하여 예측됩니다. 보정 단계에서는 이미지 구배에 따라 정의 된 이미지 관측치를 사용합니다. 1998 년 Isard와 Blake는 스플라인 형상 파라미터와 아핀 모션 파라미터의 관점에서 객체 상태를 정의했습니다. 측정치는 윤곽선의 법선 방향으로 계산 된 이미지 가장자리로 구성됩니다 (그림 15 (a) 참조). 상태는 입자 필터를 사용하여 업데이트됩니다. 필터에 대한 초기 샘플을 얻으려면 트레이닝 단계에서 연속 프레임에서 추출한 윤곽선의 상태 변수를 계산합니다. 테스트 단계에서 현재 상태 변수는 윤곽의 제어점에서 정상선을 따라 모서리 관측을 기반으로 입자 필터링을 통해 추정됩니다. 2000 년 MacCormick과 Blake는 Isard와 Blake [1998]에서 입자 필터 기반 객체 추적기를 확장하여 폐색 처리를위한 제외 원칙을 포함하여 여러 객체를 추적했습니다. 제외 원리는 입자 필터링 프레임 워크의 샘플링 단계로 통합되어 두 대상의 경우 두 대상의 관찰 공간에 특징이 있으면 다른 대상을 가리고있는 대상의 표본에 더 많이 기여합니다. 배제 원칙은 두 객체간에 만 정의되기 때문에이 접근법은 언제든지 두 객체가 교합을 겪을 때 추적 할 수 있습니다.

Chen et al. [2001] propose a contour tracker where the contour is parameterized as an ellipse. Each contour node has an associated HMM and the states of each HMM is defined by the points lying on the lines normal to the contour control point. The observation likelihood of the contour depends on the background and the foreground partitions defined by the edge along the normal line on contour control points. The state transition probabilities of the HMM are estimated using the JPDAF. Given the observation likelihood and the state transition probabilities, the current contour state is estimated using the Viterbi algorithm [1967]. After the contour is approximated, an ellipse is fit to enforce elliptical shape constraint.

Chen et al. [2001]는 윤곽선이 타원으로 매개 변수화 된 윤곽 추적기를 제안합니다. 각 윤곽 노드에는 연관된 HMM이 있고 각 HMM의 상태는 윤곽 제어점에 수직 인 선에있는 점으로 정의됩니다. 윤곽선의 관찰 가능성은 윤곽선 제어점의 법선을 따라 가장자리로 정의 된 배경 및 전경 분할에 따라 다릅니다. HMM의 상태 전이 확률은 JPDAF를 사용하여 추정됩니다. 관측 우도와 상태 전이 확률이 주어지면, 현재 윤곽 상태는 비터 비 알고리즘 [1967]을 사용하여 추정된다. 윤곽선을 근사화 한 후에 타원형을 적용하여 타원형 모양을 적용합니다.

The methods just discussed above represent the contours using explicit representation, for example, parametric spline. Explicit representations do not allow topology changes such as region split or merge [Sethian 1999]. Next, we will discuss contour tracking methods based on direct minimization of energy functional. These methods can use implicit representations and allow topology changes.

위에서 설명한 방법은 매개 변수 스플라인과 같이 명시 적 표현을 사용하여 등고선을 나타냅니다. 명시 적 표현은 영역 분할 또는 병합과 같은 토폴로지 변경을 허용하지 않습니다 [Sethian 1999]. 다음으로, 우리는 에너지 기능을 직접적으로 최소화하여 윤곽 추적 방법을 논의 할 것이다. 이 메소드는 암시 적 표현을 사용하고 토폴로지 변경을 허용 할 수 있습니다.

5.3.2.2. Tracking by Direct Minimization of Contour Energy Functional. In the context of contour evolution, there is an analogy between the segmentation methods discussed in Section 4.3 and the contour tracking methods in this category. Both the segmentation and tracking methods minimize the energy functional either by greedy methods or by gradient descent. The contour energy is defined in terms of temporal information in the form of either the temporal gradient (optical flow) [Bertalmio et al. 2000; Mansouri 2002; Cremers and Schnorr 2003], or appearance statistics generated from the object and the background regions [Yilmaz et al. 2004; Ronfard 1994].

형상 에너지 직접 최소화로 추적. 윤곽 진화의 맥락에서, 4.3 절에서 논의 된 분할 방법과이 카테고리의 윤곽 추적 방법 사이의 유추가있다. 세분화 및 추적 방법은 탐욕적인 방법 또는 그래디언트 강하로 에너지 기능을 최소화합니다. 윤곽 에너지는 시간 구배 (광학 흐름) [Bertalmio et al. 2000; Mansouri 2002; Cremers and Schnorr 2003], 또는 물체와 배경 영역으로부터 생성 된 모양 통계 [Yilmaz et al. 2004; Ronfard 1994].



Contour tracking using temporal image gradients is motivated by the extensive work on computing the optical flow. The optical flow constraint is derived from the brightness constancy constraint: 

where I is the image, t is the time, and (u, v) is the flow vector in the x and the y directions.

시간적 이미지 그라디언트를 사용하는 윤곽 추적은 광 흐름을 계산하는 광범위한 작업에 의해 동기 부여됩니다. 옵티컬 플로우 제약은 I가 이미지이고, t가 시간이며, (u, v)가 x 및 y 방향의 플로우 벡터 인 밝기 상수 제약 조건 에서 파생됩니다.

Bertalmio et al. [2000] use this constraint to evolve the contour in consecutive frames. Their objective was to compute u and v iteratively for each contour position using the level set representation (see Figure 15(b)). At each iteration, contour speed in the normal direction, , is computed by projecting the gradient magnitude on .

Bertalmio et al. [2000]이 제약 조건을 사용하여 연속 프레임에서 윤곽을 전개합니다. 그들의 목적은 레벨 세트 표현을 사용하여 각 컨투어 위치에 대해 u와 v를 반복적으로 계산하는 것이 었습니다 (그림 15 (b) 참조). 각 반복에서 법선 방향의 윤곽선 속도 은 그래디언트 크기 를 에 투영하여 계산합니다.

The authors use two energy functionals, one for contour tracking, Et, and another one for intensity morphing,  where  is computed based on .

저자는 윤곽 추적을 위해 하나, 강도 모핑을 위해 또 다른 하나 인 을 사용하여 두 개의 에너지 기능을 사용합니다. 여기서 는 을 기반으로 계산됩니다.

The intensity morphing functional, which minimizes intensity changes in the current and the previous frames,

현재 및 이전 프레임에서의 강도 변화를 최소화하는 강도 모핑 기능은,

, on the hypothesized object contour,

 is coupled with the contour tracking equation, and both functionals are minimized simultaneously. For instance,  then the contour moves with the maximum speed in its normal direction, and 

is morphed into  On the other hand, if , then the evolution speed will be zero.

 , 가설 된 객체 윤곽선에서 는 윤곽 추적 공식 와 결합되며 두 기능은 동시에 최소화됩니다. 예를 들어 이면 윤곽선은 최대 속도로 법선 방향으로 이동하고 는 로 변형됩니다.

Similarly, Mansouri [2002] also uses the optical flow constraint for contour tracking. In contrast to Bertalmio et al. [2000] which computes the flow only on the object boundary, his approach is motivated by computing the flow vector for each pixel inside the complete object region in a circular neighborhood with radius r using a brute force search. Once the flow vectors are computed, the contour energy, which is based on the brightness constancy constraint, is evaluated. This process is iteratively performed until the energy is minimized. In Figure 16, we show the results of the tracking method proposed by Mansouri [2002] in a car sequence.

비슷하게, Mansouri [2002]는 윤곽 추적을 위해 옵티컬 플로우 제약 조건을 사용합니다. Bertalmio et al. [2000]은 객체 경계상에서 만 흐름을 계산하기 때문에 무차별 대항 탐색을 사용하여 반경 r을 갖는 원형 이웃에서 완전한 객체 영역 내부의 각 픽셀에 대한 유동 벡터를 계산함으로써 동기를 얻는다. 일단 유동 벡터가 계산되면, 밝기 불변 제약에 기초한 윤곽 에너지가 평가됩니다. 이 과정은 에너지가 최소화 될 때까지 반복적으로 수행됩니다. 그림 16에서 Mansouri [2002]가 제안한 추적 방법의 결과를 차량 순서대로 보여줍니다.

In 2003, Cremers and Schnorr also used the optical flow for contour evolution, and constraint such that an object can only have homogeneous flow vectors inside the region. Their energy is a modified form of the common Mumford-Shah energy [Mumford and Shah 1989], which evolves the contour until a region with homogeneous flow vectors is achieved. They also incorporated the shape priors to better estimate the object shape. The shape priors are generated from a set of object contours such that each control point on the contour has an associated Gaussian with a mean and standard deviation of the spatial positions of the corresponding control points on all the contours.

