Data Association for Multi-Object Visual Tracking

Synthesis Lectures on Computer Vision

Synthesis Lectures on Computer Vision is edited by Gérard Medioni of the University of Southern California and Sven Dickinson of the University of Toronto. The series publishes 50- to 150 page publications on topics pertaining to computer vision and pattern recognition. The scope will largely follow the purview of premier computer science conferences, such as ICCV, CVPR, and ECCV. Potential topics include, but not are limited to:

- Applications and Case Studies for Computer Vision
- Color, Illumination, and Texture
- Computational Photography and Video
- Early and Biologically-inspired Vision
- Face and Gesture Analysis
- Illumination and Reflectance Modeling
- Image-Based Modeling
- Image and Video Retrieval
- Medical Image Analysis
- Motion and Tracking
- Object Detection, Recognition, and Categorization
- Segmentation and Grouping
- Sensors
- Shape-from-X
- Stereo and Structure from Motion
- Shape Representation and Matching

- Statistical Methods and Learning
- Performance Evaluation
- Video Analysis and Event Recognition

Data Association for Multi-Object Visual Tracking
Margrit Betke and Zheng Wu
2016

Ellipse Fitting for Computer Vision: Implementation and Applications
Kenichi Kanatani, Yasuyuki Sugaya, and Yasushi Kanazawa
2016

Computational Methods for Integrating Vision and Language
Kobus Barnard
2016

Background Subtraction: Theory and Practice
Ahmed Elgammal
2014

Vision-Based Interaction
Matthew Turk and Gang Hua
2013

Camera Networks: The Acquisition and Analysis of Videos over Wide Areas
Amit K. Roy-Chowdhury and Bi Song
2012

Deformable Surface 3D Reconstruction from Monocular Images
Mathieu Salzmann and Pascal Fua
2010

Boosting-Based Face Detection and Adaptation
Cha Zhang and Zhengyou Zhang
2010

Image-Based Modeling of Plants and Trees
Sing Bing Kang and Long Quan
2009

Data Association for Multi-Object Visual Tracking

Margrit Betke and Zheng Wu

www.morganclaypool.com

ISBN: 9781627059558 paperback
ISBN: 9781627059435 ebook

DOI 10.2200/S00726ED1V01Y201608COV009

A Publication in the Morgan & Claypool Publishers series
SYNTHESIS LECTURES ON COMPUTER VISION

Lecture #9
Series Editors: Gérard Medioni, *University of Southern California*
Sven Dickinson, *University of Toronto*
Series ISSN
Print 2153-1056 Electronic 2153-1064

Data Association for Multi-Object Visual Tracking

Margrit Betke
Boston University

Zheng Wu
The Mathworks, Inc.

SYNTHESIS LECTURES ON COMPUTER VISION #9

MORGAN & CLAYPOOL PUBLISHERS

ABSTRACT

In the human quest for scientific knowledge, empirical evidence is collected by visual perception. Tracking with computer vision takes on the important role to reveal complex patterns of motion that exist in the world we live in. Multi-object tracking algorithms provide new information on how groups and individual group members move through three-dimensional space. They enable us to study in depth the relationships between individuals in moving groups. These may be interactions of pedestrians on a crowded sidewalk, living cells under a microscope, or bats emerging in large numbers from a cave. Being able to track pedestrians is important for urban planning; analysis of cell interactions supports research on biomaterial design; and the study of bat and bird flight can guide the engineering of aircraft. We were inspired by this multitude of applications to consider the crucial component needed to advance a single-object tracking system to a multi-object tracking system—data association.

Data association in the most general sense is the process of matching information about newly observed objects with information that was previously observed about them. This information may be about their identities, positions, or trajectories. Algorithms for data association search for matches that optimize certain match criteria and are subject to physical conditions. They can therefore be formulated as solving a "constrained optimization problem"—the problem of optimizing an objective function of some variables in the presence of constraints on these variables. As such, data association methods have a strong mathematical grounding and are valuable general tools for computer vision researchers.

This book serves as a tutorial on data association methods, intended for both students and experts in computer vision. We describe the basic research problems, review the current state of the art, and present some recently developed approaches. The book covers multi-object tracking in two and three dimensions. We consider two imaging scenarios involving either single cameras or multiple cameras with overlapping fields of view, and requiring across-time and across-view data association methods. In addition to methods that match new measurements to already established tracks, we describe methods that match trajectory segments, also called tracklets. The book presents a principled application of data association to solve two interesting tasks: first, analyzing the movements of groups of free-flying animals and second, reconstructing the movements of groups of pedestrians. We conclude by discussing exciting directions for future research.

KEYWORDS

multi-object tracking, multi-target tracking, data association, multi-view tracking, multi-camera tracking, tracklet association, tracklet linking, tracklet stitching, tracking evaluation, MOT evaluation, Bayesian recursive filter, Bayesian multi-target tracking, Bayesian multi-object tracking, people tracking, animal tracking, group tracking, tracking bats, tracking birds

Contents

Preface

This book is designed to give an overview and introduction to the exciting field of visual tracking. It focuses on data association, which is needed when multiple objects must be followed through time. The tracking problem becomes particularly challenging in dense situations when numerous objects of interest have to be tracked simultaneously. These could be twenty-two soccer players in a stadium, fifty bats in the sky, or a hundred cells in a petri dish.

