Real-time Tracking of Player Identities in Team Sports

Dissertation

Nicolai Baron von Hoyningen-Huene

TECHNISCHE UNIVERSITÄT MÜNCHEN
Institut für Informatik
Lehrstuhl IX: Bildverstehen und wissensbasierte Systeme

Real-time Tracking of Player Identities in Team Sports

Nicolai Bernd Lucien Baron von Hoyningen-Huene

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzende:	Univ.-Prof. Gudrun J. Klinker, Ph.D.
Prüfer der Dissertation:	
1.	Univ.-Prof. Michael Beetz, Ph.D.
2.	Univ.-Prof. Dr. Nassir Navab

Die Dissertation wurde am 13.01.2010 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 22.04.2011 angenommen.

Abstract

The rapidly growing volume of video material is creating a strong interest in automated summarization and semantic indexing of video content. In many cases this requires computer systems to recognize and interpret the activity included in the video – a computational problem that is still unsolvable at this level of generality. The interpretation of team sports videos forms an interesting specialized sub-category of the general activity recognition problem because it is, on the one hand an important real world problem but it also has, on the other hand, some valuable simplifying structure: the visual distinctiveness of the field as well as the actors and the rules of the game that restrict the class of activities to be dealt with. The context of this research is the automated interpretation of team games based on position data of the players. The goal of perception is to automatically extract low-level trajectories labeled with their corresponding players as well as high-level semantic abstractions of a game.

This thesis investigates the problem of keeping track of player positions and identities in sports video during the games in real-time and abstracting the resulting trajectories. This computational problem is all the more complex due to four main difficulties: 1) players frequently interact and occlude each others, 2) players of the same team are very similar in appearance, 3) the appearance and characteristics of players and the field are a priori unknown and often change during the course of the game, and 4) this complex computational task must be performed in real-time.

To solve the computational problem of multi-object tracking in sports videos, we propose a distributed cognitive system that supports the probabilistic fusion of different information sources, structures the incoming perceptions by automatically building models of the tracked players and adapting these concepts online. Further, we present a method for summarizing team behavior from spatio-temporal data and outline a knowledge-based system for conceptualization.

The contributions of this thesis are (1) an innovative general multi-target tracking approach that is theoretically founded in probability theory and that exhibits linear runtime complexity in the number of measurements and targets outperforming current state-of-the-art, (2) adaptive methods to identify players based on positions as well as appearance, and (3) the implementation of a concrete real-time tracking system for soccer games ranging from the acquisition of the input signal to the final supply of abstract analyses.

Research results are experimentally validated by challenging image sequences of various domains. Further, we quantitatively evaluate the total system using the complete halftime of a world championship final captured by dynamic cameras, a publicly available dataset from English premier league captured by eight static cameras as well as for broadcasted video material.

Zusammenfassung

Die Videoanalyse ist ein häufig genutztes Werkzeug im Mannschaftssport. Die Hauptarbeit besteht dabei in der ständigen Lokalisation aller Spieler während des Spiels und der semantischen Analyse der beobachteten Bewegungen. Diese Dissertation stellt ein verteiltes kognitives System zur Automatisierung dieser Tätigkeiten vor. Zur Lösung der Aufgabe werden verschiedene Informationsquellen integriert; das System verfolgt die Spieler und entwickelt sowie adaptiert Modelle von diesen während der Laufzeit. Es bietet die automatisierte Extraktion von Teamverhalten und ermöglicht weitere Analysen über einen Konzeptualisierungsrahmen. Wissenschaftliche Beiträge bestehen in 1) einem innovativen generellen Ansatz zur Verfolgung mehrerer Objekte, 2) adaptiven Methoden zur Identifikation von Spielern anhand von Erscheinung und räumlicher Relationen mit der Hilfe von weiterentwickelten selbstorganisierenden neuronalen Netzen, und 3) der konkreten Implementierung eines Echtzeitsystems zur Verfolgung von Fußballspielern. Die Forschungsergebnisse werden umfassend, unter anderem auf Fußballspielen in voller Länge und auf Fernsehaufzeichnungen, validiert.

Danksagung

Diese Dissertation wäre ohne die Hilfe einiger Menschen nicht zustande gekommen, bzw. wäre die Zeit zur Erstellung dieser Arbeit um einiges ärmer gewesen. Zu allererst möchte ich mich bei meinem Doktorvater Michael Beetz für das Vertrauen in meine wissenschaftlichen Fähigkeiten sowie seine Erfahrungen und Visionen im Bereich der künstlichen Intelligenz bedanken. Die Finanzierung der Forschung wurde durch ihn, Bernd Radig und die Deutsche Forschungsgemeinschaft DFG ermöglicht, denen ich meinen Dank aussprechen will. Desweiteren möchte ich die Arbeit der Vorsitzenden des Prüfungskomitees Gudrun Klinker und des zweiten Gutachters dieser Dissertationsschrift Nassir Navab würdigen.

Ein großer Dank für die tatkräftige Unterstützung, die angenehme Arbeitsatmosphäre und all die anregenden fachlichen Diskussionen geht an das gesamte ASpoGAMo-Team (Suat, Bernhard, Francisco, Murat, Andreas Perzylo und Andreas Andreakis) inklusive der sportlichen Abteilung (Martin Lames, Ole, Malte und Co). Das Arbeiten mit euch hat Spaß gemacht! Quirin hat stets ohne Murren auch zeitkritische und ausserordentliche Systemadministration durchgeführt, die unsere Testsysteme und effizientes Arbeiten erst ermöglichten. Freek's Begeisterung an der Wissenschaft war und ist Motivation und Inspiration für meine Arbeit. Vielen Dank an Jan, Dominik und Andreas für die fachlichen Diskussionen während der zahlreichen Fußball-Modellversuche im kleinen Kämmerchen. Moritz und David gebührt mein Dank für ihre Beiträge zu dieser Dissertation. Dankeschön auch an Sabine und alle weiteren Kollegen vom Lehrstuhl Bildverstehen und wissensbasierte Systeme der TU München für die Zusammenarbeit und Inspiration.

Meine Freunde haben mit ihrer eigenen Sichtweise und den vielen lebenswerten Dingen neben der Informatik und Wissenschaft nicht unerheblich zu dieser Arbeit beigetragen. Meinen Eltern und Brüdern rechne ich ihre Unterstützung hoch an, auf eurem Fundament ist diese Promotion aufgebaut. Mein größter Dank gilt Bettina für ihre Nachsicht und Motivation und ihre Liebe.

Zu guter Letzt will ich die unzähligen Freizeitkicker von der Werneck im Englischen Garten in München nicht vergessen, vielen Dank für die Freude am Fußball und die manchmal so nötige Praxispause von der Theorie.

Contents

List of Figures

List of Tables

List of Symbols

Pr	probability
p	probability density function
s	sensor
x	state vector
$\hat{x}$	predicted state vector
x^i	x of particle i
x_k	x at time k
$x_{k,j}$	jth column of x at time k
N_p	current number of particles
N_{max}	maximal number of particles
N_t	number of targets
N_z	number of measurements
$\bar{x}$	empirical mean of x
m	mean
V	covariance
$\hat{V}$	predicted covariance
z	measurement vector
R	measurement noise
h	measurement model
H	linear measurement model
f	motion model
F	linear motion model
E	expectation of a random variable
v	noise
Q	noise covariance
A'	transpose of A
A^{-1}	inverse of A
Δt	time difference
$\mathcal{N}$	normal distribution
w	weight
$\tilde{w}$	normalized weight
δ	Kronecker delta function
q	importance density function
$\tilde{q}$	power spectral density
$J(l) = j$	association of measurement l and target j
$\Im(j) = l$	association of measurement l and target j
$\mathcal{Z}$	measurement space
log	natural logarithm
$\mathcal{O}$	Landau symbol
$!k$	subfactorial of k

List of Abbreviations

RBRPF	Rao-Blackwellized Resampling Particle Filter
RBPF	Rao-Blackwellized Particle Filter [213]
MCMC	Monte Carlo Markov Chain
MCJPDAF	Monte Carlo Joint Probabilistic Data Association
RBMCMCDA	Rao-Blackwellized Markov Chain Monte Carlo Data Association [135]
GNG	Growing Neural Gas [82]
LateGNG	Late Growing Neural Gas
MNG	Merge Neural Gas [232]
MGNG	Merge Growing Neural Gas
RANSAC	Random Sample Consensus [79]
PCA	Principal Component Analysis
uar	uniformly at random
i.i.d.	independent and identically distributed
MAP	maximum a posteriori
HD	High Definition video format
DV	Digital Video format
UKF	Unscented Kalman Filter [121]
EKF	Extended Kalman Filter [118]
HMM	Hidden Markov Model [197]
EMD	Earth Mover's Distance [210]
$\widehat{EMD}$	Fast EMD [187]
OWL	Web Ontology Language
GrAM	Grounded Action Models
px	pixels

Chapter 1

Introduction

> There are over 20h of video material being uploaded to YouTube every single minute.
>
> Dr. O. Heckmann, Tech Lead, Google Zürich

Multimedia archives in private and business grow at an ever increasing rate. The low cost of cameras and memory storage as well as their ease of use are the catalyst for this development. With the increasing amount of videos, the need for indexing and summarization rises to cope with the information overload. Since activities form the main content of the captured videos, understanding activities in image sequences is a key feature for future information retrieval systems.

Sports videos are a special application domain; Motions of the athletes determine the central activities of interest in such footage and video motion analysis is common practice in sports nowadays. In most sports, these motions can be sufficiently described by trajectories of players and the ball. In addition to the spatial data in terms of trajectories, the correct labeling of their originators is crucial for further analysis. However, annotating sports videos for indexing is a tedious task; the visual impression is similar throughout the game and identifying visible players can be exhausting or even impossible for the untrained. Sports video analysis consists of the extraction of spatio-temporal data labeled with the corresponding player as well as high level semantic abstractions of a game which must be automatically extracted from sports videos and made available for later retrieval. Sports are highly dynamic and thus, training for meets must be current. In the professional leagues, tactical analyses of recent games are often required during the following 24 hours to fit into the tight training

schedule. Real-time analysis meets the desires of coaches of the major clubs; its automated application can help them gain the critical difference between winning and losing.

In this work we will investigate methods for real-time tracking of the protagonists of team sports in digital video as well as automated tactical analysis based on the gathered trajectories. Tracking is defined as the localization of unique persons over time. The computational problem considered is illustrated in figure 1.1. We will develop effective and efficient algorithms to estimate trajectories and identities of all visible players during the game in footage that may be captured by single or multiple static and dynamic cameras or received from broadcast. Various features, including appearance and roles, are investigated for the purpose of identification inspired by the philosophic disquisition of Carolyn Ray on the concept of identity [202]: "Each thing that exists, is some way, or is characterized in various ways, and all these ways, including its dimensions, the ways it interacts with its environment, etc., are part of its identity." Tactical analysis and summarization of team behavior based on trajectories is subsequently addressed.