2003 년 Cremers와 Schnorr는 윤곽선의 진화를 위해 옵티컬 플로우를 사용했으며 물체가 동심원의 흐름 벡터 만 가질 수 있도록 제약을가했습니다. 그들의 에너지는 일반적인 Mumford-Shah 에너지의 변형 된 형태이다 [Mumford and Shah 1989]. 균질 한 유동 벡터를 가진 영역이 달성 될 때까지 윤곽을 전개한다. 그들은 또한 개체 모양을 더 잘 예측하기 위해 모양 사전을 통합했습니다. 형상 사전은 윤곽선의 각 제어점이 모든 윤곽선에서 해당 제어점의 공간 위치의 평균 및 표준 편차를 갖는 관련 가우스를 갖도록 객체 윤곽선 세트에서 생성됩니다

An alternative to using the optical flow is to exploit the consistency of the statistics computed inside and outside the object region from one frame to the next. This approach requires initialization of the contour in the current frame with its previous position. In this context, Ronfrad [1994] defines the energy functional governing the contour evolution based on the piecewise stationary image models formulated as Ward distances. Ward distance can be considered as a measure of image contrast [Beaulieu and Goldberg 1989]. However, Ward distance can not be analitycally defined, hence, Ronfard’s approach individually evolves each contour point based on its local neighborhood. In a similar vein, Yilmaz and Shah [2004] evolve an object contour using the color and texture models generated in a band around the object’s boundary (see Figure 17(a)). The width of the band serves as a means to combine region and boundary-based contour tracking methods into a single framework. In contrast to the aforementioned methods, Yilmaz et al. [2004] model the object shape and its changes by means of a level setbased shape model. In this model, the grid points of the level set hold the means and the standard deviations of the distances of points from the object boundary. The level set-based shape model resolves the object occlusions during the course of tracking (see Figure 17(b)).

옵티컬 플로우를 사용하는 대신 하나의 프레임에서 다음 프레임으로 오브젝트 영역 내부 및 외부에서 계산 된 통계의 일관성을 활용할 수 있습니다. 이 방법은 이전 위치의 현재 프레임에서 윤곽선을 초기화해야합니다. 이 문맥에서 Ronfrad [1994]는 와드 거리로 공식화 된 조각 별 정지 이미지 모델을 기반으로 윤곽 진화를 관리하는 에너지 기능을 정의합니다. 와드 거리는 이미지 대조의 척도로 간주 될 수있다 [Beaulieu and Goldberg 1989]. 그러나 와드 거리는 윤곽 적으로 정의 될 수 없으므로 론 페이드의 접근법은 지역 주변을 기반으로 각 윤곽선 점을 개별적으로 전개합니다. 비슷한 맥락에서, Yilmaz와 Shah [2004]는 물체의 경계선 주위의 밴드에서 생성 된 색상 및 텍스처 모델을 사용하여 물체 윤곽을 전개합니다 (그림 17 (a) 참조). 밴드의 너비는 영역 및 경계 기반 윤곽 추적 방법을 단일 프레임 워크로 결합하는 수단으로 사용됩니다. 앞서 언급 한 방법과 달리, Yilmaz et al. [2004] 레벨 집합 기반의 형상 모델을 이용하여 객체 형상과 그 변화를 모델링한다. 이 모델에서 레벨 집합의 격자 점은 객체 경계로부터의 점 거리의 평균 및 표준 편차를 유지합니다. 레벨 집합 기반 쉐이프 모델은 추적 과정에서 객체 오 클루 전을 해결합니다 (그림 17 (b) 참조).

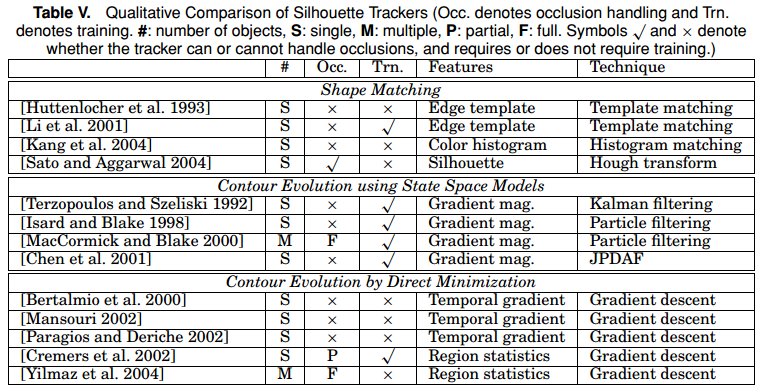
|  |
| --- |
|  |
| Fig. 17. Contour tracking results. (a) tracking of a tennis player, (b) tracking in presence of occlusion, using the method proposed by Yilmaz and Shah [2004] (c 2004 IEEE).  윤곽 추적 결과. (a) 테니스 선수 추적, (b) Yilmaz and Shah [2004] (c 2004 IEEE)에서 제안한 방법을 사용하여 폐색이있는 경우 추적. |

5.3.3. Discussion. Silhouette tracking is employed when tracking of the complete region of an object is required. In the context of region tracking, the precision and recall measures are defined in terms of the intersection of the hypothesized and correct object regions. The precision is the ratio of the intersection to the hypothesized region and recall is the ratio of the intersection to the ground truth. Important factors to distinguish different silhouette trackers are: What features are used? How is occlusion handled? Is training required or not? Moreover some algorithms only use information about the silhouette boundary for tracking, while others use the complete region inside the silhouette. Generally the region-based approaches are more resilient to noise. A qualitative comparison of contour-based silhouette tracking approaches is given in Table V.

토론. 실루엣 추적은 객체의 전체 영역을 추적해야 할 때 사용됩니다. 영역 추적의 맥락에서, 정확도 및 회상 측정치는 가정 된 객체 영역과 올바른 객체 영역의 교차점으로 정의됩니다. 정확도는 가설 영역과 교차하는 비율이며 회상 (recall)은 교차점과 지상 진리의 비율입니다. 다른 실루엣 추적기를 구분하는 중요한 요소는 다음과 같습니다. 어떤 기능이 사용됩니까? 교합은 어떻게 처리됩니까? 훈련이 필요한가 아닌가? 또한 일부 알고리즘은 추적을 위해 실루엣 경계에 대한 정보 만 사용하고 다른 알고리즘은 실루엣 내부의 전체 영역을 사용합니다. 일반적으로 지역 기반 접근법은 소음에보다 탄력적입니다. 윤곽 기반 실루엣 추적 접근법의 정 성적 비교는 표 V에 나와 있습니다.

The most important advantage of tracking silhouettes is their flexibility to handle a large variety of object shapes. Silhouettes can be represented in different ways. The most common silhouette representation is in the form of a binary indicator function, which marks the object region by ones and the nonobject regions by zeros. For contour-based methods, the silhouette is represented either explicitly or implicitly (see Figure 15). Explicit representation defines the boundary of the silhouette by a set of control points. Implicit representation defines the silhouette by means of a function defined on a grid. The most common implicit contour representation is the level sets representation.

실루엣 추적의 가장 중요한 장점은 다양한 객체 모양을 처리 할 수있는 유연성입니다. 실루엣은 다양한 방식으로 표현 될 수 있습니다. 가장 일반적인 실루엣 표현은 오브젝트 영역을 1로, 오브젝트 외 영역을 0으로 표시하는 2 진 표시기 기능의 형태입니다. 윤곽 기반 방법의 경우 실루엣이 명시 적으로 또는 암시 적으로 표현됩니다 (그림 15 참조). 명시 적 표현은 실루엣의 경계를 일련의 제어점으로 정의합니다. 암시 적 표현은 격자에 정의 된 함수를 사용하여 실루엣을 정의합니다. 가장 일반적인 암시 적 컨투어 표현은 레벨 세트 표현입니다.



The representations chosen by the silhouette-based object trackers can be in the form of motion models (similar to point trackers), appearance models (similar to kernel trackers), or shape models or a combination of these. Object appearance is usually modeled by parametric or nonparametric density functions such as mixture of Gaussians or histograms. Object shape can be modeled in the form of contour subspace where a subspace is generated from a set of possible object contours obtained from different object poses [Blake and Isard 2000]. Additionally, object shape can be implicitly modeled via a level set function where the grid positions are assigned at the distance generated from different level set functions corresponding to different object poses [Yilmaz et al. 2004]. Appearance-based shape representations are also commonly used by researchers who employ a brute force silhouette search. For edge-based shape representation, Hausdorff distance is the most widely used measure. However, Hausdorff measure is known for its sensitivity to noise. Hence, instead of using the maximum of distances, researchers have considered using an average of the distances [Baddeley 1992].

실루엣 기반 객체 추적기가 선택한 표현은 동작 모델 (포인트 추적기와 유사), 모양 모델 (커널 추적기와 유사) 또는 모양 모델 또는 이들의 조합 형태 일 수 있습니다. 오브젝트 모양은 일반적으로 가우시안 또는 히스토그램의 혼합과 같은 파라 메트릭 또는 비모수 적 밀도 함수로 모델링됩니다. 객체 모양은 다른 객체 포즈에서 얻은 가능한 객체 윤곽선 집합에서 부분 공간을 생성하는 등고선 부분 공간 형태로 모델링 할 수 있습니다 [Blake and Isard 2000]. 또한 오브젝트 형상은 레벨 설정 함수를 통해 암시 적으로 모델링 할 수 있습니다. 여기서 격자 위치는 서로 다른 오브젝트 포즈에 해당하는 서로 다른 레벨 세트 함수에서 생성 된 거리에 지정됩니다 [Yilmaz et al. 2004]. 모양 기반 모양 표현은 또한 무차별 적 실루엣 검색을 사용하는 연구자가 일반적으로 사용합니다. 모서리 기반 모양 표현의 경우 Hausdorff 거리가 가장 널리 사용되는 방법입니다. 그러나 Hausdorff 측정은 잡음에 대한 민감도로 잘 알려져 있습니다. 따라서 거리의 최대 값을 사용하는 대신 연구자는 거리의 평균을 사용하는 것을 고려했습니다 [Baddeley 1992].