Both authors have contributed original research to the field of multi-object tracking. The first author wrote her first research paper on multi-object tracking in 1996. She has since designed computer-vision systems for tracking cars, people, bats, birds, and cells. The second author, her former Ph.D. student, postdoc, and now esteemed colleague, wrote his dissertation on "Occlusion Reasoning for Multiple Object Visual Tracking" at Boston University in 2012. He is particularly known for his work on tracking flying animals with multiple cameras and coupling multi-object detection and data association in one global optimization framework.

This book describes some of the authors' own work which was supported financially by the National Science Foundation (IIS-1421943, IIS-0910908, IIS-0855065, IIS-0326483, IIS-0308213, CNS 0202067), the Office of Naval Research (024-205-1927-5), and the Air Force Office of Scientific Research. The authors are thankful for these grants and stress that any opinions, findings, and conclusions or recommendations expressed in this book are those of the authors and do not necessarily reflect the views of these funding agencies.

The authors thank the current and former members of the Image and Video Computing Research Group at Boston University for sharing their insights. Dr. Matej Kristan and Dr. Michael Felsberg gave us valuable feedback to improve the presentation of this book. We are also grateful for our parents and family members for their continuous support, and dedicate this book to them.

Margrit Betke and Zheng Wu
Boston, Massachusetts
September 2016

CHAPTER 1

An Introduction to Data Association in Computer Vision

Research on multi-object tracking has a long history in computer vision. The first systems recorded image sequences with a single video camera. In recent years, imaging systems that use several cameras with overlapping fields of view have become attractive because they enable stereoscopic reconstruction of three-dimensional (3D) object trajectories. Analyzing 3D trajectories is particularly useful when the task is to track a large number of objects [Wu et al., 2009a]. In large groups, objects may not be visible in all camera views at all times because they occlude each other. Their 3D positions may then only be inferred from information about their paths before and after the occlusion event. It is therefore important that the multi-object tracking system has an "occlusion reasoning" component. At the center of this component lies "data association." Across-time data association is the process of matching currently observed objects with previously established object tracks. Across-view data association matches objects that appear in the views of different cameras. This book first focuses on across-time data association for single-camera systems and then discusses solutions that combine across-time and across-view strategies for multi-object tracking with multi-camera systems. The book provides a comprehensive description of traditional approaches, algorithmic developments, implemented systems, and application results.

1.1 CHALLENGES

Although the problem of multi-object tracking has been studied for decades, research toward robust solutions that specifically work with visual data is ongoing. There are various reasons why visual tracking is so difficult. Before an object can be tracked, it must be detected. Methods for object detection, however, still lack robustness. Another reason is the poor scalability of the traditional data association methods to handle large numbers of objects. Frequent inter-object occlusions are sources of difficulties for both object detection and data association methods. When occlusion happens, detection and data association often become unreliable because assumptions break that existing tracking systems make. Innovative systems are needed that can reason successfully about occlusion. "Occlusion reasoning" is a crucial component to boost the accuracy of a tracking system. Occlusion reasoning is therefore an important focus of this book.

Two directions of research may be followed to tackle the problem of occlusion. One is to develop batch processing solutions, which means, instead of making decisions based on the observations seen so far, one decision at a time (online), decisions are made based on the observations

from the entire tracking sequence (offline, in a batch). The offline approach is only viable if the specific task to be solved does not require a real-time analysis but permits a wait for the solution until after the full video has been recorded. A compromise on the timing requirement yields an algorithm that computes results online but delayed, based on the observations in a sliding window of time.

The second research direction this book discusses is to provide multiple views of the objects with the expectation that, when occlusion happens in one view, it may not happen in other views that can be then used to track objects more reliably. This approach, however, introduces the need for across-view data association, a challenging process that requires spatial calibration of the cameras.

A third direction of research in multi-object visual tracking concerns the efficiency of algorithms. How fast a system computes and interprets object trajectories is a central question that is particularly important when large groups of individuals must be tracked. To speed up the tracking process, ongoing research in offline visual tracking considers a version of data association that matches trajectory segments, "tracklets," instead of position measurements.

1.2 RELATED TOPICS BEYOND THE SCOPE OF THIS BOOK

This book does not consider the tracking problem in a camera network, as, for example, the companion book by Roy-Chowdhury and Song [2011] in this series. Although camera networks may include cameras with overlapping fields of view, typically these networks have large numbers of cameras with distinct fields of views, so as to cover large areas for surveillance. This book also does not describe object detectors used in tracking and the various appearance-based tracking methods.

Multi-object tracking systems can be used for visual counting of people or animals in a crowded scene [Betke et al., 2008]. The denser the crowd is, the more difficult it becomes to count accurately. There are methods that estimate the size of a crowd without explicitly tracking each member of the crowd [Chan and Vasconcelos, 2012, Idrees et al., 2013, Lempitsky and Zisserman, 2010]. These methods do not use individual object detectors or trackers, and thus do not need data association of individuals. Describing these methods is therefore beyond the scope of this book.

1.3 APPLICATION DOMAINS

The application domain for multi-object tracking that has been most widely studied in computer vision is pedestrian tracking. Surveillance is the motivation of many works that analyze the movements of people in outdoor or indoor environments. In this book, the focus is on tracking methods that involve large groups of people. Various methods for trajectory-based abnormality detection in a crowd, for example, have been proposed [Ali and Shah, 2008, Andrade et al., 2006, Bros-

tow and Cipolla, 2006, Wang et al., 2008]. Street scenes are also analyzed for tracking multiple vehicles simultaneously [Betke et al., 2000].