1.1 Motivation of Automated Sports Video Analysis

The research on tracking systems in sports videos provides a tool for understanding intentional activities and can make a direct impact on sports, business and society by concretely affecting applications that enhance training methods as well as viewing practices for the general public. Applications for tracking identities in sports videos are manifold, as they prove advantageous for all participants of sports events, namely coaches, judges, scientists and spectators. We will point out possible applications and their likely impact, categorized according to the different beneficiaries.

1.1.1 Improvements for training and coaching

Research on real-time tracking systems equips scientists and professionals in sports with adequate methods for automatic game analysis and compound structuring of the field that paves the way for applications of new technologies in sports. Liebermann et al. give an overview of information technologies that are used in sports [159]. The authors see the main objective of such systems as providing feedback for athletes and coaches that will increase their probability of learning. They state in [159]: "For general purposes of motor learning, the impact of basic external feedback and collateral technologies – from simple video

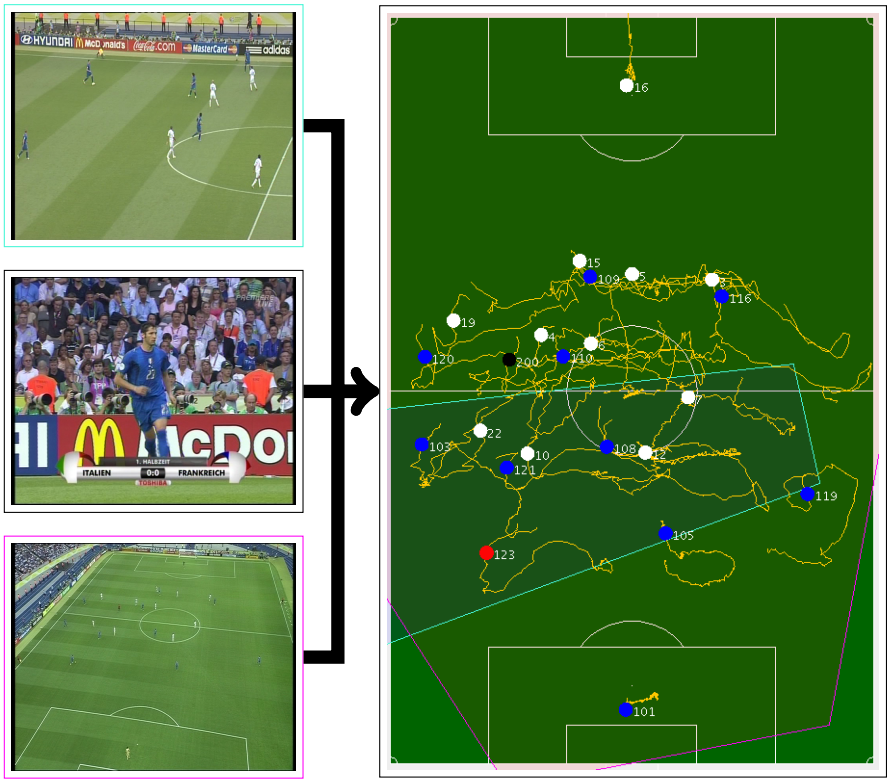


Figure 1.1: The computational problem considered in this thesis: Extract labeled trajectories of all athletes from several sports videos in real-time. On the left side, several input video streams that record the same scene from different point of views are shown. The desired output is depicted on the right side, which shows yellow trajectories as connected past positions finalized with the position at the current time as blue or white dots (according to their team affiliation). These are labeled with the corresponding player identifier. To emphasize the tracking of identities in contrast to arbitrary objects, the position of player Materazzi (jersey number 23), who is zoomed in the broadcasted close-up in the middle of the left input, is highlighted as a red dot in the output. For convenience, the captured areas are visualized as polygons of the same color as the border of the corresponding camera.

movies to complex simulators – are of major importance and should be seriously considered in the normal practice scheme".

The investigated technology allows coaches to collect trajectories and action data during training or even during competitions and replay selected scenes to the athletes to show them where improvements would be possible. The progress in the athletes' performance becomes measurable and can be compared over a

long period of time, allowing individual promotion of athletes. This advantage does not only have tactical aspects but is also lasting from the medical point of view. Physiological load diagnostics based on motion data captured during a game allows fitness trainers to adapt their program to ensure optimally effective training.

Immediate feedback is often assumed by coaches to improve skill (see [159]). Tracking systems for sports analysis should therefore provide results in (or close to) real-time. By providing immediate analysis, changes in tactics like substitutions can be organized by the coach just in time. Thus, tracking systems may substantially influence the outcome of a game. Additionally, Liebermann et al. recall in [159]: "Sometimes it may be just as effective [for learning] to give feedback information after some longer delay in a more specific and limited manner." Databases with tracked data and videos will provide many more opportunities for information retrieval than experts have ever had. For scouting purposes, future professionals may have to advertise themselves with performance curves covering their present career, just as fashion models have to present their portfolios today. The analysis of competitors is simplified (if the data are available) and weaknesses of opponents can be better taken advantage of. Teams without the technology will be forced to invest in tracking systems sooner or later to compensate the competitive disadvantage. However, since such systems come currently at a substantial price, the specter of exclusion to well financed clubs is as omnipresent as for other sports equipment (see [129]).

In addition to the undisputed advantages of tracking systems, questionnaires signal that the use of the new technology still depends heavily on the decision makers' personality. In [158] senior coaches' attitudes toward technology was studied: "Despite the fact that the coaches surveyed were generally highly experienced, those with higher education backgrounds viewed technology more favorably, but those for whom coaching was their primary livelihood did not view technology as a significant contributor to their success". Future coaches will have to bring sufficient IT knowledge to use automated sports video analysis systems or have to obtain the needed skills by trained personnel as e.g. the common use of technical staff in the NBA indicates.

1.1.2 Automatic judgment and opportunities for sport science

In the USA, instant replay judging is now common practice in hockey and football. Tracking systems can support the on-field decision by signaling irregularities via wireless communication to handheld devices worn by the referee. Larry Katz adds in [128], that "[t]he head referee on the field can be overruled by the

instant replay judges who sit and watch the replay of the action to determine the accuracy of the on-field decision". In other sports, new technologies replace the judges for detecting breaches of specific rules. Nowadays, line calls in tennis are based solely on ball tracking systems [174]. In addition to the referee assistance during video replay, tracking systems can detect goals, side-outs and offside based solely on spatial data.

As a yet disregarded operational area, anomalous performance data – detected by data mining methods in the gathered trajectories – can hint at the possible illicit use of drugs before the sporting event, although medical evidence would be still vital for final judgment. Tracking systems could also provide realistic data for simulations, which could, in the next stage, improve the education of referees and judges by allowing different viewpoints of critical game situations.

Automatic sports video analysis generates sophisticated and objective information about sports performance. Due to its design, the results are reproducible as well as comprehensible and therefore, tracking systems fit perfectly into the methodological reservoir of sports scientists. They may also give new insights into the nature of team sports because they offer precise, previously unavailable spatial data of competitions. The objective of sports science remains to investigate the applicability, advantages and shortcomings as well as the impact of the technical state-of-the-art for sports.

1.1.3 Enhancement of presentation for sports spectators

Sports spectators can benefit in various ways from the output of tracking systems. Television and media companies search for technologies to enhance images of big sports events to create greater entertainment for spectators and fans. The focus seems to be on the presentation of scenes from every possible perspective. Some research has already been done in this area of free viewpoint video generation [111, 264, 22, 276]. Grau et al. [95, 94, 96] at the British Broadcasting Corporation (BBC) investigated methods to use existing TV cameras to replay interesting incidents enhanced by 3D spatial scene information. Accurate camera calibration and player segmentation are required to produce good results; both result as part of the tracking process. With 3D models for each player – learned online by the tracking system – a virtual but still realistic image from arbitrary views would be possible even if only a few cameras are available.

Augmented reality enhancement of sport scenes is already common practice when broadcasting popular sports. But the technology is still restricted to static camera views and simple overlays. The results of tracking systems could lift this augmentation to active scenes allowing complex issues to be visualized in a

simple way. When broadcasting alpine downhill skiing, the German Television Broadcasting company ZDF superimposes the momentarily best athlete in the live video in order to increase the suspense for spectators. In the debriefing of important scenes in soccer, ghost players could visualize correct defensive behavior in the same way, thus explaining complex circumstances and tactical analyses in an appealing way.

Tracking systems can also assist in the broadcasting process (c.f. [265]) by automatic replay generation and camera view switching or selection respectively. No interesting scenes would be missed anymore by the director if signaled automatically by the system at the right time.

For the growing business segment of video-on-demand, automatic video indexing and retrieval is an important issue. Tracking systems can offer fine-grained segmentation of sports video even if the search query is not known beforehand. A huge multimedia resource center can be collected and tagged with reproducible and objective semantic information. In some cases, the tagging by tracking would be even possible for archived video footage in a batch process, creating new value for the old data. The locations of all identified players can be aggregated and matched with arbitrary patterns that could be developed in the future. In contrast to manually tagged video sequences for a special purpose, these queries would be applicable to the complete game database once instantiated. Answers to such requests can be provided as video segments augmented with query-related information. Based on the trajectories of the players and the visible areas of the cameras, the best perspective can be chosen automatically. Katz predicted the realization of elaborate audiovisual databases of player performance with instant and customizable access in [129].

The transmission of sports videos over networks with low bandwidth is still an issue. It can benefit directly from tracking system output; communication can be restricted to the position data which requires low memory compared to image sequences. After transmission, the scene is presented as virtual reconstruction on the receiving device. Mobile sports video adaption has been investigated in [265, 236]; their approach detects interesting events first and supplies video clips in turn. These clips are compressed by exploiting domain knowledge and the semantics of the presented scene.

1.2 Scientific Indexing

The research described in this dissertation was done as a part of the ASPOGAMO project. ASPOGAMO is an acronym for Automated Sport Game Analysis Model. It aims at establishing an automatic and comprehensive model for sports video analysis (see [27] for details).

The conversion from raw video data to meaningful representations of team sports is the focus of the collaborative research with sports science. Thus, the automatic abstraction of information [246] based on the positional data gathered is investigated. This process forms a key problem of Artificial Intelligence research [211] called the semantic gap, which marks the shift between raw sensor data and symbolic representations. The symbol grounding is a key problem that is faced while tracking identities. Raw input signals must be subsumed under specific player labels (or symbols); although the raw data can change over time (e.g. colors become darker due to lighting), they consistently represent the same object.

The excellence cluster CoTeSys [25] (abbreviation for cognitive technical systems) points out the importance of symbolic representations grounded in perception and action for cognitive systems. Ronald J. Brachman, Director of the Defense Advanced Research Projects Agency's (DARPA's) Information Processing Technology Office (IPTO), denominates the four core capabilities of a cognitive system as Computational perception, Representation and reasoning, Learning and Communication and interaction [34]. He designates these as the main research areas for the DARPA for mid and long term and declares: "In addition, we will be looking at the architecture of an individual cognitive system and how all of these pieces can be integrated in the most effective way." [34].