Occlusion handling is another important aspect of silhouette tracking methods. Usually methods do not address the occlusion problem explicitly. A common approach is to assume constant motion or constant acceleration where, during occlusion, the object silhouette from the previous frame is translated to its hypothetical new position. Few methods explicitly handle object occlusions by enforcing shape constraints [MacCormick and Blake 2000; Yilmaz et al. 2004].

폐색 처리는 실루엣 추적 방법의 또 다른 중요한 측면입니다. 보통 메서드는 오 클루 젼 문제를 명시 적으로 처리하지 않습니다. 일반적인 접근법은 일정한 움직임 또는 일정한 가속을 가정하는 것이다. 여기서, 가림 중에, 이전 프레임으로부터의 대상 실루엣이 가상의 새로운 위치로 변환된다. 일부 제약 조건을 적용하여 객체 차단을 명시 적으로 처리하는 메소드는 거의 없습니다 [MacCormick and Blake 2000; Yilmaz et al. 2004].

Another important issue related to silhouette trackers is their capability for dealing with object split and merge. For instance, while tracking a silhouette of a person carrying an object, when the person leaves an object, a part of the person’s contour will be placed on the left object (region split). These topology changes of region split or merge can be handled well by implicit contour representations.

실루엣 추적기와 관련된 또 다른 중요한 문제는 객체 분할 및 병합을 처리하는 능력입니다. 예를 들어, 물건을 들고있는 사람의 실루엣을 추적하는 동안, 사람이 물건을 떠날 때, 사람의 등고선의 일부가 왼쪽 물건 (지역 분할)에 놓이게됩니다. 영역 분할 또는 병합의 이러한 토폴로지 변경은 암시 적 컨투어 표현으로 잘 처리 할 수 ​​있습니다.

**6. RELATED ISSUES**

In this section, we discuss issues that arise in tracking objects in realistic scenarios. These include locating objects as they undergo occlusion and keeping unique tracks of objects as they are viewed through multiple cameras.

이 섹션에서는 현실적인 시나리오에서 개체를 추적 할 때 발생하는 문제에 대해 설명합니다. 여기에는 여러 카메라를 통해 보았을 때 오브젝트의 오 클루 전을 추적하고 고유 한 트랙을 유지하면서 오브젝트를 찾는 작업이 포함됩니다.

**6.1. Resolving Occlusion**

Occlusion can be classified into three categories: self occlusion, interobject occlusion, and occlusion by the background scene structure. Self occlusion occurs when one part of the object occludes another. This situation most frequently arises while tracking articulated objects. Interobject occlusion occurs when two objects being tracked occlude each other. Similarly, occlusion by the background occurs when a structure in the background occludes the tracked objects. Generally, for interobject occlusion, the multiobject trackers like MacCormick and Blake [2000] and Elgammal et al. [2002] can exploit the knowledge of the position and the appearance of the occluder and occludee to detect and resolve occlusion. Partial occlusion of an object by a scene structure is hard to detect since it is difficult to differentiate between the object changing its shape and the object getting occluded.

폐색은 세 가지 범주로 나눌 수 있습니다 : 자기 폐색, 개체 간 폐색 및 배경 장면 구조에 의한 폐색. 자기 폐색은 객체의 한 부분이 다른 부분을 가릴 때 발생합니다. 이 상황은 관절이있는 객체를 추적하는 동안 가장 자주 발생합니다. 개체 간 교합은 추적중인 두 ​​개체가 서로를 가릴 때 발생합니다. 유사하게, 백그라운드의 구조가 추적 된 객체를 폐색 할 때 배경에 의한 폐색이 발생합니다. 일반적으로 interobject occlusion을 위해, MacCormick과 Blake [2000]와 Elgammal et al. [2002]는 교합의 위치와 모양에 대한 지식을 활용하여 교합을 탐지하고 해결할 수 있습니다. 장면 구조에 의한 물체의 부분적인 폐색은 물체가 그 모양을 바꾸는 것과 구속되는 물체를 구별하기 어렵 기 때문에 탐지하기 어렵다.

A common approach to handle complete occlusion during tracking is to model the object motion by linear dynamic models or by nonlinear dynamics and, in the case of occlusion, to keep on predicting the object location until the object reappears. For example, a linear velocity model is used in Beymer and Konolige [1999] and a Kalman filter is used for estimating the location and motion of objects. A nonlinear dynamic model is used in Isard and MacCormick [2001] and a particle filter employed for state estimation.

추적 중에 완전한 교합을 처리하는 일반적인 접근법은 선형 동적 모델 또는 비선형 동역학에 의한 객체 모션을 모델링하고, 교합의 경우 객체가 다시 나타날 때까지 객체 위치를 계속 예측하는 것입니다. 예를 들어, 선형 속도 모델은 Beymer and Konolige [1999]에서 사용되며 Kalman 필터는 물체의 위치와 움직임을 추정하는 데 사용됩니다. 비선형 동적 모델은 Isard와 MacCormick [2001]에서 사용되고 상태 추정에 사용되는 입자 필터가 사용된다.

Researchers have also utilized other features to resolve occlusion, for example, silhouette projections [Haritaoglu et al. 2000] (to locate persons’ heads during partial occlusion), and optical flow [Dockstader and Tekalp 2001b] (assuming that two objects move in opposite directions). Occlusion can also be implicitly resolved during generation of object tracks. In 2004, Sato and Aggarwal fit a slant cylindrical to the TSV (see Section 5.2 for more details) in the spatio-temporal space to recover from occlusions. In particular, two continuous slant cylinders, which are fit to the tracked object silhouette before the occlusion and after the occlusion, will intersect with each other for the frames whre the occlusion occurred. This intersection provides continuous trajectory before, during, and after the occlusion. Free-form object contour trackers employ a different occlusion resolution approach. These methods usually address occlusion by using shape priors which are either built ahead of time [Cremers et al. 2002] or built online [Yilmaz et al. 2004]. In particular, Cremers et al. [2002] build a shape model from subspace analysis (PCA) of possible object shapes to fill in missing contour parts. Yilmaz et al. [2004] build online shape priors using a mixture model based on the level set contour representation. Their approach is able to handle complete object occlusion.

연구원은 또한 실루엣 피처와 같은 폐색을 해결하기 위해 다른 기능을 활용했습니다 [Haritaoglu et al. 2000] (부분 폐색시 사람의 머리를 찾음) 및 옵티컬 플로우 [Dockstader and Tekalp 2001b] (두 객체가 반대 방향으로 움직인다 고 가정). Occlusion은 객체 트랙 생성 중에 암묵적으로 해결 될 수도 있습니다. 2004 년 Sato와 Aggarwal은 occlusions에서 회복하기 위해 시공간 공간에서 TSV에 원통형으로 기울어졌습니다 (자세한 내용은 5.2 절 참조). 특히, 오 클루 전 및 오 클루 전 후에 추적 된 객체 실루엣에 맞는 두 개의 연속 슬로프 실린더는 오 클루 전이 발생한 프레임에 대해 서로 교차합니다. 이 교집합은 교합 전, 도중 및 후에 연속적인 궤도를 제공합니다. 자유형 오브젝트 컨투어 트래커는 서로 다른 오 클루 전 해상도 방식을 사용합니다. 이러한 방법은 대개 미리 작성된 형태 사전을 사용하여 교합을 처리합니다 [Cremers et al. 2002] 온라인으로 만들거나 [Yilmaz et al. 2004]. 특히, Cremers et al. [2002] 가능한 객체 형상의 부분 공간 분석 (PCA)에서 형상 모델을 작성하여 누락 된 형상 부분을 채 웁니다. Yilmaz et al. [2004] 레벨 집합 윤곽 표현에 기반한 혼합 모델을 사용하여 온라인 형상 사전을 구축합니다. 그들의 접근 방식은 완전한 객체 오 클루 전을 처리 할 수 ​​있습니다.

The chance of occlusion can be reduced by an appropriate selection of camera positions. For instance, if the cameras are mounted on airborne vehicles, that is, when a birds-eye view of the scene is available, occlusions between objects on the ground do not occur. However, oblique view cameras are likely to encounter multiple object occlusions and require occlusion handling mechanisms. Multiple cameras viewing the same scene can also be used to resolve object occlusions during tracking [Dockstader and Tekalp 2001a; Mittal and Davis 2003]. These methods are discussed in the next section.