In ecology and conservation biology, tracking systems are applied to study animal abundance and behavior. Censusing populations of animals is imperative for quantifying their ecological and economic impact on terrestrial and aquatic ecosystems and facilitating conservation efforts [Betke et al., 2008]. The behavior of caged and wild animals is investigated in laboratories and the field, respectively, with single or multiple camera systems. This book gives an overview of multi-object tracking systems used in ecology research.

From the perspective of a computer vision researcher, an important criterion to characterize the different application domains is whether the tracking system can make the assumption that the movement is constrained by a plane [Khan and Shah, 2006]. This plane may be the ground pedestrians walk on, the surface of a lake that ducks swim on, or the thin layer of hydrogel that live cells move in. Works that use this "ground plane assumption" for biological applications include two-dimensional tracking of swimming ducks [Lukeman, 2014], dancing bees [Veeraraghavan et al., 2008], meandering fibroblast cells [House et al., 2009], and foraging mosquitofish [Herbert-Read et al., 2011] and golden shiners [Katz et al., 2011] in shallow fish tanks.

Various tracking systems have been developed for biological applications that reconstruct unconstrained three-dimensional movements of animals in flight. A ground-plane assumption is then not made. Works that describe systems for studying flying insects include three-dimensional tracking of fruit flies [Wu et al., 2011a] and midges [Attanasi et al., 2014b]. Natural flocks of birds and colonies of bats have been studied in field experiments, including Mexican Free-tailed Bats [Betke et al., 2008, Theriault et al., 2014], European starlings [Ballerini et al., 2008b], and chimney swifts [Evangelista et al., 2015, Theriault et al., 2014].

In cell biology and biomaterial engineering, the behavior of live cells moving under the microscope is analyzed with visual tracking systems [Bise et al., 2009, House et al., 2009, Li et al., 2007, 2008b, 2006, Maška et al., 2014, Rittscher, 2010]. In medicine, the movement of multiple surgical clips surrounding abdominal tumors is tracked in preparation for radiation treatment [Betke et al., 2006].

The need of forensics specialists to interpret crime scenes motivated an out-of-the ordinary application of multi-object tracking—Zarrabeitia et al. [2014] developed a technique to reconstruct the three-dimensional trajectories of blood droplets.

Computer graphics is an important application area for multi-object visual tracking systems. Data-driven graphics approaches aim to learn from real-world trajectories of individuals in large groups. Analysis of measured trajectories and behaviors are then used to statistically train models for simulation of group motion. For example, Lee et al. [2007] extracted two-dimensional trajectories of individuals in a human crowd from an aerial view and then learned an agent model based on the features of the extracted trajectories to simulate a virtual crowd. Li et al. [2015] proposed a data-driven approach to simulate insects at different scales in order to generate virtual

insect swarming. Wang et al. [2015] introduced a quantitative metric to compare the results of their simulations with real-world trajectories of insect swarms.

1.4 SIMULATION TESTBEDS

Simulation testbeds are valuable tools that are sometimes underutilized. They can be helpful for two tasks: (1) planning the recording of video data and (2) validating algorithm performance.

For the first task, simulation testbeds are useful for scientists who aim to collect new multi-object data sets and develop tracking systems that interpret these data. To capture videos of group motion with multiple cameras successfully, practitioners should consider simulating the experiment before conducting it. This is particularly advisable in scenarios where the motion of individual group members cannot be influenced by the experimenter, for example, when filming free-flying birds or bats. A simulation can help scientists to choose appropriate camera equipment and carefully consider the placement of their cameras.

Theriault et al. [2014] provided source code for simulating multi-camera experiments.

Towne et al. [2012] also simulated the design of stereo vision system with overlapping fields of view that was aimed to analyze flight trajectories of bats.

For the second task, virtual worlds with simulated multi-object motion can be created that serve as ground-truth synthetic data for algorithm validation. For example, Wu et al. [2009a] created a simulation of a column of virtual bats flying in tight formations. The simulation enabled Wu et al. to test their data association algorithm on a range of difficulty levels, from sparse to highly dense groups.

Another example for synthetic experiments to evaluate the performance of a tracking system was provided by Zarrabeitia et al. [2014]. They conducted a synthetic experiment with 100 trajectories of spherical objects that simulated the movement of blood droplets. The performance of the proposed method and alternative algorithms was then evaluated on these trajectories.

1.5 EXPERIMENTAL BENCHMARKS

Among the various application domains for multi-object tracking, analyzing the movements of people is widely studied in computer vision. Various benchmarks have been created, most notably, the PETS 2009 benchmark [PETS] and the MOTChallenge [Leal-Taixé et al., 2015a,b]. Chapter 9 critically discusses the common use of the PETS 2009 benchmark.

To address the need for a benchmark with video data from beyond the visible spectrum, Wu et al. [2014] created the thermal infrared video benchmark TIV that includes videos consisting of almost 64,000 frames of pedestrians, marathon runners, bicycles, vehicles, and flying animals. TIV comes with ground truth annotations and source code of baseline methods, which researchers can use to compare against.

A benchmark for comparison of cell-tracking algorithms was introduced by Maška et al. [2014] and tested by six algorithms submitted to the Cell Tracking Challenge workshop at the

IEEE International Symposium on Biomedical Imaging 2013. A winner of the challenge was not declared, and continuing research on multi-cell tracking relies on the use of the benchmark.