Although one may correlate the tracking of players in sports videos with the computational perception area on first sight, a closer look reveals that a tracking system for identities constitutes a complete cognitive system: it perceives the game through various sensors and it relies on an internal representation of the different player identities that can adapt over time by statistical learning; tracking results are communicated and corrective interactions can help to improve the system continuously by automatic adaption of the models. Hence, insights in tracking systems for automated sports video analysis contribute to the research of cognitive systems.

Sports embody an optimal test bed for the development and evaluation of cognitive algorithms at an early stage. They provide a wide range of complex human actions and motions on the one hand, while on the other hand, they restrict and constrain the number of protagonists and the diversity of actions in a manageable set-up. Game rules offer almost complete and already unambiguously specified domain knowledge including the main goals and intentions of each team. Visual recognition of the players is often eased by visually distinct jerseys to increase the entertainment value for the audience. Only specific interactions between players are permitted by the laws of the game to ensure fairness and thrilling competitions. Finally, the popularity of sports can boost the social acceptance of artificial cognitive systems for the average citizen.

The topic of sports video analysis can be subsumed under the Robot World Cup Initiative (RoboCup) [138], which aims to foster AI and intelligent robotics research by providing a standard problem as the future soccer world cup with robot versus human teams. Tracking identities in sports video also has its own scientific raison d'être. It opens up a great range of opportunities for further research. Inspection of the data gained by tracking systems can help us understand group behavior and natural multi-agent systems in a manageable area [112]. Human motion analysis provides a vast source for imitation learning and development of simulations for autonomous robot control. The detection of actions and their intended aims will constitute the key technology for integrating artificial cognitive systems in human society, allowing collaboration and providing a natural interface to humans.

Besides Artificial Intelligence, this dissertation touches various research areas inside computer science: Methods of computer vision are developed and used for perception. Tracking and sensor fusion is grounded in probabilistic estimators as part of control theory [13] and statistics [93], multi-target tracking of a fixed number of targets builds a subfield of these domains. Adaptivity and the segmentation inside the proposed tracking system build on results of machine learning and the nearby knowledge discovery in data bases (KDD), which include research in data mining methods[72]. The identification of persons constitutes an objective for visual surveillance; an overview is given by [108]. Due to our claim to present tracking results immediately, the proposed application can be indexed below the field of real-time systems.

Beside the methodologically based classification, this work constitutes research in automated sports video analysis as a subdivision of sport science. Surveys concerning advances in this field can be found in [254] or [275]. The area is divided in semantic and tactic analysis (see [277]). The first focuses on the detection of semantic events and the latter recognizes and discovers tactic patterns in the games. Event detection exploits cinematic features and various information sources like the World Wide Web. Tactic analysis is mostly based on the trajectories of the players and the ball which were extracted from sports videos. This work belongs largely to the area of tactical analysis. However, there is no sharp boundary between these subdivisions, as the integration of semantic analysis into the tracking framework demonstrates.

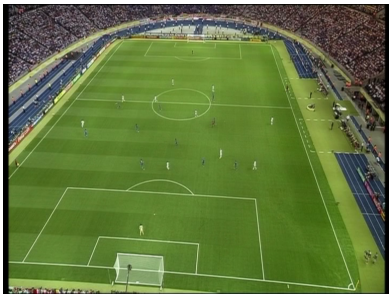
1.3 Challenges

Every tracking system for sports video analysis faces a number of technical challenges inherent in the problem and the domain of interest. Multiple interacting targets must be tracked concurrently, while occlusions of single protagonists

occur frequently and on purpose, as interactions are part of the game. The motion of human players is complex and hitherto unknown for real competition scenarios (one aim of the tracking system is to investigate the typical motions of these). Hence, the position of unseen athletes can be predicted only for a limited time horizon, which hampers the processing of cut broadcasted material. Despite good visual discrimination between different teams, athletes of the same club are hardly distinguishable, which exacerbates their re-identification after an intermission of the video stream. Although identifiers like jersey numbers are attached to the players, their usage is unreliable as they are mostly facing away from the camera or appear covered or distorted in the video image.

(a) Overlays in broadcast

(b) Shadows and reflections



(c) Small players

(d) Motion blur

Figure 1.2: Several challenges for tracking systems inherent in sports video footage.

Often, the sensory set-up of the system cannot be chosen freely or cannot even be controlled at all (in the case of broadcasted material). This restriction remains valid for the quality of the video stream input, where players occupy merely small regions in the images in order to capture a big fraction of the playing field; motion blur and compression artifacts degrade the quality even further. In addition, background clutter is typically introduced by imperfect segmentation, which cannot be excluded in practice due to the diversity of

the video and conditions of the scene input. Broadcasted material contains overlays and visual effects that block the scene of interest. The problem of synchronization and fusion of several sensors like multiple cameras is given most of the time, since a balance between coverage and level of detail must be found. Some of these challenges that are due to the characteristics of the sensor input are depicted in figure 1.2.

Many processing-relevant details like lighting conditions or the actual staffing of tracked teams are not known beforehand. Adaptivity of the tracking system is therefore required to cope with these imponderabilities. Last but not least, all of these difficulties must be handled close to real-time to gain acceptance from potential users of the system. The processing of huge amounts of data, due to the high entropy of the underlying image sequences, still exhausts modern computer hardware. Despite the expected improvements of processing speed according to Moore's law, the bulk of visual data and thus, the computational requirements, grow at a similar rate, too, due to a quasi-simultaneous upgrade of sensors and communication bandwidths, as shown by the recent introduction of the high definition format.

1.4 Contributions

The contributions of this dissertation are twofold: firstly, we propose novel algorithms that can be applied to tracking and identification of humans and athletes in general, thus advancing the state-of-the-art in the field and secondly, we incorporate and extend these methods to implement an adaptive real-time tracking system for soccer video streams.

Novel general algorithms are developed for multi-target tracking and online unsupervised vector quantization. The Rao-Blackwellized Resampling Particle Filter is especially designed to track several easily confusable targets which interact and occlude frequently. The method constitutes a sampling importance resampling particle filter for complete formations modeled as multi-Gaussians, where importance sampling is achieved by analytically solving an optimal (in the sense of maximum a-posteriori) fusion of predicted positions and measurements based on sampled associations between them. Our approach allows for constraining the multiplicity of measurements per target is possible in contrast to classical methods like JPDAF or MHT. All assumptions made in the development of the algorithm are explicated and justified. The proposed method exhibits robust real-time performance due to the linear runtime complexity in the number of measurements and targets and its inherent potential to be parallelized. While preserving or improving accuracy, it is significantly faster than the state-of-the-art MCMC approach by [135] and suffers from less theoretical

faults than state-of-the-art Rao-Blackwellized Particle Filter (RBPF) approach by [213]. Thorough evaluations of competing applications demonstrate the effectiveness and efficiency of our method.

Further, we propose novel enhancements of the state-of-the-art incremental vector quantization method Growing Neural Gas (GNG) by Fritzke [82], which forms a self-organizing neural network based on Competitive Learning with a variable number of prototypes as cluster centers. The learning algorithm is improved to also handle continuous data by delaying the update to a change in the best matching cluster for the current record and applying a more informed adaption of the past cluster prototype. We called the revised method LateGNG and observed better accuracy and faster runtime performance than the original method. The method was also extended to unsupervised clustering of time-series. The proposed Merge Growing Neural Gas (MGNG) combines the state-of-the-art recursive temporal context of Merge Neural Gas (MNG) [232] with the incremental GNG and thereby enables the analysis of unbounded and possibly infinite time series in an online manner. The algorithm has no need to define the number of clusters a priori and only constant parameters are used. In order to focus on frequent sequence patterns, an entropy maximization strategy is utilized that controls the creation of new neurons (prototypes). Experimental results demonstrate reduced time complexity compared to MNG while retaining similar accuracy in time series representation. The novel vector quantization methods are applied to learn appearance models, spatial distributions including formations of the athletes, and also utilized for the analysis of team behavior.

This dissertation discusses all steps needed to implement a real-time tracking system for computer-aided soccer game analysis. We propose a distributed adaptive real-time architecture for tracking systems in sports, integrating semantic sports video analysis. The complete process, ranging from the input sensors to the final information retrieval system for tactical analysis is covered. We provide a robust and efficient segmentation of soccer players in image sequences captured by static or dynamic cameras. Color histograms and texture are investigated and evaluated for their usefulness as features for the identification of players at a distance. Negative information is employed in tracking for players who are not captured by dynamic cameras. We further develop novel methods to extract the labeling of players based on tactical lineups and spatial information. The system is evaluated with videos captured by single or multiple static and dynamic cameras including broadcasted material, demonstrating the effective application of the developed algorithms. We are the first to publish quantitative results on soccer matches of full length.

1.5 Thesis Outline

The present thesis is organized as follows:

Chapter 2 proposes a general real-time tracking framework for sports videos with an emphasis on the big picture. After a review of scientific and commercial work, the main components and the data flow of such – across-the-board distributed – systems are identified and a framework connecting all components is explained in depth. The theoretical background for sensor fusion forms the backbone of the integration of localization, identification and tracking modules. As part of the framework, we introduce the idea behind the adaptivity and cognitivity of our system, which is able to improve its tracking capabilities over time. The chapter offers a broad view of the complete work and provides a roadmap through the following sections, which are specific to single or combined components of the framework.

Chapter 3 is concerned with the data acquisition and input of tracking systems. It thus surveys typical sensors that have been used for tracking in the sports domain. For the rest of the thesis we focus on digital cameras for the visible spectrum. Several methods for segmenting the player in images as a region of interest are presented and our approach is deduced. We review works on camera calibration and estimation that provide the basis for integrating several (possibly dynamic) sensors on a common coordinate system. Prospects for contributing event detection in broadcasts and additional information sources to a tracking system are discussed as well. In addition, the localization of players in the foreground regions is explicated in order to discern true player regions from wrong ones and robustly deducing the most probable position of athletes in individual and merged regions.

Chapter 4 proposes an innovative and general method for tracking multiple targets. After reviewing different approaches for single and multi-target tracking, we deduce our method as an estimation of the posterior probability distribution of all target positions from the theoretical point of view, elaborating all the assumptions made. We discuss implementational issues which improve tracking performance based on the theoretical foundation. The resulting algorithm is demarcated to two representative approaches of the state-of-the-art in multiple target-tracking. The evaluation of the proposed method was done independently from the system for sports video analysis and includes tracking of simulated targets, ants and of basketball players which were captured by laser range finders.

Chapter 5 considers the capability of spatial information to identify protagonists in team sports, where typically a stringent role allocation is present. We transform this problem to a search for the best assignment of all persons to their identity. Several solutions like searching, sorting, graph matching and sampling are discussed and compared for effectiveness as well as efficiency. The model for the position of each role is either borrowed from the tactical lineup (as an excerpt from a broadcast or the internet) or learned during the game. We also investigate different models for the incremental accumulation of previous positions and their impact on identification performance.

Chapter 6 reveals appearance as a key feature for identification of the protagonists on the sports field. We use the literature to extract a general framework to specify the steps for this task. The disposition of visual information for the purpose of identification is investigated in terms of increasing complexity of appearance ranging from color histograms via appearance models to gestures and gait. We propose methods for learning and classifying the appearances online and in real-time. The different methods for labeling foreground regions with player identities are compared experimentally.