카메라 위치를 적절하게 선택하면 교합 기회를 줄일 수 있습니다. 예를 들어, 카메라가 공중 차량에 탑재 된 경우, 즉 장면의 조감도를 이용할 수있는 경우 지상의 물체 사이의 교합이 발생하지 않습니다. 그러나, 사시 뷰 카메라는 다수의 객체 폐색을 겪을 가능성이 있으며 폐색 처리 메커니즘을 필요로합니다. 동일한 장면을 보는 여러 카메라를 추적하는 동안 객체 폐색을 해결하는 데 사용할 수도 있습니다 [Dockstader and Tekalp 2001a; Mittal and Davis 2003]. 이러한 방법은 다음 절에서 설명합니다.

**6.2. Multiple Camera Tracking**

The need for using multiple cameras for tracking arises for two reasons. The first reason is the use of depth information for tracking and occlusion resolution. The second reason for using multiple cameras is to increase the area under view since it is not possible for a single camera to observe large areas because of a finite sensor field-of-view. An important issue in using multiple cameras is the relationship between the different camera views which can be manually defined [Collins et al. 2001; Cai and Aggarwal 1999] or computed automatically [Lee et al. 2000; Khan and Shah 2003] from the observations of the objects moving in the scene. For the tracking algorithms that require the depth estimation, high computational cost is another concern. However, due to the availability of successful commercial products, off-the-shelf real-time depth recovery systems are available which can be employed. In addition, methods like Mittal and Davis [2003] do not perform dense depth estimation but compute a sparse depth map (a single depth estimate for each object using a region-based stereo method) which also reduces the computational load. Multi-camera tracking methods like Dockstader and Tekalp [2001a] and Mittal and Davis [2003] have demonstrated superior tracking results as compared to single camera trackers in the case of persistent occlusion between the objects.

추적을 위해 여러 카메라를 사용해야하는 이유는 두 가지 이유에서입니다. 첫 번째 이유는 추적 및 폐색 해결에 깊이 정보를 사용하기 때문입니다. 여러 카메라를 사용하는 두 번째 이유는 유한 카메라 시야각 때문에 단일 카메라가 넓은 영역을 관찰 할 수 없으므로 시야 면적을 늘리는 것입니다. 여러 카메라를 사용할 때 중요한 문제는 수동으로 정의 할 수있는 여러 카메라 뷰 간의 관계입니다 [Collins et al. 2001; Cai and Aggarwal 1999] 또는 자동 계산 [Lee et al. 2000; Khan and Shah 2003]에서 그 장면에서 움직이는 물체들의 관찰로부터. 깊이 추정을 요구하는 추적 알고리즘의 경우, 높은 계산 비용이 또 다른 관심사이다. 그러나, 성공적인 상업용 제품의 가용성으로 인해, 상용 가능한 실시간 깊이 회수 시스템이 이용 가능하다. 또한 Mittal과 Davis [2003]와 같은 방법은 조밀 한 깊이 추정을 수행하지 않지만 계산 부하를 줄이는 스파 스 깊이 맵 (영역 기반 스테레오 방법을 사용하여 각 객체에 대한 단일 깊이 추정)을 계산합니다. Dockstader와 Tekalp [2001a], Mittal and Davis [2003]와 같은 다중 카메라 추적 방법은 객체 간의 지속적인 오 클루 전의 경우 단일 카메라 추적기와 비교할 때 우수한 추적 결과를 나타 냈습니다.

The aforementioned multi-camera tracking methods assume stationary cameras. Recently, Kang et al. [2003] used a combination of stationary and pan-tilt-zoom cameras with overlapping views for tracking. In many situations, it is not possible to have overlapping camera views due to limited resources or large areas of interest. Methods for tracking in such a scenario inherently have to deal with sparse object observations due to nonoverlapping views. Therefore some assumptions have to be made about the object speed and the path in order to obtain the correspondences across cameras [Huang and Russell 1997; Kettnaker and Zabih 1999; Javed et al. 2003]. Note that these methods, which establish object correspondence across nonoverlapping cameras, assume 1) the cameras are stationary and 2) the object tracks within each camera are available. The performance of these algorithms depends greatly on how much the objects follow the established paths and expected time intervals across cameras. For scenarios in which spatio-temporal constraints cannot be used, for example, objects moving arbitrarily in the nonoverlap region, the only tracking-by-recognition approach can be employed, which uses the appearance and the shape of the object to recognize it when it reappears in a camera view.

를 가정한다. 최근, Kang et al. [2003]은 추적을 위해보기가 중첩 된 고정 및 팬 틸트 - 줌 카메라의 조합을 사용했습니다. 대부분의 경우 제한된 리소스 또는 관심 영역으로 인해 카메라 뷰가 중복 될 수 없습니다. 이러한 시나리오에서 추적하는 방법은 본질적으로 겹치지 않는보기로 인한 희소물 관찰을 처리해야합니다. 그러므로 카메라 들간의 일치 성을 얻기 위해서는 물체 속도와 경로에 대한 몇 가지 가정이 필요하다 [Huang and Russell 1997; Kettnaker and Zabih 1999; Javed et al. 2003]. 비 중첩 카메라간에 객체 대응을 설정하는 이러한 방법은 1) 카메라가 고정되어 있고 2) 각 카메라 내의 객체 트랙을 사용할 수 있다고 가정합니다. 이러한 알고리즘의 성능은 오브젝트가 설정된 경로를 따르는 정도와 예상되는 카메라 간 시간 간격에 따라 크게 달라집니다. 시공간 제약을 사용할 수없는 시나리오, 예를 들어 비 중첩 영역에서 임의로 움직이는 물체는 물체의 모양과 모양을 사용하여 인식 할 수있는 유일한 추적 별 접근법을 사용할 수 있습니다 카메라보기에 다시 나타납니다.

**7. FUTURE DIRECTIONS**

Significant progress has been made in object tracking during the last few years. Several robust trackers have been developed which can track objects in real time in simple scenarios. However, it is clear from the papers reviewed in this survey that the assumptions used to make the tracking problem tractable, for example, smoothness of motion, minimal amount of occlusion, illumination constancy, high contrast with respect to background, etc., are violated in many realistic scenarios and therefore limit a tracker’s usefulness in applications like automated surveillance, human computer interaction, video retrieval, traffic monitoring, and vehicle navigation. Thus, tracking and associated problems of feature selection, object representation, dynamic shape, and motion estimation are very active areas of research and new solutions are continuously being proposed.

지난 몇 년 동안 물체 추적에 상당한 진전이있었습니다. 간단한 시나리오에서 실시간으로 객체를 추적 할 수있는 몇 가지 견고한 추적기가 개발되었습니다. 그러나이 설문 조사에서 검토 된 논문에서 추적 문제를 다루기 쉽도록 만든 가정, 예를 들어 움직임의 부드러움, 최소 교합 정도, 조명 지속성, 배경과의 대비 등을 위반 한 것은 분명합니다. 많은 현실적인 시나리오에서 자동 감시, 컴퓨터 상호 작용, 비디오 검색, 교통 모니터링 및 차량 탐색과 같은 응용 프로그램에서 추적기의 유용성을 제한합니다. 따라서, 특징 선택, 객체 표현, 동적 형태 및 모션 추정의 추적 및 관련 문제는 매우 활발한 연구 분야이며 새로운 솔루션이 지속적으로 제안되고있다.

One challenge in tracking is to develop algorithms for tracking objects in unconstrained videos, for example, videos obtained from broadcast news networks or home videos. These videos are noisy, compressed, unstructured, and typically contain edited clips acquired by moving cameras from multiple views. Another related video domain is of formal and informal meetings. These videos usually contain multiple people in a small field of view. Thus, there is severe occlusion, and people are only partially visible. One interesting solution in this context is to employ audio in addition to video for object tracking. There are some methods being developed for estimating the point of location of audio source, for example, a person’s mouth, based on four or six microphones. This audio-based localization of the speaker provides additional information which then can be used in conjunction with a video-based tracker to solve problems like severe occlusion.

추적의 한 가지 문제는 예를 들어 방송 뉴스 네트워크 또는 가정용 비디오에서 얻은 비디오와 같이 제한되지 않은 비디오에서 객체를 추적하기위한 알고리즘을 개발하는 것입니다. 이 비디오는 시끄럽고 압축되어 있고 체계가 없으며 일반적으로 여러보기에서 카메라를 이동하여 얻은 편집 된 클립이 포함되어 있습니다. 또 다른 관련 비디오 도메인은 공식 및 비공식 회의입니다. 이 동영상에는 일반적으로 작은 분야의 여러 인물이 포함됩니다. 따라서 심각한 교합이 있으며 사람들은 부분적으로 만 볼 수 있습니다. 이 컨텍스트에서 흥미로운 솔루션 중 하나는 객체 추적을 위해 비디오 외에도 오디오를 사용하는 것입니다. 4 개 또는 6 개의 마이크를 기반으로 한 오디오 소스의 위치 (예 : 사람의 입)를 예측하기위한 몇 가지 방법이 개발되고 있습니다. 이 오디오 기반의 스피커 현지화는 심한 교합과 같은 문제를 해결하기 위해 비디오 기반 추적기와 함께 사용할 수있는 추가 정보를 제공합니다.