1.6 ORGANIZATION OF THE BOOK

This book is both an introduction to traditional data association methods and a comprehensive and critical compilation of state-of-the-art approaches. Consistency of notation across book chapters was sometimes sacrificed for the sake of consistency with the notation used in the original works. The reason for this notation choice was to make it easier for readers to learn about a method and its relationship to other works in this book and then deepen their understanding of the method by studying the original paper where it was introduced.

The remainder of the book is organized as follows.

Chapter 2 formally introduces the problem of data association and discusses traditional sequential data association methods. These include Joint Probabilistic Data Association (JPDA), Multiple Hypothesis Tracking (MHT), and the Global Nearest Neighbor Standard Filter (GNNSF), and they process video data when it arrives, frame by frame or in a "sliding window." After 30 years, these methods are still being used and improved over time, specifically to address the multi-object tracking problem.

Chapter 3 expands the introduction of classic methods to batch data association approaches. Batch approaches have access to the entire video data at once (i.e., do not have to wait until the data arrives sequentially), and therefore have the conceptual advantage that they can interpret the entirety of the visual information and thus potentially make fewer association mistakes. We will discuss the Markov Chain Monte Carlo Data Association (MCMCDA) method, the Network Flow Data Association (NFDA) method, and the Probabilistic Multiple Hypothesis Tracking (PMHT) method.

Chapter 4 introduces evaluation measures for tracking systems, in particular, the widely used *USC* and *CLEAR MOT* criteria. In addition to the classical measures of false positive and false negative track detections and precision in detection, these criteria also include track interruptions and switches. In this chapter it is argued that scores reported in the literature should be interpreted with a "grain of salt"—accepted but with a degree of skepticism about their utility. More research is needed on protocols for evaluating tracking systems.

Chapter 5 extends the problem of tracking multiple objects from a single camera to multiple cameras with overlapping fields of view. Here, in addition to *temporal* data association, i.e., frame-by-frame, the *spatial* data association, view-to-view, must be considered. The chapter discusses how to solve the two *across-view* and *across-time* data association steps for tracking dense groups of objects moving in free 3D space. Two methods, the "tracking-reconstruction method" and the "reconstruction-tracking method" are described that differ in the order in which the temporal and spatial associations are resolved.

Chapter 6 describes the track linking problem, which is also called the offline "tracklet stitch problem." The chapter describes a graph representation useful for track linking. The graph

characterizes the object-interaction events that lead to occlusion in video sequences. A combinatorial algorithm is then introduced that processes this graph to resolve occlusion ambiguities. The chapter gives guidance on how to formulate the resolving process as a bipartite matching, minimum-cost flow, or set-cover problem, depending on the space-time characteristics of the occlusion events.

Chapter 7 presents three extensions of the traditional multi-object formulations. The first handles merged and split measurements, the second incorporates machine learning, and the third combines detection and data association into a single objective function.

Chapters 8 and 9 discuss the challenges that are particular to different application domains. The focus of Chapter 8 is the tracking of flying animals, which requires the reconstruction of three-dimensional trajectories. Chapter 9 reviews multi-object tracking systems for pedestrians and vehicles in urban scenes, which requires the reconstruction of planar motion.

Chapter 10 concludes with a discussion of future directions of research.

CHAPTER 2

Classic Sequential Data Association Approaches

The radar literature describes fundamental algorithms for tracking multiple targets within a dynamic system [Bar-Shalom and Fortmann, 1988]. This chapter reviews the most important sequential data association algorithms as they apply to the problem of visual tracking of multiple objects in an online process. Disambiguating measurement-to-track[1] associations for all objects in a scene may not be possible within one time step, especially if the objects have similar appearance. Nonetheless sequential tracking methods, in particular, the Global Nearest Neighbor Standard Filter (GNNSF) (Sec. 2.3) and the Joint Probabilistic Data Association (JPDA) method (Sec. 2.4) are popular, which must, in one time step, process the set of candidate assignments and decide on the most likely measurement-to-track associations.

If the requirement for sequential, time-step-by-time-step decisions can be relaxed, the likelihood of candidate associations can typically be estimated more accurately using a "look-ahead" or "deferred-logic tracking," approach. Uncertainties in the current time step may be resolved when evidence for or against a hypothesized association has been collected in subsequent frames. The classic deferred-logic method is Multiple Hypotheses Tracking (MHT) (Sec. 2.5). Multiple Hypotheses Tracking enumerates all possible combinations of object associations through time by building a hypothesis tree, and selects the best path through the tree, i.e., the path with the highest likelihood, as its solution. In practice, MHT requires various heuristics to prune the hypothesis tree in order to avoid its exponential growth [Cox and Hingorani, 1996].

Throughout this chapter, we assume a method for tracking a single object is given. Common choices are the well-known Bayesian recursive estimation techniques such as Kalman filtering [Brown and Hwang, 1997] or particle filtering [Isard and Blake, 1998, Pérez et al., 2002]. We here briefly describe the Kalman filter as an example for a single-object tracker (Sec. 2.1) that is particularly suited for multi-object tracking—each object is simply tracked by a separate Kalman filter.

[1]We use "measurement," "observation," and "detection" interchangeably in this book.