Chapter 7 evaluates the real-time tracking system for soccer videos by combining the various methods of the former chapters. We investigate the performance for broadcasted material and videos that have been recorded for the purpose of tracking. Experiments include challenging sequences and soccer games of full length.

Chapter 8 is concerned with the utilization of the trajectory data that have been gathered from video footage. We survey recent works on automatic tactical sports video analysis by providing analyses of static scenes and team behavior. A novel approach for team behavior analysis is proposed in terms of automatically extracted probabilistic automatons which represents behavior by a Markov process. Further we outline a novel framework that integrates learning and logics to provide a service for elaborate information retrieval.

Chapter 9 draws final conclusions of the research done for this dissertation. We summarize the work presented in this thesis and emphasize the scientific contributions. Finally, we discuss directions for future research.

Each chapter typically begins with the computational problem under inspection and a review of related work on the specific topic. It then devises and

evaluates effective methods for solving the task in question and concludes by highlighting our contributions to the discussed field. Soccer provides the exemplary domain for the task of tracking identities throughout the thesis, but we mark methods explicitly if they are tailored specifically to soccer and cannot be easily transferred to other sports.

We have tried to keep dependencies between the chapters low; although the overall structure follows the data flow of the investigated tracking systems, the reader can simply skip sections and read those of most interest.

Some aspects of the research presented in this thesis have already been published and presented at conferences and journals to the scientific community. Below is an excerpt from the list of these publications:

- Nicolai v. Hoyningen-Huene and Michael Beetz. Importance sampling as one solution to the data association problem in multi-target tracking. In AlpeshKumar Ranchordas and Helder Araujo, editors, *VISIGRAPP 2009*, number 68 in Communications in Computer and Information Science (CCIS), pages 309–325. Springer-Verlag Berlin Heidelberg, 2010.
- Nicolai von Hoyningen-Huene and Michael Beetz. Robust Real-Time Multiple Target Tracking. In *Asian Conf. on Computer Vision (ACCV)*, Xi'an, China, Sep. 2009.
- Nicolai von Hoyningen-Huene and Michael Beetz. Rao-Blackwellized Resampling Particle Filter for Real-Time Player Tracking in Sports. In AlpeshKumar Ranchordas and Helder Araujo, editors, *Int. Conf. on Computer Vision Theory and Applications (VISAPP)*, volume 1, pages 464–470, Lisboa, Portugal, Feb. 2009. INSTICC, INSTICC press.
- Andreas Andreakis, Nicolai von Hoyningen-Huene, and Michael Beetz. Incremental unsupervised time series analysis using merge growing neural gas. In José Carlos Príncipe and Risto Miikkulainen, editors, *Proc. of Int. Workshop on Advances in Self-Organizing Maps (WSOM)*, volume 5629 of *Lecture Notes in Computer Science*, pages 10–18. Springer, 2009.
- Michael Beetz, Nicolai v. Hoyningen-Huene, Bernhard Kirchlechner, Suat Gedikli, Francisco Siles, Murat Durus, and Martin Lames. ASPOGAMO: Automated Sports Game Analysis Models. *Int. Journal of Computer Science and Sports (IJCSS)*, 2009.
- Suat Gedikli, Jan Bandouch, Nico von Hoyningen-Huene, Bernhard Kirchlechner, and Michael Beetz. An adaptive vision system for tracking soccer players from variable camera settings. In *Proc. of Int. Conf. on Computer Vision Systems (ICVS)*, 2007.
- Michael Beetz, Suat Gedikli, Jan Bandouch, Bernhard Kirchlechner, Nico v. Hoyningen-Huene, and Alexander Perzylo. Visually tracking football games based on TV broadcasts. In *Proc. of Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 2007.
- Nicolai v. Hoyningen-Huene, Bernhard Kirchlechner, and Michael Beetz. GrAM: Reasoning with Grounded Action Models by Combining Knowledge Representation and Data Mining. In *Towards Affordance-based Robot Control*, 2007.

- Michael Beetz, Jan Bandouch, Suat Gedikli, Nico von Hoyningen-Huene, Bernhard Kirchlechner, and Alexis Maldonado. Camera-based observation of football games for analyzing multi-agent activities. In *Proc. of Int. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, 2006.

Chapter 2

Cognitive Real-time Tracking Framework

All tracking systems share common structures induced by the data processing pipeline from raw input to estimated trajectories. This chapter presents an overall picture of such systems. After a review of existing tracking systems in business and research, objectives and similarities are identified and structured. The components of a typical realization can be subsumed under five layers, namely the sensor, the preprocessing, the information, the tracking and the analysis layer. Based on this categorization, we develop a general framework for sports video analysis that is of a cognitive nature. The real-time requirement demands hard response times and makes parallel execution of subtasks by a cluster of computers obligatory. The necessary organization of separate modules and their intercommunication is explicated. The proposed framework suggests the processes on the tracking layer to synchronize the submodules utilizing standard network protocols, while they integrate the computed evidence. We adopt the Bayesian approach to handle this task of information fusion in a natural and sound way. Finally, a systematic method is outlined that supports the adaptivity of the total system and thus changes its nature to a cognitive one.

2.1 Related Work

To get an idea of present implementations of tracking systems, we survey commercial products that are available for automatic tracking in the sports environment as well as frameworks published in the scientific literature for the same purpose.

2.1.1 Commercial tracking technologies in sports

The business field for tracking systems in sports video analysis is evolving rapidly and gaining increasing interest. An overview of existing commercial soccer technologies can be found in [219, 179]. Many companies have been founded as spin-offs of universities or big companies with separate research departments due to the great technological requirements needed to develop such systems. AMISCO is one of the leading systems for automated analysis of professional soccer matches captured by multiple stationary static cameras. It was developed at the University of Nice [86]. LiberoVision [157] offers free viewpoint video generation for several kinds of sports and evolved from the ETH Zurich [264]. LucentVision, a real-time analysis system for tennis matches, was developed at the Bell Laboratories (former R&D organization of AT&T) of Alcatel-Lucent [191, 192, 193]. The TRACAB Image Tracking System [241] tracks soccer players in real-time based on Saab's military tracking technologies. These companies primarily provide the service of delivering the extracted spatial data of selected games, often enriched by further semantic annotations like events and actions, after these data have been manually gathered. The tracking system is controlled and maintained as a black box transparent to the purchaser. The customer is usually equipped with software for reviewing the data and accessing predefined statistics.

Instead of supplying analysis as service, Ascensio System Ltd. [166] offers the installation of soccer tracking systems which are then operated by the purchaser. ProZone Sports Ltd. primarily provides software tools for visual game analysis, but their MatchInsight system is capable of tracking soccer players as well.

Other systems focus on video enhancing aspects. Piero, iview or Hawk-Eye have been commissioned by broadcasting companies such as the BBC [96, 228]; TrackVision has been developed by Orad Hi-Tec Systems [183]; the FoxTrax Hockey Puck Tracking System is used by the Fox Broadcasting Company [41]. Companies like Impire or Opta Sportsdata, which offer game databases and statistics for media and fans, will be switching from manual annotation to automatic tracking systems sooner or later, either by taking over new technologies or developing them from scratch. An example of the latter is Impire (subsidiary firm of Cairos), which is developing the VIS.TRACK system [3].

Such systems rely on multiple stationary, controlled cameras in the stadium, preprocessing the image sequences on the spot with the help of computer clusters supervised by salaried experts. They subsequently defer the final association of trajectories and player identities as well as semantic enrichment to remote offices. Most commercial systems have (obviously) not published their algorithms; neither have these been evaluated scientifically in respect to their tracking qual-

ity. In many cases, statistical and agglomerated data are supplied as a final product, what makes confirming their results even more difficult. Due to the black box architecture, the amount of human interaction needed for aiding the tracking process cannot be assessed externally.

2.1.2 Tracking frameworks for sports analysis

Athletes in almost all kinds of sports have been tracked in videos so far. In the following, various sports are listed annotated with an exemplary publication describing some kind of tracking in that domain: soccer [252], American football [113], hockey [154], ice hockey [36], tennis [193], volleyball [173], handball [188], badminton [44], basketball [143] and squash [143]. Although most approaches are tailored to specific domains, several more general frameworks have been proposed for tracking players in sports videos. The vast majority of these propose an implementation in the form of a distributed system to cope with the high computational demand for real-time performance. Usually, six to eight static cameras are used to cover the whole playing field while keeping the computational demand feasible for each sensor. Each camera is usually connected via optical fiber cable to an individual processing unit, which allows local tracking of the players in its field of view. An additional processing unit typically synchronizes and merges these tracks in the playing field coordinate system. In the following paragraphs, we will look at several proposed system architectures in more detail.

The Institute of Intelligent Systems for Automation in Bari has developed a distributed real-time system for ball and player tracking in soccer [151, 58]. Six high-definition (HD) camera signals are processed by multiple dual-core units with high-end graphic cards to form local tracks. A single supervisor unit synchronizes the data using a queue and merges them as the mid-point of the lines connecting the different projections.

Researchers from the Digital Imaging Research Centre have proposed a similar system [268, 269, 270]. The individual processing units for each of the eight static cameras are called Feature Servers. The centralized Tracker is responsible for collecting and synchronizing the local tracks. To keep the load of the network that connects these modules low, a single (broadcast) request is issued by the Tracker at a given time. This is handled by all Feature Servers. A message-based network protocol has been developed for communication. The Tracker acts as a single interface unit that initializes and controls the different components. The estimated game-state is finally converted to XML output that is used by third-party applications to deliver results to their respective target audiences.

The commercial LucentVision system uses eight static cameras placed around a tennis stadium to track the players and the ball in real-time [193]. The framework follows a competitive multi-threaded approach to select the camera that should be used for high-speed ball tracking, while only a single view is used for player tracking.

Müller Junior and de Oliveira Anido describe a distributed real-time system composed of four modules in [122]. One Frame Reader process per camera provides synchronized frames that are sent periodically (at a low frame rate) to Global Detector modules. These modules detect players in the full frame image. Based on what it detects, new Object Tracker processes are originated by a centralized Initiation/Verification module. The Object Trackers request a small rectangular sub-image from the Frame Reader and track the allocated player locally. The Initiation/Verification component merges the positions provided based on their proximity, skipping occluded players. They justify their solution by claiming that most computational time is usually spent on preprocessing large camera signals rather than tracking.

In semantic sports video analysis, the fusion of different information sources can be achieved by temporal alignment [266]. Duan et al. proposed a generic mid-level representation framework [60], focusing on the scalability of systems to different types of sports.

2.2 Distributed Real-time System Architecture

As the review of implementations has revealed, a tracking system for sports analysis usually employs multiple sensors to gather the raw input data. This shows the need for a distributed architecture that allows the delivery of continuous position estimates of all players (and possibly of the ball) in real-time, while simultaneously providing robustness, reliability and scalability of the total system. We propose a distributed framework that is applicable for the majority of different types of sports. We identify the main types of components of this framework and itemize their dynamic interaction. Figure 2.1 depicts the categorization of a typical tracking system for ball games with opposing teams. We explain the proposed conceptualization in the next section and detail the dynamics in the affiliating one, always referring to the scheme of figure 2.1.