In general, an important issue that has been neglected in the development of tracking algorithms is integration of contextual information. For example, in a vehicle tracking application, the location of vehicles should be constrained to paths on the ground as opposed to vertical walls or the sky. Recent work in the area of object recognition [Torralba 2003; Kumar and Hebert 2003] has shown that exploiting contextual information is helpful in recognition. In addition, advances in classifiers [Friedman et al. 2000; Tipping 2001] have made accurate detection of scene context possible, for example, man made structures, paths of movement, class of objects, etc. A tracker that takes advantage of contextual information to incorporate general constraints on the shape and motion of objects will usually perform better than one that does not exploit this information. This is because a tracker designed to give the best average performance in a variety of scenarios can be less accurate for a particular scene than a tracker that is attuned (by exploiting context) to the characteristics of that scene.

The use of a particular feature set for tracking can also greatly affect the performance.

일반적으로, 추적 알고리즘의 개발에서 무시 되어온 중요한 문제는 문맥 정보의 통합입니다. 예를 들어, 차량 추적 애플리케이션에서 차량의 위치는 수직 벽이나 하늘과 달리지면상의 경로로 제한되어야합니다. 물체 인식 분야의 최근 연구 [Torralba 2003; Kumar and Hebert 2003]는 문맥 정보를 악용하는 것이 인식에 도움이된다는 것을 보여 주었다. 또한 분류 자의 발전 [Friedman et al. 2000; Tipping 2001]은 사람이 만든 구조, 이동 경로, 객체의 클래스 등 가능한 장면의 컨텍스트를 정확하게 감지했습니다. 객체의 모양과 동작에 대한 일반적인 제약을 통합하기 위해 상황 정보를 이용하는 추적기는 대개 이 정보를 악용하지 않는 것보다 잘 수행하십시오. 이는 다양한 시나리오에서 최상의 평균 성능을 제공하도록 설계된 추적기가 해당 장면의 특성에 맞게 (상황을 악용하여) 조정되는 추적기보다 특정 장면에 대해 덜 정확할 수 있기 때문입니다.

추적을 위해 특정 기능 세트를 사용하면 성능에 큰 영향을 줄 수 있습니다.

Generally, the features that best discriminate between multiple objects and, between the object and background are also best for tracking the object. Many tracking algorithms use a weighted combination of multiple features assuming that a combination of preselected features will be discriminative. A wide range of feature selection algorithms have been investigated in the machine learning and pattern recognition communities. However, these algorithms require offline training information about the target and/or the background. Such information is not always available. Moreover, as the object appearance or background varies, the discriminative features also vary. Thus, there is a need for online selection of discriminative features. Some work has been done in this area for online selection of individual features [Collins and Liu 2003; Stern and Efros 2002]. However, the problem of efficient online estimation of discriminative feature sets remains unresolved. One promising direction to achieve this goal is the use of the online boosting methods [Oza 2002] for feature selection.

일반적으로 여러 개체와 개체와 배경을 가장 잘 구분하는 기능은 개체를 추적하는 데 가장 적합합니다. 많은 추적 알고리즘은 사전 선택된 피쳐들의 조합이 차별적 일 것이라는 가정하에 여러 피쳐의 가중치 조합을 사용합니다. 다양한 기능 선택 알고리즘이 기계 학습 및 패턴 인식 커뮤니티에서 조사되었습니다. 그러나 이러한 알고리즘은 대상 및 / 또는 배경에 대한 오프라인 교육 정보가 필요합니다. 그러한 정보는 항상 이용 가능한 것은 아닙니다. 더욱이, 물체의 외관 또는 배경이 변함에 따라, 판별 특징 또한 다양하다. 따라서, 차별적 인 특징을 온라인으로 선택할 필요가있다. 개별 기능의 온라인 선택을 위해이 영역에서 일부 작업이 수행되었습니다 [Collins and Liu 2003; 스턴 (Stern)과 에프 로스 (Efros) 2002]. 그러나 차별적 인 기능 집합의 효율적인 온라인 평가 문제는 해결되지 않았습니다. 이 목표를 달성하기위한 한 가지 유망한 방향은 기능 선택을위한 온라인 부스팅 방법 [Oza 2002]의 사용입니다.

In a similar vein, most tracking algorithms use prespecified models for object representation. The capability to learn object models online will greatly increase the applicability of a tracker. Motion-based segmentation [Vidal and Ma 2004; Black and Anandan 1996; Wang and Adelson 1994] and multibody factorization [Costeira and Kanade 1998; Gear 1998] methods have been used to learn models for multiple objects moving in a scene. However, these approaches assume rigid body motion. Unsupervised learning of object models for multiple nonrigid moving objects from a single camera remains an unsolved problem. One interesting direction that has largely been unexplored is the use of semisupervised learning techniques for modeling objects. These techniques (cotraining [Levin et al. 2003; Blum and Mitchell 1998], transductive SVMs [Joachims 1999], constrained graph cuts [Yu and Shi 2004]) do not require prohibitive amounts of training data. Moreover, they can not only learn nonrigid shapes and/or appearance, but they can also encode the knowledge of the background in the form of negative training data.

비슷한 맥락에서, 대부분의 추적 알고리즘은 객체 표현을 위해 미리 지정된 모델을 사용합니다. 객체 모델을 온라인으로 학습하는 기능은 추적기의 적용 가능성을 크게 높입니다. 모션 기반 분할 [Vidal and Ma 2004; 블랙 앤 아난드 1996; Wang and Adelson 1994] 및 다 물체 인자 분해 [Costeira and Kanade 1998; Gear 1998] 방법은 한 장면에서 움직이는 여러 물체에 대한 모델을 학습하는 데 사용되었습니다. 그러나 이러한 접근법은 강체 운동을 가정합니다. 단일 카메라에서 여러 개의 비 열성 움직이는 물체에 대한 물체 모델의 감독되지 않은 학습은 여전히 ​​해결되지 않은 문제입니다. 크게 탐구되지 않은 흥미로운 한 가지 방향은 객체 모델링을위한 반자동 학습 기술의 사용이다. 이러한 기술 (cotraining [Levin et al 2003, Blum and Mitchell 1998], transductive SVM [Joachims 1999], 제한된 그래프 절단 [Yu and Shi 2004])은 엄청난 양의 훈련 데이터를 필요로하지 않는다. 또한, 그들은 극한의 모양이나 외양을 배울 수있을뿐만 아니라 부정적인 훈련 데이터의 형태로 배경 지식을 인코딩 할 수도 있습니다.

Probabilistic state-space methods including Kalman Filters [Bar-Shalom and Foreman 1988], JPDAFs [Cox 1993], HMMs [Rabiner 1989], and Dynamic Bayesian Networks (DBNs) [Jensen 2001] have been extensively used to estimate object motion parameters. Among these methods, DBNs are probably the most general method for representation of conditional dependencies between multiple variables and/or image observations. They also provide a principled framework for fusing information from different sources. However, there is a need for more efficient solutions for inference before DBNs are more commonly used in tracking applications.

Kalman Filters [Bar-Shalom and Foreman 1988], JPDAFs [Cox 1993], HMMs [Rabiner 1989], Dynamic Bayesian Networks (DBNs) [Jensen 2001]를 포함한 확률 론적 상태 공간 기법이 객체 모션 파라미터를 추정하는 데 광범위하게 사용되었다. 이 중 DBN은 아마도 여러 변수 및 / 또는 이미지 관측 사이의 조건부 종속성을 표현하는 가장 일반적인 방법 일 것입니다. 또한 서로 다른 출처의 정보를 융합하는 원칙적인 프레임 워크를 제공합니다. 그러나 DBN이 추적 응용 프로그램에서보다 일반적으로 사용되기 전에 추론을위한보다 효율적인 솔루션이 필요합니다.

Overall, we believe that additional sources of information, in particular prior and contextual information, should be exploited whenever possible to attune the tracker to the particular scenario in which it is used. A principled approach to integrate these disparate sources of information will result in a general tracker that can be employed with success in a variety of applications.

전반적으로, 우리는 추가 정보 소스, 특히 선행 정보 및 문맥 정보가 가능할 때마다 트래커를 사용되는 특정 시나리오에 맞게 조정해야한다고 생각합니다. 이러한 서로 다른 정보 소스를 통합하는 원칙적 접근 방식은 다양한 애플리케이션에서 성공할 수있는 일반 추적기로 귀결됩니다.

**8. CONCLUDING REMARKS**

In this article, we present an extensive survey of object tracking methods and also give a brief review of related topics. We divide the tracking methods into three categories based on the use of object representations, namely, methods establishing point correspondence, methods using primitive geometric models, and methods using contour evolution. Note that all these classes require object detection at some point. For instance, the point trackers require detection in every frame, whereas geometric region or contours-based trackers require detection only when the object first appears in the scene. Recognizing the importance of object detection for tracking systems, we include a short discussion on popular object detection methods. We provide detailed summaries of object trackers, including discussion on the object representations, motion models, and the parameter estimation schemes employed by the tracking algorithms. Moreover, we describe the context of use, degree of applicability, evaluation criteria, and qualitative comparisons of the tracking algorithms. We believe that, this article, the first survey on object tracking with a rich bibliography content, can give valuable insight into this important research topic and encourage new research.