2.1 ADVANTAGES OF KALMAN FILTERS FOR USE IN MULTI-OBJECT TRACKING

The Kalman filter solves the "kinematic state estimation problem," which refers to computing the prediction $\hat{x}(t)$ of the n-dimensional state $x(t)$ of an object in a tracking scenario. In computer vision, the object state typically means its projected 2D position in an image and respective apparent velocity, or its 3D position and 3D velocity in the scene. The state can also include other time-varying quantities, for example, a point in the wing flap cycle of a tracked bird. The state equation of the Kalman filter has a Markov form:

$$x(t+1) = A(t)x(t) + w(t), \tag{2.1}$$

where $A(t)$ is an assumed known state transition matrix, and $w(t)$ is zero-mean, white, Gaussian process noise with assumed known covariance $Q(t)$. The process noise $w(t)$ models randomness due to the movement of the object, for example, its sudden acceleration. Measurement $z(t)$ is modeled as a linear combination of the system state variable $x(t)$:

$$z(t) = H(t)x(t) + v(t), \tag{2.2}$$

where $H(t)$ is a $m \times n$ measurement matrix and $v(t)$ is zero-mean, white, Gaussian measurement noise with covariance $R(t)$. For readers unfamiliar with the Kalman filter, we recommend signal processing books [Bar-Shalom et al., 2001, Kay, 1993, Stone et al., 1999] and online resources (`http://www.cs.unc.edu/~welch/kalman/`) for a more detailed introduction. There are also short introductions to Kalman filtering in computer vision text books [Forsyth and Ponce, 2003, Sonka et al., 2008].

The Kalman filter has a number of advantages for application to multi-object tracking [Blackman and Popoli, 1999]:

1. Matrix $E[(x-\hat{x})(x-\hat{x})^T]$, which is also called the *state covariance matrix* P, provides a convenient measure of estimation accuracy. The reason is that, for each component of the state vector x, the Kalman solution $\hat{x}$ minimizes the Bayesian Mean Squared Error

$$\text{BMSE}(\hat{x}_i) = Diag(E[\,(x-\hat{x})(x-\hat{x})^T\,], i), \tag{2.3}$$

   where $Diag(P, i)$ indicates the ith diagonal element of matrix P. As long as the object dynamics and the measurement noise are accurately modeled, the BMS error is minimized (we recommend the book by Kay [1993] for an insightful derivation of the optimality of the Kalman filter). The estimation accuracy is needed as input to some data association functions.

2. The Kalman filter has a closed-form solution to maintain the prediction $\hat{x}(t|t-1)$ and correction $\hat{x}(t|t)$ of the state x, as well as the prediction $P(t|t-1)$ and correction $P(t|t)$ of the state covariance matrix P.

3. The Kalman filter automatically adapts to changing detection histories and thus handles missed detections due to occlusion by other objects.

4. The residual vector

$$\tilde{z}(t) = z(t) - H(t)\,\hat{x}(t|t-1), \tag{2.4}$$

also called the *innovation vector*, is the difference between the actual and predicted measurements with covariance matrix

$$S(t) = H(t)\,P(t|t-1)\,H^T(t) + R(t). \tag{2.5}$$

By increasing the elements of this residual covariance matrix S, it is possible to model the expected error due to uncertain data association and thus, at least partially, compensate for the effects of misassociation in dense multi-object tracking scenaries.

2.2 GATING

A technique called "gating" is used by many data association methods to reduce computation. Gating defines a search region for a list of measurement-to-track candidates and eliminates unlikely measurement-to-track candidates that are located outside this region. The validation gate is usually set up around measurements using the residual covariance matrix $S(t)$. Association is allowed within a gate if the norm d^2 of the residual vector $\tilde{z}(t)$ is bounded by G, i.e.,

$$d^2(t) = \tilde{z}(t)^T\,S^{-1}(t)\,\tilde{z}(t) \;\leq\; G. \tag{2.6}$$

Since the m-dimensional Gaussian probablity density f for the residual $\tilde{z}$ is

$$f(\tilde{z}) = \frac{1}{2\pi^{m/2}\sqrt{det(S)}}\exp(-d^2/2), \tag{2.7}$$

the gate G is an iso-probability contour obtained when intersecting a Gaussian with a hyperplane. The shape of the validation gate is a hyper-ellipsoid, as illustrated in Fig. 2.1, middle. The norm d^2 is assumed to have a chi-squared distribution with degrees of freedom determined by the dimension m of measurements [Blackman and Popoli, 1999].

2.3 GLOBAL NEAREST NEIGHBOR STANDARD FILTER (GNNSF)

Probably the simplest but most widely used data association method is the Global Nearest Neighbor Standard Filter (GNNSF). It considers all possible measurement-to-track assignments within appropriate gating regions and generates the most likely assignment hypothesis by solving a 2D binary assignment problem. The assignment hypothesis is used to set irrevocable assignments