2.2.1 Main components

The components of a tracking system for sports analysis can be ascribed to the following five abstract layers according to their contribution to the data flow which is shown in figure 2.1:

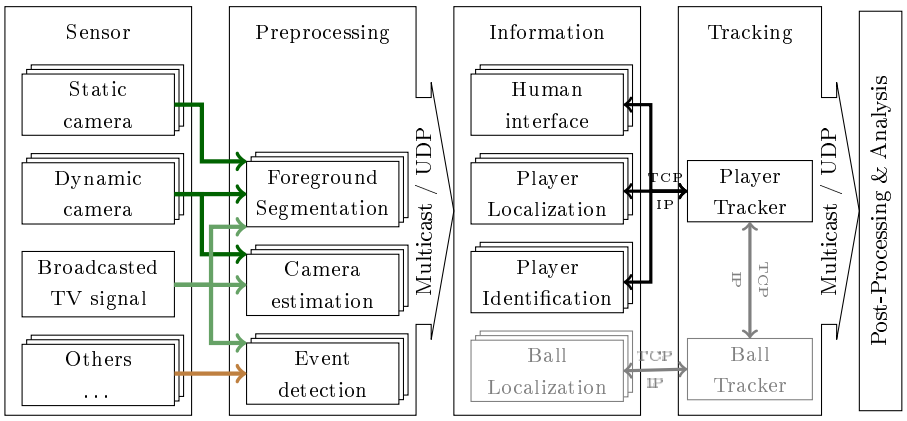


Figure 2.1: The main components of a sports video analysis system are subsumed under five layers (visualized as blocks). The data flow between them is visualized as well. Components are illustrated by rectangles inscribed by their class, while staggered rectangles point to (possibly) multiple instances of these modules. Data flow can be followed with the help of arrows: Fine arrows depict directed connections for data transmission; broad arrows represent an omnidirectional supply of data that must be subscribed to in order to receive data (multiple subscriptions are also possible). The links are labeled with the transmission protocol suggested by the framework, otherwise arbitrary, custom protocols can be used.

Sensor layer This layer contains raw signal sources that capture coarse data from the real world and send them to the next layer in digital form. The Sensor layer primarily contains separate hardware with specialized processors and software. Typical representatives are cameras, laser range finders and other kinds of mechanical and electronic devices. The choice in favor of specific components determines the rate and complexity of the total system, since non-trivial software problems can mostly be avoided by a clever application of the right sensors.

Preprocessing layer The raw input data are preprocessed by modules of this layer to reduce the need for information bandwidth. The resulting data are provided to one or many consuming processes in the Information layer. The Preprocessing layer is necessary for lowering the computational demand by avoiding redundant computations and reducing the data by filtering, aggregation and abstraction.

Information layer An information module provides evidence about probability distributions of player locations and/or their identities. This informa-

tion is based on some kind of internal semantic model for players that is either hardcoded or could have been learned and adapted incrementally. The modules must have a common interface for communicating with a Tracking process of the next layer, so that they can easily be interchanged.

Tracking layer A module of the Tracking layer synchronizes multiple modules of the Information layer and merges evidence provided from these. A conclusive estimate of the locations of all players together with their identity labels is built and made available for further analysis. Processes at this layer are time-critical and must provide some kind of scheduling or dropping of outdated evidences. A queue of the time labeled evidences is sufficient for this task.

Analysis layer The Analysis layer contains post-processing modules that refine the trajectories for persistent storage or visualization and present a selection to interested end-consumers. These processes are usually decoupled from the rest of the tracking system and are not subject to the real-time requirement.

A mainly vision-based system for sports analysis is illustrated in figure 2.1. Component classes are itemized for each layer and the typical data flow is depicted. In the following, each module is explicated according to the layer it belongs to:

The sensor layer contains static as well as panning, tilting and zooming cameras that have been (not necessarily permanently) installed at the playing field or in the stadium. Usually, multiple sensors of the same type are used, although there is no restriction to this set-up. The camera(s) may be controlled by the system or may have to be taken as given, like a broadcasted television signal. When TV and custom signals are available, the broadcasted signal should be exploited mostly as a secondary information source. Tracking in broadcasted image sequences is more difficult and time-consuming than in separate single-camera streams since the TV signal only provides discontinuous and partial views of the playing field. On the other hand, it is better suited for player identification due to its ability to zoom as well as due to informative overlays. It thus constitutes a unique source for event detection based on cinematic features. Active sensors, that belong to this layer as well, will be discussed in detail in section 3.1.

Components located on the preprocessing layer are strongly coupled with their corresponding sensors. From data of visual sensors, the foreground must be segmented because it primarily contains the player regions. Foreground segmentation processes can work on full frames of single cameras, partial images as

in [122] or on a single image synthesized from multiple cameras. The mapping of image to real-world coordinates is needed for the purpose of player localization. Individual mappings must be provided by camera estimation processes for each non-static camera. The broadcasted signal needs an additional partitioning into shots as well as mappings of each shot to the camera in use. Processes providing semantic sports video analysis can be integrated into our framework as event detectors at the Preprocessing layer. Events can serve as additional input for information processes. Corner kick detection, for example, can influence the model selection and its certainty for an identity sensor which is based on the tactical line-up of the players. Event detection may even facilitate hints concerning a player's position. For example, the performer of the penalty kick is roughly localized around an area in front of the goal. If ball tracking is employed, the position of single players could be deduced from ball action events relying on good synchronization of the different sensors used. This type of deduction of positions would be located on the information layer, although feedback from the tracking layer is utilized.

Player localization and identification components constitute the information layer. They prepare and provide the relevant evidence for the tracking process. While localization modules are usually restricted to the visual domain and implemented by multiple instantiations of the same method, identification modules can vary greatly in quality since identity is conducted of manifold continuities in appearance and behavior. Evidence about identities could have been encoded by other agents which are external to the system, and may just need to be extracted from symbolic representations. Likewise, live tickers in the World Wide Web can be parsed to gather substitutions, for example. Human operators can be modeled as processes at the information layer, providing player positions and/or identities with probabilities based on the user interface (e.g. the resolution of the presented scene image).

The tracking layer usually contains only a single tracker component for each tracking task. Players as well as various pieces of sports equipment (ball, puck...) constitute the objects of interest in most sports. Since we are concentrating on player tracking in this work, the ball components of figure 2.1 are shaded in gray for convenience.

The analysis layer covers the widest variety of components. Databases and knowledge discovery techniques are utilized to store the trajectories for later retrieval or refine the spatio-temporal data for specific analyses and statistics. This layer provides interfaces for information retrieval such as web services, delivery of enhanced video material to mobile devices or layouted documents that summarize aspects of the game. Semantic and tactical game analysis is built on this layer.

2.2.2 Distributed realization

The components of the different layers of a tracking system are implemented by hard- and software modules.

Sensors are primarily realized as separate special hardware (like CCD cameras), although this does not necessarily have to be the case (as with the World Wide Web). Contingent on their computational demand and the number of consumers, preprocessors would be realized as threads on the same or on different machines than the sensor processes, whereas the latter is common for image processing. Graphical processing units (GPU), however, could do this work as well. In rare cases like laser scans, preprocessing is already included in the sensor hardware. Video storage and preprocessing of camera frames can be handled simultaneously on the same machine using special hardware like frame grabber cards conveying the CPU usage exclusively for preprocessing. Real-time capability by design is crucial for all preprocessing nodes. A sufficient reduction of the amount of data is also important to avoid overloading the network and likewise, avoiding data loss.

Information layer processes are not restricted to a specific implementation. On the one hand, they can be located on the same machine as the preprocessing node if they are enforced to by high communication requirements. On the other hand, an information module can be distributed on several machines itself. In addition to the mentioned hardware, network routers provide the local network, where the preprocessed data are subscribed from by information modules.

We suggest an autonomous, centralized process for each tracking task (ball or players). Since the tracker builds a fusing node, it should be located on a single server with multiple high-performance network access and adequate computing facilities.

Components of the analysis layer must be implemented according to their objective. They should be implemented on separate computers as well as separate networks than the ones used by the core real-time tracking system in order to reduce interference and ensure compliance with the real-time requirement.

2.2.3 Connectivity inside the framework

To achieve the objective of delivering consistent trajectories, all modules must work together. Therefore, the underlying communication networks play a crucial role for the performance of the system as a whole. The connections between sensors and preprocessing nodes are almost exclusively one-to-one links and specific to the type of sensor used. There is no general way to describe this link; the spectrum ranges from internal PCI cards to satellite connections (e.g. for GPS). Camera signals can be transmitted directly via fiber optics or by Gigabit

Ethernet network via CAT5 cables; specialized protocols like GigE-Vision exist for this task. CAT5 cables can transfer signals over a distance up to 100m without repeaters and are therefore well suited for connecting cameras installed in stadium ceilings to the corresponding preprocessing nodes.

Distributed preprocessing modules should provide their information on the (local) network via multicast (UDP). The main advantage of using multicast is that data can be sent to multiple processes on the information layer in a flexible way, requiring a fixed network bandwidth regardless of the number of consumers. In addition, temporal decoupling is ensured since UDP is a connection-less network protocol. Although dropping of UDP packets at overload of the network buffers constitutes a prerequisite for real-time processing, care must be taken to adjust the buffer sizes for UDP to use the possible bandwidth to full capacity and to avoid data loss.

Information processes must subscribe to the desired multicasted service in order to receive the preprocessed data. We recommend detached buffering processes on the receiving machines. These buffer the incoming data to a file; the sensor process(es) work on these files only and are therefore decoupled temporally. In addition, they do not have to share the same process space, which increases the real-time capability and robustness of the total system. This buffering can be done with minimal CPU usage since it is predominantly an I/O task.

All components refer to a global clock, which can be achieved by a distributed clock via the network (a common tool in most operating systems). At the beginning, input data are tagged by a time stamp at the preprocessing layer. Synchronization of information flow is achieved by the tracker via a queue of parallel blocking TCP/IP connections to the information processes. Late information responses can be dropped by individual timeout thresholds inside the protocol, assuring the real-time capability by standard software tools that are available on every modern operating system. Information sources can be included or detached flexibly by establishing or closing a TCP/IP connection. This proves to be especially helpful for human monitoring modules.

Information is fused in a Bayesian framework (described in detail in the next section 2.3.1): the tracker requests probabilities and innovations for the current estimate given predicted positions of all players. Since this information usually requires little memory for representation, a low communication bandwidth is ensured. Although not covered in detail in this work, ball localization components and a ball tracker are often part of sports analysis systems. Ball and player tracking modules should be connected via a synchronized communication channel over which they exchange estimates of the tracked state.

The analysis layer is decoupled from the remaining system by multicast

connection (UDP). Care should be taken that no information is lost due to insufficient bandwidth or the workload of the retrieving components. An interposition component may be appropriate for providing reliable and lossless data transmission.