이 기사에서는 객체 추적 방법에 대한 광범위한 설문 조사를 제공하고 관련 주제에 대해 간략하게 검토합니다. 추적 방법을 객체 표현의 사용, 즉 점 대응 관계를 설정하는 방법, 원시 기하학 모델을 사용하는 방법 및 윤곽 진화를 사용하는 방법의 세 가지 범주로 나눕니다. 이러한 모든 클래스는 어떤 시점에서 객체 감지가 필요합니다. 예를 들어, 점 추적기는 모든 프레임에서 탐지가 필요하지만 기하학적 영역 또는 등고선 기반 추적기는 객체가 장면에 처음 나타날 때만 탐지가 필요합니다. 추적 시스템에 대한 물체 감지의 중요성을 인식하여 널리 사용되는 물체 감지 방법에 대한 짧은 토론이 포함됩니다. 우리는 객체 표현 자, 동작 모델 및 추적 알고리즘에 사용되는 매개 변수 추정 체계에 대한 토론을 포함하여 객체 추적자의 상세한 요약을 제공합니다. 또한, 우리는 사용의 맥락, 적용 정도, 평가 기준, 추적 알고리즘의 질적 비교를 기술한다. 풍부한 서지 정보가 포함 된 객체 추적에 대한 첫 번째 조사 인이 기사는이 중요한 연구 주제에 대한 중요한 통찰력을 제공하고 새로운 연구를 장려 할 수 있다고 생각합니다.

REFERENCES

AGGARWAL, J. K. AND CAI, Q. 1999. Human motion analysis: A review. Comput. Vision Image Understand. 73, 3, 428–440. ALI, A. AND AGGARWAL, J. 2001. Segmentation and recognition of continuous human activity. In IEEE Workshop on Detection and Recognition of Events in Video. 28–35. AVIDAN, S. 2001. Support vector tracking. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 184–191. BADDELEY, A. 1992. Errors in binary images and an l version of the haus- dorff metric. Nieuw Archief voor Wiskunde 10, 157–183. BALLARD, D. AND BROWN, C. 1982. Computer Vision. Prentice-Hall. BAR-SHALOM, Y. AND FOREMAN, T. 1988. Tracking and Data Association. Academic Press Inc. BARRON, J., FLEET, D., AND BEAUCHEMIN, S. 1994. Performance of optical flow techniques. Int. J. Comput. Vision 12, 43–77. BEAULIEU, J. AND GOLDBERG, M. 1989. Hierarchy in picture image segmentation: A step wise optimization approach. IEEE Trans. Patt. Analy. Mach. Intell. 11, 150–163. BERTALMIO, M., SAPIRO, G., AND RANDALL, G. 2000. Morphing active contours. IEEE Trans. Patt. Analy. Mach. Intell. 22, 7, 733–737. BEYMER, D. AND KONOLIGE, K. 1999. Real-time tracking of multiple people using continuous detection. In IEEE International Conference on Computer Vision (ICCV) Frame-Rate Workshop.. BIRCHFIELD, S. 1998. Elliptical head tracking using intensity gradients and color histograms. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 232–237. BLACK, M. AND ANANDAN, P. 1996. The robust estimation of multiple motions: Parametric and piecewisesmooth flow fields. Comput. Vision Image Understand. 63, 1, 75–104. BLACK, M. AND JEPSON, A. 1998. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. Int. J. Comput. Vision 26, 1, 63–84. BLAKE, A. AND ISARD, M. 2000. Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion. Springer. BLUM, A. AND MITCHELL, T. 1998. Combining labeled and unlabeled data with co-training. In 11th Annual Conference on Computational Learning Theory. 92–100. BLUM, A. L. AND LANGLEY, P. 1997. Selection of relevant features and examples in machine learning. Artific. Intell. 97, 1-2, 245–271. BOSER, B., GUYON, I. M., AND VAPNIK, V. 1992. A training algorithm for optimal margin classifiers. In ACM Workshop on Conference on Computational Learning Theory (COLT). 142–152.

BOWYER, K., KRANENBURG, C., AND DOUGHERTY, S. 2001. Edge detector evaluation using empirical roc curve. Comput. Vision Image Understand. 10, 77–103. BREGLER, C., HERTZMANN, A., AND BIERMANN, H. 2000. Recovering nonrigid 3d shape from image streams. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 690–696. BROIDA, T. AND CHELLAPPA, R. 1986. Estimation of object motion parameters from noisy images. IEEE Trans. Patt. Analy. Mach. Intell. 8, 1, 90–99. CAI, Q. AND AGGARWAL, J. 1999. Tracking human motion in structured environments using a distributed camera system. IEEE Trans. Patt. Analy. Mach. Intell. 2, 11, 1241–1247. CANNY, J. 1986. A computational approach to edge detection. IEEE Trans. Patt. Analy. Mach. Intell. 8, 6, 679–698. CASELLES, V., KIMMEL, R., AND SAPIRO, G. 1995. Geodesic active contours. In IEEE International Conference on Computer Vision (ICCV). 694–699. CHAM, T. AND REHG, J. M. 1999. A multiple hypothesis approach to figure tracking. In IEEE International Conference on Computer Vision and Pattern Recognition. 239–245. CHANG, Y. L. AND AGGARWAL, J. K. 1991. 3d structure reconstruction from an ego motion sequence using statistical estimation and detection theory. In Workshop on Visual Motion. 268–273. CHEN, Y., RUI, Y., AND HUANG, T. 2001. Jpdaf based hmm for real-time contour tracking. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 543–550. COLLINS, R., LIPTON, A., FUJIYOSHI, H., AND KANADE, T. 2001. Algorithms for cooperative multisensor surveillance. Proceedings of IEEE 89, 10, 1456–1477. COMANICIU, D. 2002. Bayesian kernel tracking. In Annual Conference of the German Society for Pattern Recognition. 438–445. COMANICIU, D. AND MEER, P. 1999. Mean shift analysis and applications. In IEEE International Conference on Computer Vision (ICCV). Vol. 2. 1197–1203. COMANICIU, D. AND MEER, P. 2002. Mean shift: A robust approach toward feature space analysis. IEEE Trans. Patt. Analy. Mach. Intell. 24, 5, 603–619. COMANICIU, D., RAMESH, V., AND MEER, P. 2003. Kernel-based object tracking. IEEE Trans. Patt. Analy. Mach. Intell. 25, 564–575. COOTES, T., EDWARDS, G., AND TAYLOR, C. 2001. Robust real-time periodic motion detection, analysis, and applications. IEEE Trans. Patt. Analy. Mach. Intell. 23, 6, 681–685. COSTEIRA, J. AND KANADE, T. 1998. A multibody factorization method for motion analysis. Int. J. Comput. Vision 29, 3, 159–180. COX, I. AND HINGORANI, S. 1996. An efficient implementation of reid’s multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. IEEE Trans. Patt. Analy. Mach. Intell. 18, 2, 138–150. COX, I. J. 1993. A review of statistical data association techniques for motion correspondence. Int. J. Comput. Vision 10, 1, 53–66. CREMERS, D., KOHLBERGER, T., AND SCHNORR, C. 2002. Non-linear shape statistics in mumford-shah based segmentation. In European Conference on Computer Vision (ECCV). CREMERS, D. AND SCHNORR, C. 2003. Statistical shape knowledge in variational motion segmentation. I. Srael Nent. Cap. J. 21, 77–86. DOCKSTADER, S. AND TEKALP, A. M. 2001a. Multiple camera tracking of interacting and occluded human motion. Proceedings of the IEEE 89, 1441–1455. DOCKSTADER, S. AND TEKALP, M. 2001b. On the tracking of articulated and occluded video object motion. Real Time Image 7, 5, 415–432. EDWARDS, G., TAYLOR, C., AND COOTES, T. 1998. Interpreting face images using active appearance models. In International Conference on Face and Gesture Recognition. 300–305. ELGAMMAL, A., DURAISWAMI, R., HARWOOD, D., AND DAVIS, L. 2002. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. Proceedings of IEEE 90, 7, 1151–1163. ELGAMMAL, A., HARWOOD, D., AND DAVIS, L. 2000. Non-parametric model for background subtraction. In European Conference on Computer Vision (ECCV). 751–767. FIEGUTH, P. AND TERZOPOULOS, D. 1997. Color-based tracking of heads and other mobile objects at video frame rates. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 21–27. FREUND, Y. AND SCHAPIRE, R. 1995. A decision-theoretic generalization of on-line learning and an application to boosting. Computat. Learn. Theory. 23–37. FRIEDMAN, J., HASTIE, T., AND TIBSHIRANI, R. 2000. Additive logistic regression: A statistical view of boosting. annals of statistics. Ann. Stat. 38, 2, 337–374.