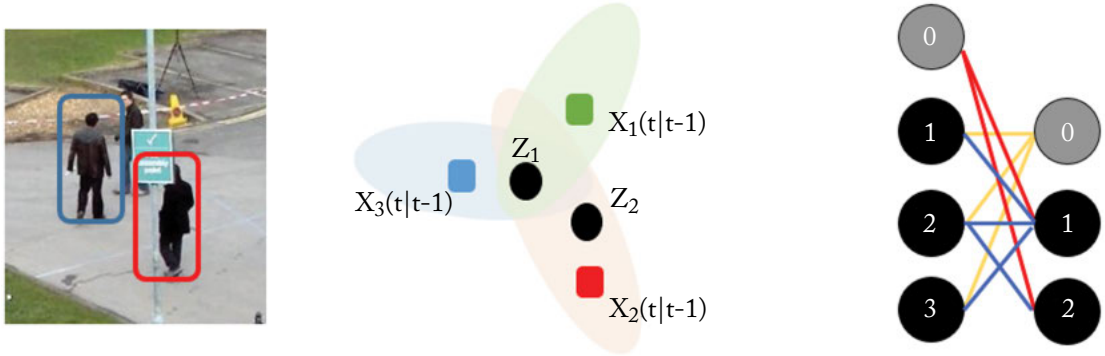

Figure 2.1: Example of data association problem with Global Nearest Neighbor Standard Filter (GNNSF). The scene contains three pedestrians walking toward each other. At time t, two measurements z_1 and z_2 were returned by the pedestrian detector, while one pedestrian who is occluded by the street sign was missed. Detections falling into the gating region of an object, shown as a shaded region, are potential assignment candidates. The gating region is typically a hyper-ellipsoid if the residual measurement vector $\tilde{z}$ is Gaussian distributed. For the GNNSF method, the cost of pairing of a measurement z and a predicted position $\hat{x}(t|t-1)$ is defined as the Mahalanobis distance between them. The cost to identify a missed detection or a false alarm also needs to be properly modeled with prior knowledge of detection and false alarm rates. GNNSF models the 2D assignment combinatorial problem as a bipartite graph where nodes on the left column represent the objects, and nodes on the right column represent the detections. Possible pairings are expressed as edges between nodes with their associated costs. The node with a zero index, also known as a "dummy node," is used as the placeholder for missing detections or false alarms. The optimal solution to this 2D bipartite assignment problem gives the best set of matchings such that, 1) every detection and object are assigned, and 2) the total cost is a minimum.

that cannot be modified by future data. A "toy example" to describe the GNNSF data association process is given in Fig. 2.1.

Briefly summarized, the GNNSF method consists of the following steps:

1. Predict the measurements and their covariances to estimate the validation gates.
2. Compute the cost for all possible measurement-to-track assignments within each gating region.
3. Formulate the 2D assignment problem and obtain a global optimal solution as the best assignment hypothesis.
4. Perform tracking by updating the state of each object and its covariance from the assignment result.

The first and last steps of the GNNSF method are standard procedures for Bayesian recursive filters such as the Kalman filter [Brown and Hwang, 1997] or the particle filter [Isard and Blake, 1998], and will therefore not be discussed here. The core step of the GNNSF method is the setup of the 2D assignment problem so that it can be solved efficiently. Formally, the objective function is

$$\begin{aligned} &\min_{x_{i,j}} \sum c_{i,j} x_{i,j} \\ &s.t. \sum_{i:i>0} x_{i,j} = 1 \\ &\sum_{j:j>0} x_{i,j} = 1 \\ &x_{i,j} \in \{0,1\} \end{aligned} \tag{2.8}$$

where $x_{i,j}$ is the binary variable to assign the ith object to the jth measurement, and $c_{i,j}$ is the corresponding cost. It is also common to add a "dummy object" and a "dummy measurement," both indexed by zero, to represent missing detections and false alarms, respectively. The constraints enforce the assignment solution to be exhaustive and exclusive, except for the dummy object/measurement.

In the optimization literature, the formulation in Eq. (2.8) is known as the bipartite matching problem [Bertsekas, 1991, Veenman et al., 2001]. Various polynomial-time algorithms solve the problem, such as the Hungarian method (also called the Munkres or Kuhn-Munkres method), the Auction method, and the Jonker-Volgenant-Castanon (JVC) method [Bar-Shalom and Li, 1995, Bertsekas and Castañón, 1992]. The algorithms all have worst time complexity of $O(n^3)$, where n is either the number N of objects or M of measurements. Their average performance differs in practice depending on the range of the cost values. Readers may refer to the book by Bertsekas [1991] for details on these algorithms.

The most compelling advantages of the GNNSF method are its simplicity and scalability. The 2D assignment problem has been well-studied in the optimization literature, and the computation can be performed very quickly in practice. A careful design of the association cost is necessary, because GNNSF makes "hard decisions" on the data association without accounting for the possibility that they might be erroneous. In fact, very often GNNSF only works well when objects are widely spaced, or are distinctive in appearance (in the latter case, the cost function evaluates appearance dissimilarity). In addition, the object detector has to maintain a high detection rate and a low false alarm rate. When a Bayesian recursive filter is used, one way to improve GNNSF performance is to increase the entries in the covariance matrix to capture the uncertainty of possible misassociation [Nahi, 1969, Singer and Stein, 1971]. Alternatively, one may consider the uncertainty in association conditions by allowing a track to be updated by a weighted combination of all the measurements in its gate. This essentially leads to the Joint Probabilistic Data Association (JPDA) method, as we will discuss in the next section.

2.4 JOINT PROBABILISTIC DATA ASSOCIATION (JPDA)

The Joint Probabilistic Data Association (JPDA) method [Bar-Shalom and Fortmann, 1988] is a sequential tracking method like GNNSF. Unlike GNNSF, which updates the track based on a single measurement from the best assignment solution, JPDA takes all the measurements within the gate into account. In JPDA, the state estimate of a track is in the form of a weighted sum of contributions given by all the feasible measurements, i.e., the expectation over all the association hypotheses.