2.3 Sensor Fusion

There are usually multiple sources providing evidence about the location or identity of a player, be it due to the application of multiple sensors or due to exploitation of multiple cues inside a single signal. Although most of the evidence refers to visual sensors exhibiting spatial measurements, various features based on auditory or textual input could be used to obtain hints about the identity of a player in the scene. In addition, supervision by human operators can be seen as a further information source that produces measurements with presumably high certainty. All of these inputs must be merged in order to exploit a maximum of information to estimate the players' locations and to correctly label them.

2.3.1 Bayesian framework

Our framework adopts the Bayesian approach (c.f. [13, 47, 6]) to estimate the tracked state by selecting one of the possible hypotheses based on its assigned probability. This section recalls the applied concepts of probability theory.

The concept of probability Pr can be interpreted as a measure of belief in events $A, B \in \Gamma$ as outcomes of a random experiment. For a sound theory it must satisfy the axioms

$$\Pr(A) \geq 0,\ \Pr(\Gamma) = 1,\ A \cap B = \varnothing \rightarrow \Pr(A \cup B) = \Pr(A) + \Pr(B). \tag{2.1}$$

Porting discrete events to continuous values, random variables are introduced as real-valued functions assuming a certain value to the outcome of a random experiment. The probability of a realization r of a random variable is given by its probability density function p (further abbreviated as pdf) satisfying

$$\Pr(r \leq \infty) = \int_{-\infty}^{\infty} p(r)\, dx = 1. \tag{2.2}$$

Several random variables r_i are called independent iff

$$p(r_1, \ldots, r_n) = \prod_{i=1}^{n} p(r_i). \tag{2.3}$$

The conditional probability of a random variable r_1 given another r_2 is defined as

$$p(r_1 | r_2) = \frac{p(r_1, r_2)}{p(r_2)}. \tag{2.4}$$

In our framework, the locations of all players constitute the state to track and are modeled as a multi-dimensional random variable x attached with a (prior) probability density function which evolves over time. Evidence about this state can be observed (possibly partially or indirectly) in terms of measurements z. The task of the tracking system is to estimate the most likely state sequence $x_{0:k}$ up to time k given all observed evidence $z_{1:k} = \{z_1, \ldots, z_k\}$ so far. A real-time estimator supplies the current estimate x_k according to the posterior pdf $p(x_k|z_{1:k})$. The optimal Bayesian solution in finding the mode of this posterior pdf is given by the maximum a posteriori (MAP) estimate

$$x_k^{MAP} = \arg\max_{x_k} p(x_k|z_{1:k}). \tag{2.5}$$

Since this posterior cannot be computed directly, we make use of the definition 2.4 of conditional probability to convert the posterior pdf to

$$p(x_k|z_{1:k}) = \frac{p(x_k, z_{1:k})}{p(z_{1:k})}. \tag{2.6}$$

Since the maximization of equation 2.6 is independent of the denominator $p(z_{1:k})$, it can be converted to

$$x_k^{MAP} = \arg\max_{x_k} p(x_k, z_{1:k}). \tag{2.7}$$

If we assume that the measurements z_i are conditionally independent given the state x_i at their corresponding observation time i, the connection of measurements and states at time k can be described by the measurement model $h_k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_n} \to \mathbb{R}^{n_z}$ as

$$z_k = h_k(x_k, n_k) \tag{2.8}$$

with n_k denoting the independent and identically distributed (i.i.d.) measurement noise at timepoint k.

Using Bayes' rule

$$p(r_1|r_2) = \frac{p(r_2|r_1)\, p(r_1)}{p(r_2)}, \tag{2.9}$$

the MAP estimate for the posterior pdf results in

$$x_k^{MAP} = \arg\max_{x_k} p(z_k|x_k)\, p(x_k|z_{1:k-1}). \tag{2.10}$$

If we further assume that all x_i are conditionally independent given their predecessor x_{i-1} (known as Markov assumption of order one), the evolution of the state over time can be written using the process model $f_k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_v} \to \mathbb{R}^{n_x}$ as

$$x_k = f_k(x_{k-1}, v_{k-1}) \tag{2.11}$$

with the subscripts $k \in \mathbb{N}$ denoting ordered points in time, which are not necessarily equally distant. v denotes the process noise which is assumed to be independent and identically distributed (i.i.d.).

Based on the Markov assumption and the total probability theorem

$$p(r_1) = \int_{-\infty}^{\infty} p(r_1, r_2)\, dr_2, \tag{2.12}$$

we obtain the Chapman-Kolmogorov equation

$$p(x_k|z_{1:k-1}) = \int p(x_k|x_{k-1})\, p(x_{k-1}|z_{k-1})\, dx_{k-1}. \tag{2.13}$$

The joint distribution $p(x_{0:k}, z_{1:k})$ can be unfolded to

$$p(x_{0:k}, z_{1:k}) = p(x_0) \prod_{i=1}^{k} p(x_i|x_{i-1})\, p(z_i|x_i) \tag{2.14}$$

with the initial pdf $p(x_0)$ assumed to be known a priori. The initial state x_0 can be assigned to be uniformly distributed over the whole state space if no knowledge is available beforehand resulting in a maximum likelihood (ML) estimation.

Combining equations 2.10 and 2.13 we can rewrite the MAP estimate in its recursive form

$$x_k^{MAP} = \arg\max_{x_k} \int p(z_k|x_k)\, p(x_k|x_{k-1})\, dx_{k-1}. \tag{2.15}$$

The pdf, that has to be maximized, decomposes into the prior $p(x_k|x_{k-1})$ and the (measurement) likelihood $p(z_k|x_k)$. The estimation can usually be divided into two stages accordingly: the prediction stage, which relates the previous and the current state, and the update stage which incorporates new evidence from the measurements. The MAP estimate is unbiased since

$$E\left[x_k^{MAP}\right] = E[x_k] \tag{2.16}$$

with E denoting the expectation of a random variable

$$E[r] = \int_{-\infty}^{\infty} r p(r)\, dr. \tag{2.17}$$

The positive definite (or semidefinite) covariance matrix of the MAP estimate

$$var\left(x_k^{MAP}\right) = E\left[\left(x_k^{MAP} - E[x]\right)\left(x_k^{MAP} - E[x]\right)^T\right] \tag{2.18}$$

describes its quality or rather its certainty by the expected deviation of the true value.

2.3.2 Probabilistic combination

Various evidence is combined according to the equations of the Bayesian approach described in the previous section. We assume the observations of each

sensor s_i to be conditionally independent of all other sensors given the current state x_k. Therefore, the measurement likelihood factors as

$$p\left(z_k|x_k\right) = \prod_i p\left(z_k^{s_i}|x_k\right). \tag{2.19}$$

The individual probabilities can be evaluated independently with distinct memory space for each information node. This kind of fusion resembles the boosting principle [81], where multiple (albeit simple) experts vote for a decision resulting in a stronger estimate than each single one, in a natural way.

Each Player Localization module s_i provides spatial measurements $z_k^{s_i}$ in compliance with its measurement model $h_k^{s_i}$ similar to equation 2.8. Instead of using a global measurement model h_k mapping the tracked state to a stacked version of $z_k = \left(z_k^{s_i}\right)_i$, we process each localization measurement sweep after another. According to Welch and Bishop [260], this single-constraint-at-a-time or SCAAT tracking approach allows to "generate estimates more frequently, with less latency, with improved accuracy, and [...] to estimate the [...] positions online concurrently while tracking". The uncertainty in the measurement process is provided by covariance matrices for the measured positions. Player Localization components offer the currently measured positions with their covariances and the field of view they are observing.

Player Identification modules estimate probabilities for the identity of a player at a specified position. They contribute to the final estimate by the measurement likelihood of equation 2.19. All player identification components have to provide an interface to supply the probability for a given estimate of player positions.

Bayesian filtering seeks the posterior pdf which integrates all the available evidence of past and present expressed in terms of probabilities [47] and is therefore well suited for the task of sensor fusion.

2.4 Adaptivity and Cognitivity

In contrast to most industrial computer vision environments, sports provide a lot of unforeseen conditions and events. This uncertainty is evident for the measurement model $h_k^{s_i}$ of some sensor s_i in particular; the model may be unknown before the game taking place or could change over time. For instance, the tactical line-up is published only shortly before the kickoff and lighting conditions as well as the appearances of the athletes can be forecasted only roughly in advance. Additionally, these attributes may be subject to significant change by adaptive team behavior, change in the weather pattern, bandages as result of injuries and numerous other reasons from practice.

To cope with this diversity, the internal representations and parameters of the tracking system have to be adapted accordingly. Since not every single information module is usually affected at once, the models can be learned and adjusted based on the current estimate of the player positions which reflect the overall information extracted from all available sources.

With the location of a player given by the certain current estimate, supervised learning methods can be used to automatically model the relation between raw sensor input and the corresponding position or identity. Several supervised methods based on stochastic learning have been proposed in the machine learning area (see [211, 33] for an overview). Supervised learning tries to estimate a function from an input sample to its corresponding output value given a sequence of input-output pairs. The methods differ in the representation language for the function, but the learning process is always based on the statistics of the training sequence. The search for a mapping of raw sensor input to an identity constitutes a multi-class learning problem, while the mapping to a position, which constitutes a continuous variable, can be found by regression. The learning must be incrementally and online; the learned model must be available for classification all the time. Probability distributions are required for the Bayesian sensor fusion, so distributions should be learned supporting a soft classification.

Unsupervised learning methods [92] can be applied to train models given the inputs only. These models can be used for classification as soon as they are assigned to an identity. Distances of unknown data to the learned models can be transformed to the needed probability distribution. The unsupervised approach offers the possibility of deferring the association to the time when it is known, while the learning process can be active all the time. An analogous result can be achieved for supervised methods only if models can be merged afterwards. To fulfill the real-time requirements, learning and especially classification should be executed in bounded time ranges. The class of online learning methods accommodates suited candidates for this demand.

Following the bootstrapping idea, the knowledge gained from initial sensors is spread to different possibly not-initialized modalities over time which could stabilize the former ones in return. This approach introduces cognition to the tracking system, increasing its robustness. Since this bootstrap method is highly self-referential, it is vulnerable for drifting away from the correct solution. This can be prohibited if the adaptation is done in a very conservative manner by setting high thresholds for the minimal certainty in the estimate that serves as evidence for other sensors.

A somewhat similar approach was suggested by Song et al. quite recently in [226]. They differentiate two phases, which they call „tracking for learn-

ing" and „learning for tracking". Bags of features (color and texture of image patches) are sampled and learned for non-interacting targets forming several weak Classification and Regression Trees. After splitting of intersecting or occluding players, the identities are determined for further tracking according to the boosting principle as the majority vote of the individual weak classifiers.

2.5 Conclusions

In this chapter we proposed a real-time framework for computer-aided sports analysis. The components of this scheme are categorized into five layers: the sensor and preprocessing layer are tightly coupled. The latter rapidly processes raw to enhanced data and provides these to various information sources via multicast. Player localization and identification components offer probabilities and spatial measurements to the tracking layer. Synchronization is ensured by the tracker via queued TCP/IP connections on top of the network. The fusion of information follows a Bayesian approach to estimate the maximum a posteriori position of all players. The integration of semantic and tactical sports video analysis was integrated seamlessly in the framework. The remaining analysis layer refines the extracted trajectories for later retrieval and provides arbitrary knowledge discovery tools.