GAO, X., BOULT, T., COETZEE, F., AND RAMESH, V. 2000. Error analysis of background adaption. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 503–510. GAVRILA, D. M. 1999. The visual analysis of human movement: A survey. Comput. Vision Image Understand. 73, 1, 82–98. GEAR, C. W. 1998. Multibody grouping from motion images. Int. J. Comput. Vision 29, 2, 133–150. GREENSPAN, H., BELONGIE, S., GOODMAN, R., PERONA, P., RAKSHIT, S., AND ANDERSON, C. 1994. Overcomplete steerable pyramid filters and rotation invariance. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 222–228. GREWE, L. AND KAK, A. 1995. Interactive learning of a multi-attribute hash table classifier for fast object recognition. Comput. Vision Image Understand. 61, 3, 387–416. HARALICK, R., SHANMUGAM, B., AND DINSTEIN, I. 1973. Textural features for image classification. IEEE Trans. Syst. Man Cybern. 33, 3, 610–622. HARITAOGLU, I., HARWOOD, D., AND DAVIS, L. 2000. W4: real-time surveillance of people and their activities. IEEE Trans. Patt. Analy. Mach. Intell. 22, 8, 809–830. HARRIS, C. AND STEPHENS, M. 1988. A combined corner and edge detector. In 4th Alvey Vision Conference. 147–151. HARRISSRC. Harris Source Code. http://www.cs.uwa.edu.au/∼pk/Research/MatlabFns/Spatial/harris.m. HAUSDORFF, F. 1962. Set Theory. Chelsea, New York, NY. HORN, B. AND SCHUNK, B. 1981. Determining optical flow. Artific. Intell. 17, 185–203. HUANG, T. AND RUSSELL, S. 1997. Object identification in a bayesian context. In Proceedings of International Joint Conference on Artificial Intelligence. 1276–1283. HUE, C., CADRE, J. L., AND PREZ, P. 2002. Sequential monte carlo methods for multiple targettracking and data fusion. IEEE Trans. Sign. Process. 50, 2, 309–325. HUTTENLOCHER, D., NOH, J., AND RUCKLIDGE, W. 1993. Tracking nonrigid objects in complex scenes. In IEEE International Conference on Computer Vision (ICCV). 93–101. INTILLE, S., DAVIS, J., AND BOBICK, A. 1997. Real-time closed-world tracking. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 697–703. IRANI, M. AND ANANDAN, P. 1998. Video indexing based on mosaic representations. IEEE Trans. Patt. Analy. Mach. Intell. 20, 6, 577–589. ISARD, M. AND BLAKE, A. 1998. Condensation - conditional density propagation for visual tracking. Int. J. Comput. Vision 29, 1, 5–28. ISARD, M. AND MACCORMICK, J. 2001. Bramble: A bayesian multiple-blob tracker. In IEEE International Conference on Computer Vision (ICCV). 34–41. JAIN, R. AND NAGEL, H. 1979. On the analysis of accumulative difference pictures from image sequences of real world scenes. IEEE Trans. Patt. Analy. Mach. Intell. 1, 2, 206–214. JAVED, O., RASHEED, Z., SHAFIQUE, K., AND SHAH, M. 2003. Tracking across multiple cameras with disjoint views. In IEEE International Conference on Computer Vision (ICCV). 952–957. JENSEN, F. V. 2001. Bayesian Networks and Decision Graphs. Springer. JEPSON, A., FLEET, D., AND ELMARAGHI, T. 2003. Robust online appearance models for visual tracking. IEEE Trans. Patt. Analy. Mach. Intell. 25, 10, 1296–1311. JOACHIMS, T. 1999. Transductive inference for text classification using support vector machines. In International Conference on Machine Learning. 200–209. KALMANSRC. Kalman Filtering Source Code. http://www.ai.mit.edu/∼murphyk/Software/index.html. KANADE, T., COLLINS, R., LIPTON, A., BURT, P., AND WIXSON, L. 1998. Advances in cooperative multi-sensor video surveillance. Darpa IU Workshop. 3–24. KANG, J., COHEN, I., AND MEDIONI, G. 2003. Continuous tracking within and across camera streams. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 267–272. KANG, J., COHEN, I., AND MEDIONI, G. 2004. Object reacquisition using geometric invariant appearance model. In International Conference on Pattern Recongnition (ICPR). 759–762. KASS, M., WITKIN, A., AND TERZOPOULOS, D. 1988. Snakes: active contour models. Int. J. Comput. Vision 1, 321–332. KETTNAKER, V. AND ZABIH, R. 1999. Bayesian multi-camera surveillance. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 117–123. KHAN, S. AND SHAH, M. 2003. Consistent labeling of tracked objects in multiple cameras with overlapping fields of view. IEEE Trans. Patt. Analy. Mach. Intell. 25, 10, 1355–1360. KLTSRC. KLT Source Code. [http://www.ces.clemson.edu/∼stb/klt/](http://www.ces.clemson.edu/%E2%88%BCstb/klt/).

KOCKELKORN, M., LUNEBURG, A., AND SCHEFFER, T. 2003. Using transduction and multiview learning to answer emails. In European Conference on Principle and Practice of Knowledge Discovery in Databases. 266–277. KUHN, H. 1955. The hungarian method for solving the assignment problem. Naval Research Logistics Quart. 2, 83–97. KUMAR, S. AND HEBERT, M. 2003. Discriminative random fields: A discriminative framework for contextual interaction in classification. In IEEE International Conference on Computer Vision (ICCV). 1150–1157. LAWS, K. 1980. Textured image segmentation. PhD thesis, Electrical Engineering, University of Southern California. LEE, L., ROMANO, R., AND STEIN, G. 2000. Monitoring activities from multiple video streams: Establishing a common coordinate frame. IEEE Trans. Patt. Recogn. Mach. Intell. 22, 8 (Aug.), 758–768. LEVELSETSRC. Level Set Source Code. http://www.cs.utah.edu/∼whitaker/vispack/. LEVIN, A., VIOLA, P., AND FREUND, Y. 2003. Unsupervised improvement of visual detectors using co-training. In IEEE International Conference on Computer Vision (ICCV). 626–633. LI, B., CHELLAPPA, R., ZHENG, Q., AND DER, S. 2001. Model-based temporal object verification using video. IEEE Trans. Image Process. 10, 6, 897–908. LIYUAN, L. AND MAYLOR, L. 2002. Integrating intensity and texture differences for robust change detection. IEEE Trans. Image Process. 11, 2, 105–112. LOWE, D. 2004. Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision 60, 2, 91–110. LUCAS, B. D. AND KANADE., T. 1981. An iterative image registration technique with an application to stereo vision. In International Joint Conference on Artificial Intelligence. MACCORMICK, J. AND BLAKE, A. 2000. Probabilistic exclusion and partitioned sampling for multiple object tracking. Int. J. Comput. Vision 39, 1, 57–71. MACKAY, D. J. C. 1998. Introduction to Monte Carlo methods. In Learning in Graphical Models, M. I. Jordan, Ed. NATO Science Series. Kluwer Academic Press, 175–204. MALLAT, S. 1989. A theory for multiresolution signal decomposition: The wavelet representation. IEEE Trans. Patt. Analy. Mach. Intell. 11, 7, 674–693. MANSOURI, A. 2002. Region tracking via level set pdes without motion computation. IEEE Trans. Patt. Analy. Mach. Intell. 24, 7, 947–961. MATTHIES, L., SZELISKI, R., AND KANADE, T. 1989. Kalman filter-based algorithms for estimating depth from image sequences. Int. J. Comput. Vision 3, 3, 209–238. MEANSHIFTSEGMENTSRC. Mean-Shift Segmentation Source Code. http://www.caip.rutgers.edu/riul/research/ code.html. MEANSHIFTTRACKSRC. Mean-Shift Tracking Source Code. http://www.intel.com/technology/computing/opencv/ index.htm. MIKOLAJCZYK, K. AND SCHMID, C. 2002. An affine invariant interest point detector. In European Conference on Computer Vision (ECCV). Vol. 1. 128–142. MIKOLAJCZYK, K. AND SCHMID, C. 2003. A performance evaluation of local descriptors. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 1615–1630. MITTAL, A. AND DAVIS, L. 2003. M2 tracker: A multiview approach to segmenting and tracking people in a cluttered scene. Int. J. Comput. Vision 51, 3, 189–203. MOESLUND, T. AND GRANUM, E. 2001. A survey of computer vision-based human motion capture. Comput. Vision Image Understand. 81, 3, 231–268. MONNET, A., MITTAL, A., PARAGIOS, N., AND RAMESH, V. 2003. Background modeling and subtraction of dynamic scenes. In IEEE International Conference on Computer Vision (ICCV). 1305–1312. MORAVEC, H. 1979. Visual mapping by a robot rover. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI). 598–600. MUGHADAM, B. AND PENTLAND, A. 1997. Probabilistic visual learning for object representation. IEEE Trans. Patt. Analy. Mach. Intell. 19, 7, 696–710. MUMFORD, D. AND SHAH, J. 1989. Optimal approximations by piecewise smooth functions and variational problems. Comm. Pure Appl. Mathemat. 42, 5, 677–685. MURTY, K. 1968. An algorithm for ranking all the assignments in order of increasing cost. Operations Resear. 16, 682–686. OLIVER, N., ROSARIO, B., AND PENTLAND, A. 2000. A bayesian computer vision system for modeling human interactions. IEEE Trans. Patt. Analy. Mach. Intell. 22, 8, 831–843. OZA, N. C. 2002. Online ensemble learning. PhD Thesis, University of California, Berkeley.