The JPDA method relies on a few assumptions:

1. The number of established objects, N, at the given time step must be known. This implies that the classic JPDA method does not explicitly handle track birth and termination.

2. Each object has its own motion dynamic and measurement models, e.g., Eqs. (2.1) and (2.2). The past state estimates of the objects, under the Markovian assumption, are given by sufficient statistics, such as mean and covariance.

3. Each object produces at most one measurement, and each measurement either originates from a unique object or from background clutter.

We use the notation $X^{(t)} = \{x_1, x_2, ..., x_N\}$ to define the set of N object states at time t and $Z^{(t)} = \{z_1, z_2, ..., z_M\} \bigcup Z^{(t-1)}$ the set of M measurements at time t and all the previous measurements up to $t-1$. The task of the JPDA algorithm is to evaluate the conditional probability of the joint assignment event:

$$\theta = \bigcap_{j=1}^{M} \theta_{j,i_j} \tag{2.9}$$

where θ_{j,i_j} is the event that the measurement z_j originates from the object in state x_{i_j}, where $0 \leq i_j \leq N$.

The JPDA method first generates two matrices. The validation matrix Ω is a binary $M \times (N+1)$ rectangular matrix that represents all feasible measurement-to-track pairings as the result of gating. Matrix Ω is defined as

$$\Omega = [\omega_{j,i}] = \begin{pmatrix} 1 & \omega_{1,1} & \omega_{1,2} & \dots & \omega_{1,N} \\ 1 & \omega_{2,1} & \omega_{2,2} & \dots & \omega_{2,N} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & \omega_{M,1} & \omega_{M,2} & \dots & \omega_{M,N} \end{pmatrix}, \tag{2.10}$$

where $\omega_{j,0}$ means the jth measurement originates from background clutter, and for all $i > 0$, $\omega_{j,i} = 1$ if the jth measurement originates from the i-th object.

The feasibility matrix $\hat{\Omega}(\theta) = [\hat{\omega}_{j,i}(\theta)]$ describes that the joint event θ can be generated from the validation matrix Ω in Eq. (2.10). It has the same size as the validation matrix Ω and

is defined by

$$[\hat{\Omega}(\theta)]_{j,i} = \hat{\omega}_{j,i} = 1 \quad \text{if event } \theta_{j,i} \text{ occurs.} \tag{2.11}$$

Each row of $\hat{\Omega}(\theta)$ has only one nonzero entry and each column (except for the first column) has at most one nonzero entry. This ensures the third assumption described earlier is fulfilled: Each object produces at most one measurement, and each measurement either originates from a unique object or from background clutter.

The next step of the JPDA method is to compute the probability of the joint event θ conditioned on all measurements $Z^{(t)}$ [Bar-Shalom and Li, 1995]. For its computation, the probability P_D of object detection is first determined experimentally. Indicator variable τ_j for the j-th measurement being generated from a true object, and indicator variable δ_i for the i-th object being associated with some measurement are used. The total number ϕ of false alarms must also be described, either with parametric or non-parametric models.

For a parametric version of JPDA, the Poisson probability mass function is typically chosen to model the number of false alarms ϕ:

$$\mu_F(\phi) = e^{-\lambda V} \frac{(\lambda V)^{\phi}}{\phi!}, \tag{2.12}$$

where V is the volume of the surveillance space (in 2D, the region of the image frame under observation; in 3D, the full observation volume) and λ is the spatial density of false alarms (average number of false alarms in the surveillance space). Using Bayes' rule, the probability of a joint event θ conditioned on all measurements up to the current time, $Z^{(t)}$, is then

$$p(\theta \mid Z^{(t)}) \;\propto\; \prod_j \{\lambda^{-1} f_{i_j}(z_j)\}^{\tau_j} \prod_i (P_D)^{\delta_i} (1 - P_D)^{1-\delta_i} \tag{2.13}$$

where

$$f_{i_j}(z_j) = \mathcal{N}(z_j; \hat{z}_{i_j}(t|t-1), S_{i_j}(t)), \tag{2.14}$$

and $\hat{z}_{i_j}(t|t-1)$ is the predicted measurement of object i_j with associated innovation covariance $S_{i_j}(t)$.

For the nonparametric version, a constant can be chosen to model the number of false alarms

$$\mu_F(\phi) = constant, \tag{2.15}$$

which gives the joint association probability

$$p(\theta|Z^{(t)}) \propto \phi! \prod_j \{V f_{i_j}(z_j)\}^{\tau_j} \prod_i (P_D)^{\delta_i} (1 - P_D)^{1-\delta_i}. \tag{2.16}$$

Once the probability $p(\theta|Z^{(t)})$ of the joint association event θ has been estimated using Eq. (2.13) or Eq. (2.16), the JPDA method computes the marginal association probability

$p(\omega_{j,i}|Z^{(t)})$, the probability that the jth measurement belongs to the ith object. This is achieved by summing the probabilities for all joint events θ in which the marginal event of interest $\theta_{j,i}$ occurs:

$$p(\omega_{j,i}|Z^{(t)}) = \sum_{\theta} p(\theta|Z^{(t)})\, \hat{\omega}_{j,i}(\theta). \tag{2.17}$$

These marginal probabilities then serve as the weights to update the state estimate $\hat{x_i}$ of each object:

$$\begin{aligned} \hat{x_i}(t|t) &= E[x_i(t)|Z^{(t)}] \\ &= \sum_{j} p(\omega_{j,i}|Z^{(t)})\, E[x_i(t)|\,\omega_{j,i}, Z^{(t)}] \end{aligned} \tag{2.18}$$

where the expectation of the state $x_i(t)$, given the event $\omega_{j,i}$ and all the measurements $Z^{(t)}$, can be computed following the regular state update procedure of the Kalman filter. Note that the summation in Eq. (2.18) implies that the states of the objects conditioned on the past measurements are mutually independent. For each object, JPDA updates the state of the object using all plausible measurements, each multiplied by the appropriate scalar weighting coefficient.