Further we emphasized the idea of a bootstrapping system which improves tracking performance over time by incremental adaptation through statistical learning. The current estimate serves as labeled training data and allows for supervised online learning. Destabilization of this self-referential process is inhibited by distributing evidence only if the certainty in the current estimate is high. The system can be called cognitive because it develops its own representations and transfers knowledge between the different sensor modules, enabling it to cope with previously unseen environments. Moreover, the learned models constitute interesting analyses in their own right.

Chapter 3

Data Acquisition

This chapter enumerates the different kinds of sensors that could or already have been used as primary input for computer aided sports video analysis. In the remainder we focus on sports videos captured by standard cameras and present common methods for segmenting players in single video frames. This segmentation is usually tailored to the domain it is applied to. Approaches therefore differ significantly between all kinds of sports, especially if played indoors as opposed to outdoors, or if they are based, for example, on strong assumptions like static cameras. Our approach for outdoor soccer is based on the homogeneity of the playing field. It therefore states a nonparametric method which is robust in different lighting conditions. Two computational problems are discussed in this chapter: the preprocessing phase must compute the segmentation of all potential player regions in the current video frame and the transformation of pixels to real-world coordinates; given the segmented foreground in single or multiple video streams and given known camera calibrations, the likely player locations in real-world coordinates must likewise be determined. These tasks are depicted in figure 3.1. The output from preprocessing and localization becomes the basic input for all further processing steps and is thus crucial for the performance of the other modules.

3.1 Sensors

Several different types of sensors have been used to analyze players in sports. The most straight-forward approach to gathering individual information on the players consists in affixing gauges to the protagonists. These measure and send various parameters together with an unique identifier to a central processing unit. At NFL football games, six accelerometers together with a transmitter are placed in the helmets of the players; they measure the linear and rota-

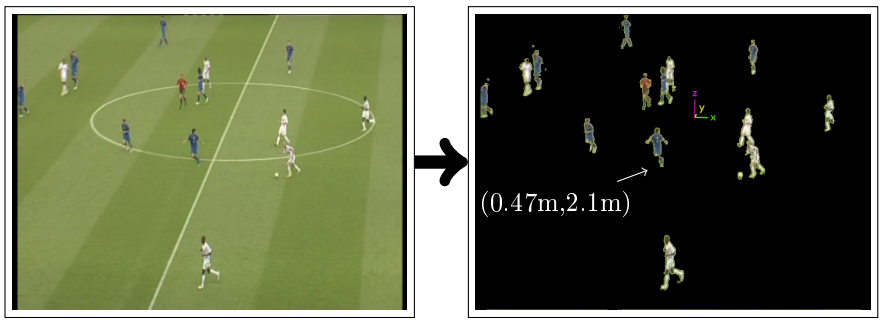


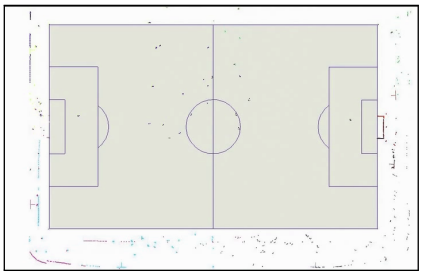
Figure 3.1: The preprocessing step segments potential player regions and provides a transformation of image to real world coordinates (visualized as the projected right-handed world coordinate system of unit length 1m). The localization determines the probable real-world positions based on these regions.

tional acceleration of the head and send the data to a sideline response system for immediate analysis (c.f. [218]). The results are used primarily for medical purposes. Active transmitters have also been attached to balls and players [216, 184, 201, 106, 239]. They send high frequency radio signals with identifiers to surrounding receivers, thus simplifying the tracking process to triangulation. On the other hand, they are very expensive. This is so because they rely on powerful transmitters and receivers that must be located around each playing field. Systems for referee support are currently seldom in use; a tracking system for soccer developed by the Cairos company in collaboration with the Fraunhofer Institute [239, 244] was stopped after preliminary tests during the U17 world championship 2005 in Peru, because of – among other things – low accuracy for the ball of about 10cm compared to an investment of 250,000 euros for each stadium.

Laser rangefinders provide an alternative which is frequently deployed in robotics for self-localization. They constitute active localization sensors that triangulate objects in 3D, relying on the time differences between reflected laser beams. Laser scans have the advantage of providing primarily depth information and allow easy (and therefore fast) player segmentation by simple thresholding. But this meal doesn't come for free. Because laser rangefinders are highly sensitive to occlusion, they can only be used up to a limited distance and lack information about the identity of the detected objects other than shape. To cope with these disadvantages, RFID tags have been attached to the players in addition. These emit a unique signal for each athlete, such that they could support identification of laser scan data. Unfortunately, experiments [10] revealed that RFID tags are not suited for this task.

The BORG Lab at Georgia Tech experimented with multiple laser rangefind-

ers in sports. Despite the fact that their projects are developed primarily for tracking animals, experiments were conducted for tracking 4-on-4 basketball and soccer games as well. Therefore, laser rangefinders were positioned at the border of the playing field; examples of perceived laser scans are depicted in figure 3.2. A non-technical drawback of laser lies in its nature as an active sensor, which is typically not permitted in competitions because it could possibly influence the play (e. g. by dazzling or distraction).



(a) Data of soccer game gathered by several laser scanners placed around the pitch.

(b) A 4-on-4 basketball game observed by four laser scanners which are depicted by colored circles.

Figure 3.2: Combined scans of several long range lasers acquired by the BORG Lab of Georgia Tech for player tracking in sports.

In some cases, infrared cameras have been applied. They have similar advantages and disadvantages as laser sensors. The body of the player can be distinguished easily from the background based on the obvious temperature difference, but ranges are short and charateristic attributes of persons are lost. The FoxTrax Hockey Puck Tracking System [41] turns infrared into an active sensor; a modified ice-hockey puck which emits infrared pulses is tracked by twenty pulse detectors and ten synchronized infrared cameras for the purpose of augmented broadcasting. Since only a single object is tracked by many cameras, the difficulties mentioned above – namely occlusions and the lack of identifiers – are circumvented.

Charge-coupled device (CCD) cameras for the visual spectrum are the most frequently applied sensory component by far because these passive sensors do not influence the game, are comparatively cheap and provide rich information about the game. An important parameter of a camera is its image format and resolution. Commonly, raw image data, PAL, DigitalVideo (DV) or High Definition (HD) are used. Videos with high resolution (like HD) reach the limit of Gigabit ethernet real-time capability and are therefore usually transmitted in a lossy compressed form. Still, e.g. HD requires a high bandwidth and also more computational power due to the necessary decompression. To cover a big

area, cameras with a wide focus are employed and thus the tracked players are as small as down to 10 pixels height if resolution is not high enough. Multiple perspectives of the same area can avoid occlusions more easily and are necessary for generating free-viewpoint video. However, Xu et al. recommend that a "[g]ood resolution of each area, especially the goal-mouths, is more important [for tracking] than multiple views of each area" [268].

If possible, static cameras are used to reduce the computational burden since special foreground segmentation techniques can be applied. Additionally, dynamic cameras state the need for continuous camera calibration, although special hardware solutions as e.g. Pan-Tilt units exist and can be utilized if the camera is accessible.

In addition, one can make use of available data previously perceived and digitalized by humans. Many sources for event detection exist, such as live newstickers of the game in the internet or television and radio commentary (c.f. [50] for automated extraction). The difficulties of extracting semantics from text and spoken language have spawned their own research areas called natural language understanding and speech recognition.

3.2 Camera Calibration and Estimation

Camera calibration is the process of determining the mapping of image to playing field coordinates and vice versa. This mapping is commonly represented by a so-called camera matrix of size 3×4, assuming that the camera satisfies the pinhole camera model with projective geometry. The matrix is derived by the intrinsic parameters of the camera (such as focal length, image size and principal point) and the extrinsic parameters that determine the position as well as the pose of the camera in the real world. If radial lens distortion is taken into account, the transformation becomes non-linear and can no longer be represented by a matrix.

The homography from image- to world-coordinates can be estimated by matching a minimum of four non-collinear points in the image to known points in real-world coordinates using Tsai's method [242]. Intrinsic camera parameters are mostly estimated with the help of a planar calibration board showing a checker-board pattern. In the sports domain, the playing field itself can often be used to calibrate the camera since its dimensions and the geometry of lines and marks are usually determined by the rules of the game (at least in professional competitions). In soccer, the field size is not fixed [76] and must be measured or estimated as well. The calibration of static cameras can be completed before the game starts.

In the case of non-static cameras, the calibration must be done continuously

and is called camera (parameter) estimation. Pan-tilt units can be used for measuring some of the changing extrinsic parameters by hardware. Contrary, camera resectioning in broadcasted material must rely solely on information encoded in the images and the knowledge of the geometry of the playing courts. Playing field lines are extracted mostly via Hough transform [234, 253, 274]. Bebie and Bieri use template matching to identify the lines in [22]. Advertisement board detection and matching is applied by Yoon et al. in [274] to track the camera at the sides of the playing field when not enough lines are visible. Assuming stationary cameras that are variable in pan, tilt and zoom only, a wire frame of the court can be tracked continuously by point correspondences [258, 69, 68, 240, 42, 88]. RANSAC [79] has been used to improve the robustness and the speed of the tracking to real-time [68, 240, 42, 88]. Gedikli [88] additionally exploits optical flow measurements to track the camera even when no lines are visible.

We follow the approach of Gedikli [88] to estimate the camera parameters, including a fixed radial lens distortion. The estimated coordinate transformation is therefore non-linear, and represents the true mapping more precisely.

3.3 Foreground Segmentation

Foreground segmentation in sports videos can be categorized according to three methods: background subtraction (mostly used for static cameras), color segmentation and segmentation by motion. This preprocessing step takes up a significant amount of computational time since it must be applied to all of the massive image data gathered by each camera. The last step in segmentation usually includes cleaning the segmented image by morphological operations like opening, closing or erosion [122]. A survey of image thresholding techniques and a quantitative performance evaluation thereof is given in [220].

3.3.1 Background subtraction

For static cameras, time differencing and background subtraction is widely used in sports analysis [126, 122, 111, 151, 90]. Background subtraction builds up a model for the background and usually thresholds the difference between the current image and this model to extract the foreground. Algorithms differ mainly in the complexity of the model, ranging from a preliminary gathered background image (often called reference frame) to Gaussian color models for each pixel [229] to joint color distributions for the whole image (i.e. extracted by PCA). Piccardi [190] as well as Parks and Fels [185] give surveys about recent research in background subtraction methods. Since shadows usually reflect a change in

the color values of their corresponding pixels, they are recognized as foreground even if not intended to. Figueroa et al. [78] recover the background by omitting shadows of the player.

The idea of background subtraction was also ported to dynamic cameras by stitching together images over time (often called mosaicing), each usually showing only a fragment of the field. This constructs a complete mosaic image, which in turn can be used as background model [20, 161, 274, 136, 206]. The technique is not restricted to color or intensity values. For instance, Müller Junior and Ricardo de Oliveira Anido applied background subtraction with reference frame based on the gradient image of the scene [122].