PAPAGEORGIOU, C., OREN, M., AND POGGIO, T. 1998. A general framework for object detection. In IEEE International Conference on Computer Vision (ICCV). 555–562. PARAGIOS, N. AND DERICHE, R. 2000. Geodesic active contours and level sets for the detection and tracking of moving objects. IEEE Trans. Patt. Analy. Mach. Intell. 22, 3, 266–280. PARAGIOS, N. AND DERICHE, R. 2002. Geodesic active regions and level set methods for supervised texture segmentation. Int. J. Comput. Vision 46, 3, 223–247. PARK, S. AND AGGARWAL, J. K. 2004. A hierarchical bayesian network for event recognition of human actions and interactions. Multimed. Syst. 10, 2, 164–179. PARTICLEFLTSRC. Particle Filtering Source Code. http://www-sigproc.eng.cam.ac.uk/smc/software.html. PASCHOS, G. 2001. Perceptually uniform color spaces for color texture analysis: an empirical evaluation. IEEE Trans. Image Process. 10, 932–937. RABINER, L. R. 1989. A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE 77, 2, 257–286. RANGARAJAN, K. AND SHAH, M. 1991. Establishing motion correspondence. Conference Vision Graphies Image Process 54, 1, 56–73. RASMUSSEN, C. AND HAGER, G. 2001. Probabilistic data association methods for tracking complex visual objects. IEEE Trans. Patt. Analy. Mach. Intell. 23, 6, 560–576. COLLINS, R. AND LIU, Y.. 2003. On-line selection of discriminative tracking features. In IEEE International Conference on Computer Vision (ICCV). 346–352. REID, D. B. 1979. An algorithm for tracking multiple targets. IEEE Trans. Autom. Control 24, 6, 843–854. RITTSCHER, J., KATO, J., JOGA, S., AND BLAKE, A. 2000. A probabilistic background model for tracking. In European Conference on Computer Vision (ECCV). Vol. 2. 336–350. RONFARD, R. 1994. Region based strategies for active contour models. Int. J. Comput. Vision 13, 2, 229–251. ROSALES, R. AND SCLAROFF, S. 1999. 3d trajectory recovery for tracking multiple objects and trajectory guided recognition of actions. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 117–123. ROWE, S. AND BLAKE, A. 1996. Statistical mosaics for tracking. Israel Verj. Cap. J. 14, 549–564. ROWLEY, H., BALUJA, S., AND KANADE, T. 1998. Neural network-based face detection. IEEE Trans. Patt. Analy. Mach. Intell. 20, 1, 23–38. SALARI, V. AND SETHI, I. K. 1990. Feature point correspondence in the presence of occlusion. IEEE Trans. Patt. Analy. Mach. Intell. 12, 1, 87–91. SATO, K. AND AGGARWAL, J. 2004. Temporal spatio-velocity transform and its application to tracking and interaction. Comput. Vision Image Understand. 96, 2, 100–128. SCHUNK, B. 1986. The image flow constraint equation. Comput. Visison Graphics Image Process. 35, 20–46. SCHWEITZER, H., BELL, J. W., AND WU, F. 2002. Very fast template matching. In European Conference on Computer Vision (ECCV). 358–372. SERBY, D., KOLLER-MEIER, S., AND GOOL, L. V. 2004. Probabilistic object tracking using multiple features. In IEEE International Conference of Pattern Recognition (ICPR). 184–187. SETHI, I. AND JAIN, R. 1987. Finding trajectories of feature points in a monocular image sequence. IEEE Trans. Patt. Analy. Mach. Intell. 9, 1, 56–73. SETHIAN, J. 1999. Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics Computer Vision and Material Sciences. Cambridge University Press. SHAFIQUE, K. AND SHAH, M. 2003. A non-iterative greedy algorithm for multi-frame point correspondence. In IEEE International Conference on Computer Vision (ICCV). 110–115. SHI, J. AND MALIK, J. 2000. Normalized cuts and image segmentation. IEEE Trans. Patt. Analy. Mach. Intell. 22, 8, 888–905. SHI, J. AND TOMASI, C. 1994. Good features to track. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 593–600. SIFTSRC. SIFT Source Code. http://www.cs.ucla.edu/∼vedaldi/code/siftpp/assets/siftpp/versions/. SONG, K. Y., KITTLER, J., AND PETROU, M. 1996. Defect detection in random color textures. Israel Verj. Cap. J. 14, 9, 667–683. STAUFFER, C. AND GRIMSON, W. 2000. Learning patterns of activity using real time tracking. IEEE Trans. Patt. Analy. Mach. Intell. 22, 8, 747–767. STENGER, B., RAMESH, V., PARAGIOS, N., COETZEE, F., AND BUHMANN, J. 2001. Topology free hidden markov models: Application to background modeling. In IEEE International Conference on Computer Vision (ICCV). 294–301.

STERN, H. AND EFROS, B. 2002. Adaptive color space switching for face tracking in multi-colored lighting environments. In IEEE International Conference on Automatic Face and Gesture Recognition. 0249. STREIT, R. L. AND LUGINBUHL, T. E. 1994. Maximum likelihood method for probabilistic multi-hypothesis tracking. In Proceedings of the International Society for Optical Engineering (SPIE.) vol. 2235. 394–405. SZELISKI, R. AND COUGHLAN, J. 1997. Spline-based image registration. Int. J. Comput. Vision 16, 1-3, 185– 203. TANIZAKI, H. 1987. Non-gaussian state-space modeling of nonstationary time series. J. Amer. Statist. Assoc. 82, 1032–1063. TAO, H., SAWHNEY, H., AND KUMAR, R. 2002. Object tracking with bayesian estimation of dynamic layer representations. IEEE Trans. Patt. Analy. Mach. Intell. 24, 1, 75–89. TERZOPOULOS, D. AND SZELISKI, R. 1992. Tracking with kalman snakes. In Active Vision, A. Blake and A. Yuille, Eds. MIT Press. TIEU, K. AND VIOLA, P. 2004. Boosting image retrival. Int. J. Comput. Vision 56, 1, 17–36. TIPPING, M. E. 2001. Sparse bayesian learning and the relevance vector machine. J. Mach. Learn. Resear. 1, 1, 211–244. TORRALBA, A. 2003. Contextual priming for object detection. Int. J. Comput. Vision 53, 2, 169–191. TORRESANI, L. AND BREGLER, C. 2002. Space-time tracking. In European Conference on Computer Vision (ECCV). 801–812. TOYAMA, K., J. KRUMM, B. B., AND MEYERS, B. 1999. Wallflower: Principles and practices of background maintenance. In IEEE International Conference on Computer Vision (ICCV). 255–261. VAPNIK, V. 1998. Statistical Learning Theory. John Wiley NY. VASWANI, N., ROYCHOWDHURY, A., AND CHELLAPPA, R. 2003. Activity recognition using the dynamics of the configuration ofinteracting objects. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 633–640. VEENMAN, C., REINDERS, M., AND BACKER, E. 2001. Resolving motion correspondence for densely moving points. IEEE Trans. Patt. Analy. Mach. Intell. 23, 1, 54–72. VIDAL, R. AND MA, Y. 2004. A unified algebraic approach to 2-d and 3-d motion segmentation. In European Conference on Computer Vision (ECCV). 1–15. VIOLA, P., JONES, M., AND SNOW, D. 2003. Detecting pedestrians using patterns of motion and appearance. In IEEE International Conference on Computer Vision (ICCV). 734–741. VITERBI, A. J. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans. Inform. Theory 13, 260–269. WANG, J. AND ADELSON, E. 1994. Representing moving images with layers. IEEE Image Process. 3, 5, 625– 638. WREN, C., AZARBAYEJANI, A., AND PENTLAND, A. 1997. Pfinder: Real-time tracking of the human body. IEEE Trans. Patt. Analy. Mach. Intell. 19, 7, 780–785. WU, Z. AND LEAHY, R. 1993. An optimal graph theoretic approach to data clustering: Theory and its applications to image segmentation. IEEE Trans. Patt. Analy. Mach. Intell. 11, 1101–1113. XU, N. AND AHUJA, N. 2002. Object contour tracking using graph cuts based active contours. In IEEE International Conference on Image Processing (ICIP). 277–280. YILMAZ, A., LI, X., AND SHAH, M. 2004. Contour based object tracking with occlusion handling in video acquired using mobile cameras. IEEE Trans. Patt. Analy. Mach. Intell. 26, 11, 1531–1536. YILMAZ, A., SHAFIQUE, K., AND SHAH, M. 2003. Target tracking in airborne forward looking imagery. J. Image Vision Comput. 21, 7, 623–635. YU, S. X. AND SHI, J. 2004. Segmentation given partial grouping constraints. IEEE Trans. Patt. Analy. Mach. Intell. 26, 2, 173–183. ZHONG, J. AND SCLAROFF, S. 2003. Segmenting foreground objects from a dynamic textured background via a robust kalman filter. In IEEE International Conference on Computer Vision (ICCV). 44–50. ZHOU, S., CHELLAPA, R., AND MOGHADAM, B. 2003. Adaptive visual tracking and recognition using particle filters. In Proceedings IEEE International Conference on Multimedia and Expo (ICME). 349–352. ZHU, S. AND YUILLE, A. 1996. Region competition: unifying snakes, region growing, and bayes/mdl for multiband image segmentation. IEEE Trans. Patt. Analy. Mach. Intell. 18, 9, 884–900.