The computational burden of the JPDA algorithm is the generation of the feasibility matrix $\hat{\Omega}(\theta)$ and the evaluation of the marginal probability in Eq. (2.17). The number of feasibility matrices increases exponentially with the increase of number of measurements or objects, which prohibits many real-world applications with a large number of objects. Early attempts to speed up JPDA either completely circumvented the step to generate the feasibility matrices or improved the evaluation step using a tree structure [Fisher and Casasent, 1989, Roecker and Phillis, 1993, Van Wyk et al., 2004, Zhou and Bose, 1993]. Alternatively, recent work by Rezatofighi et al. [2015] tried to approximate the marginal probability with the m highest probability hypotheses. Such an approximation is based on the observation that, in practice, the m highest probability hypotheses account for almost all but a tiny fraction of the total probability mass.

We here briefly outline the approximation algorithm by Rezatofighi et al. [2015]. Instead of considering the full event space Θ of all possible joint events θ, the appoximation algorithm constructs a subspace Θ^m that only contains the most likely m entries in Θ based on their probability mass. The key is to select m so that $m \ll |\Theta|$. The summation in Eq. (2.17) is then reduced to m terms without scarifying too much accuracy. To find the m-best hypotheses, the method reformulates the data association problem as a 2D assignment problem, like the one in GNNSF, Eq. (2.8). The cost of pairing is modeled by the negative logarithm of the individual association probabilities. The objective function to find the best assignment can be expressed concisely as:

$$\min_{y \in \{0,1\}^n} C^T y, \quad s.t. \quad Ay \leq b, \tag{2.19}$$

where y is a binary indicator vector of length $N(M+1)$ such that element $y_n = \hat{\omega}_{j,i}$, and C is the cost vector with $c_n = -\log(p(z_j|x_i, \hat{\omega}_{j,i}))$. Matrix A and vector b are set to enforce the assignment constraints.

Let C_m be the m-th smallest objective value in Eq. (2.19), and $y^{(m)}$ be its corresponding solution. The solution $y^{(m)}$ can be obtained by successively adding a new constraint to the original problem:

$$\begin{aligned} y^{(m)} &= \underset{y}{\operatorname{argmin}}\, C^T y \\ s.t. \quad & Ay \leq b \\ & y \neq y^{(l)}, \forall\ 1 \leq l < m. \end{aligned} \tag{2.20}$$

Instead of solving the above integer linear programming problem m times, a more efficient binary tree partition method was proposed to remove redundant constraints and inactive variables. Details of this algorithm were provided by Rezatofighi et al. [2015].

While the computational burden of the JPDA method can be reduced by various approximation approaches, other disadvantages of the method remain. First, track birth and termination are not explicitly considered in the formulation, so they have to be handled separately. Second, using all measurements to update the covariance matrix may exacerbate the risk of misassociation, because an increased covariance matrix may introduce additional measurement candidates in the gating region of a track. Finally, the JPDA method also suffers from a coalescence problem [Fitzgerald, 1985] where objects that are close to each other will tend to come closer because of the weighted state updates. To enhance the JPDA method, many extensions have been developed [Habtemariam et al., 2013, Kennedy, 2008, Lennart et al., 2011, Mahalanabis et al., 1990, Roecker, 1995], with applications on visual tracking of pedestrian and cells [Nezamoddini-Kachouie and Fieguth, 2007, Rezatofighi et al., 2012, 2015].

2.5 MULTIPLE HYPOTHESES TRACKING (MHT)

Multiple hypotheses tracking (MHT) [Reid, 1979a] is a deferred logic approach that hypothesizes all possible data associations over time and uses measurements that are received later in time to resolve ambiguities in the current frame. In contrast to the JPDA method, which decides on the most likely measurement-to-track associations at each time step, MHT propagates the current hypotheses in anticipation of subsequent data for better estimation. It also provides a principled formulation to handle the complete life cycle of tracks including birth, growth, and termination.

Although the computational cost for the exponentially growing number of hypotheses theoretically limits the scalability of MHT, many heuristic techniques have been adopted to enable a real-time performance. Although a successful implementation of the MHT algorithm is challenging, it is probably still the most widely used algorithm in the tracking community even after 40 years since its publication in the radar literature [Reid, 1979a] and 20 years since its publication in the computer vision literature [Cox and Hingorani, 1996]. MHT still provides competitive results computer vision datasets [Chenouard et al., 2009, Kim et al., 2015].

The key computation of MHT is to evaluate the probability of a new set $\Theta_l^{(t)}$ of assignment hypotheses at time t using Bayes' rule. This set $\Theta_l^{(t)}$ refers to the lth hypothesis of a joint cumu-