Broadcasted video footage poses an additional processing burden, because frequent overlays decrease the effectiveness of background subtraction methods. They can be blocked and excluded from the image if detected as static regions or by template matching methods. The audience on the stands, which could also have a negative impact on background subtraction, can be masked based on the knowledge of all camera parameters. Cuts with a total change of perspective must be handled by adequate selection of the background model. This, however, is not a trivial task at all.

3.3.2 Color segmentation

Vandenbroucke et al. found the regions containing soccer players by adaptive color space segmentation in [247]. Gaussian mixture models for colors have been applied in several publications [178, 26, 89, 24]. Spagnolo et al. learn the color classes for each team in an unsupervised manner in [227]. Instead of segmenting the players by color directly, some methods [2, 109, 162] detect the players as holes by connected component analysis on the playing field that is found by a trained color histogram for the dominant color of the green.

Renno et al. [205] learns the color of shadows by an unsupervised learning procedure and is therefore able to exclude these shadows from a foreground image.

Since the jersey colors of the different teams and the color of the playing field are designed to be distinguishable by the referees and the audience, segmentation by color should be well suited for the task of foreground segmentation. For outdoor sports, however, changing light conditions and shadows due to clouds and stadium architecture complicate algorithms based on color only. The handling of these cases require online learning. Also, commonly worn white jerseys intersect with the color of the court lines. Artifacts based on video compression and blurred motion further exacerbate the problem of detecting players. Special care must be taken for handling advertisement boards which could be of same

color than player jerseys.

3.3.3 Segmentation by motion

Rarely, segmentation by motion is used for foreground extraction [61, 90, 140, 102]. This method is based on the assumption of a consistent motion model of the background (which can also be no motion) and is therefore suited for static as well as dynamic cameras. The motion vectors for each pixel can be computed by the optical flow method by Lucas Kanade [167] or varieties thereof, which are all based on the consistency of color intensities over time. The RANSAC (Random Sample Consensus) algorithm [79] is often applied to handle outliers and improve robustness for the background motion estimation. All pixels whose motion vectors differ from the expected background motion are marked as foreground. Segmentation by motion is not very accurate since optical flow methods are mostly imprecise and not robust against image quality losses.

3.3.4 Our approach for soccer

Our approach tailors foreground segmentation to outdoor sports on grass fields. Since the playing field constitutes a large, homogeneous area, we use a local variance filter to segment the players as proposed in [27]. A small kernel (usually 3×3) is moved over the intensity (grayscale) image of the current frame and the variance in this rectangular area is computed. Player regions typically evoke high values since they differ significantly from the green and their texture contains irregularities in color. Although this approach is not immune to shadows of the players, the segmentation of light shadows is reduced if compared to typical background subtraction with color models. Since variance in the stands typically shows high local variance as well, the local variance filter is applied to the video image masked to the projected playing field (including an offset from the outer court lines) for additional speed-up. Different foreground segmentation approaches are applied to video frames captured by static as opposed to dynamic cameras. In both cases, the resulting foreground regions are delivered to registered information modules as run-length encoded (RLE) regions including their color information. This procedure reduces the bandwidth needs for transmission as well as the computational time for decoding the received data.

Static cameras

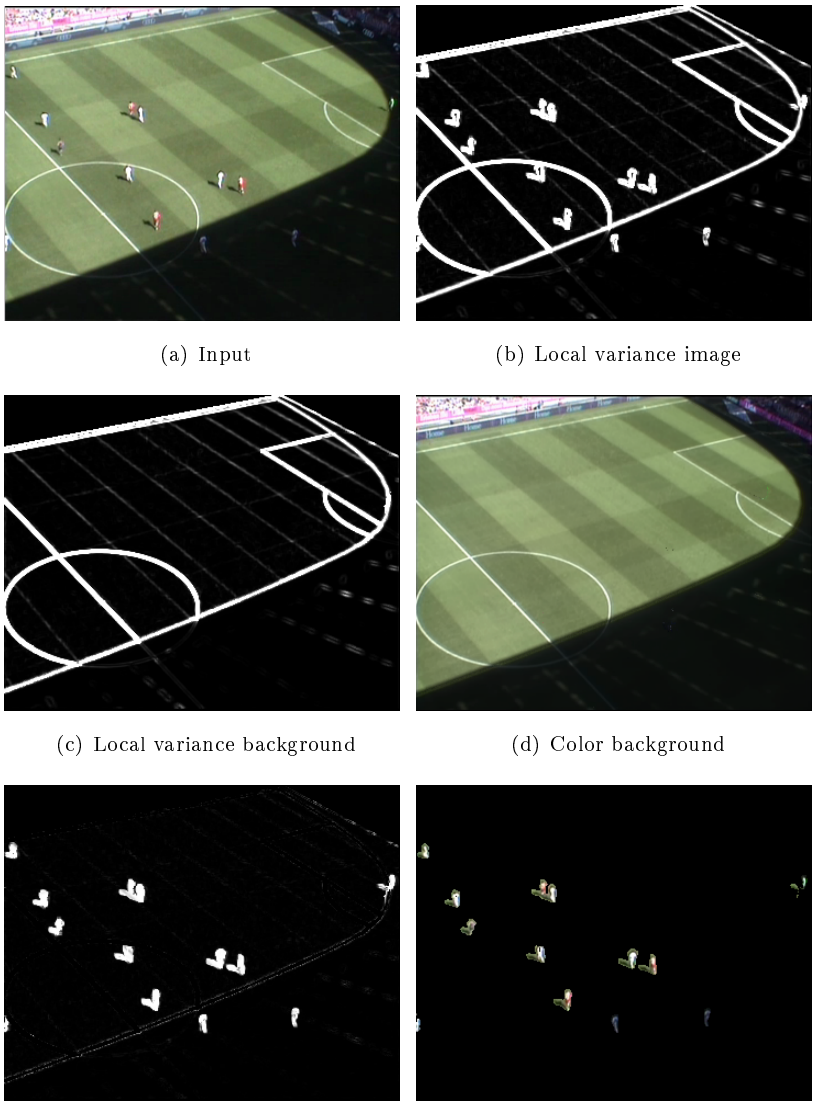
In case of static cameras, raw local variance image cannot be utilized directly since the court lines and markings as well as shadow and reflection contours arising from the stadium architecture induce high local variance as well. There-

fore, background subtraction is applied directly to the local variance image. We employ the technique of effective Gaussian mixture learning for each variance pixel, following the proposal by Lee [150]. Because positions of artifacts and lines in the image are constant or change slowly, they are incorporated into the background and therefore omitted in the result of the subtraction. Unfortunately, players passing the court lines are cut into several pieces. To handle this problem, we learn an additional color background, as seen in figure 3.3(d). The update and differencing of the background image is masked by the variance background image to detect changes on the lines only. The sparse variance background significantly reduces the computational demands of this usually expensive step. Finally, the resulting intensity image is binarized by applying a Gaussian blur followed by a thresholding operation. The blurring step is advantageous because it closes small holes in the regions. The complete foreground segmentation for static cameras is depicted in figure 3.3. Our approach shows sufficient accuracy while keeping the computational demand low. Four static cameras, providing images with dimension 704×576 downsampled to half of their size, are preprocessed in the described way on a single 2.5 GHz quad core computer in real-time with 30fps.

Dynamic cameras

In case of dynamic cameras, the variance is thresholded adaptively, taking only pixels with local variance larger than the mean of all values plus one standard deviation. Hard shadows as shown in figure 3.3(a) cannot be handled and remain as artifacts in the foreground. But this problem is not as severe as for static cameras, since dynamic cameras typically zoom in more closely, capturing a smaller area. They can therefore adapt their lens aperture to provide a more balanced illumination level, which avoids the problem.

The court lines, which also show up in the local variance image, are cleaned by projecting the virtual lines into the image based on the camera estimation by [88] and erasing these pixels. To prevent the erasure of pixels that belong to player regions intersecting the lines, we make use of the fact that the line width for soccer fields is fixed to 13 cm by official rules [76]. We delete pixels if and only if the number of pixels that have a high local variance and that are located on the normal to the projected court line is below a threshold determined by the projected line width. In other words, pixels are kept if the width of the line at that point is consistent with the forecast, otherwise an intersection with a player is assumed. The center circle, all arcs and even the court lines are approximated by line segments to take the non-linear camera transformation into account. Bresenham's line algorithm [35] is used to iterate over the normals



(a) Input (b) Local variance image

(c) Local variance background (d) Color background

(e) Final mask (f) Foreground

Figure 3.3: Segmentation for static cameras.

of the projected line segments and count the white pixels. The deletion of the pixels is done accordingly if they are below the threshold. The algorithm is visualized in figure 3.4.

Players, or parts of them that visually extend beyond the playing field, would be omitted due to the preliminary clipping of the video frame to the playing field. But even without the clipping step, the problem would persist since the boards with advertisements usually contain a high variability in color values and appear together with the player as a single bright region in the local

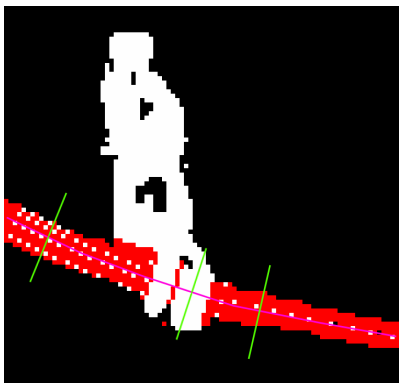
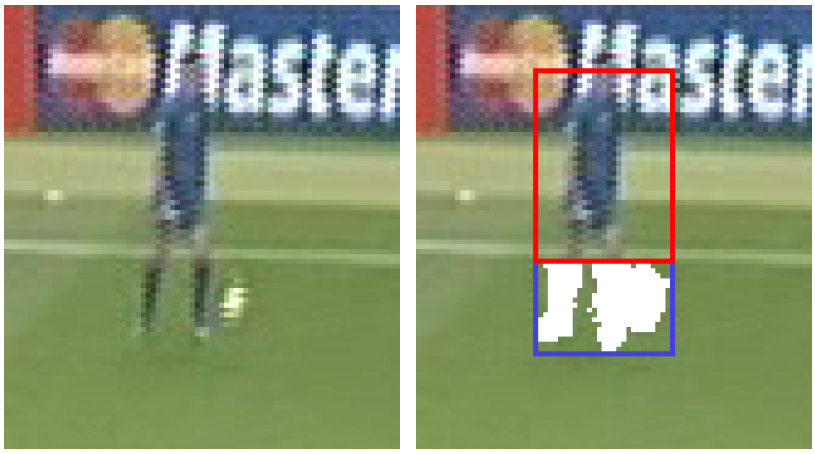


Figure 3.4: After smart line erasure, only white pixels remain in the final image. The selective line erasure works as follows: non-black pixels are counted along the normals (colored in green) of the projected line segments (colored in pink). If their number is below a threshold derived from the projection of the real line width of 13 cm, the pixels along the normal are deleted (colored in red). Due to discretisation, some artifacts (white pixels surrounded by red ones) remain in the image, but will be removed by morphological operations afterwards.

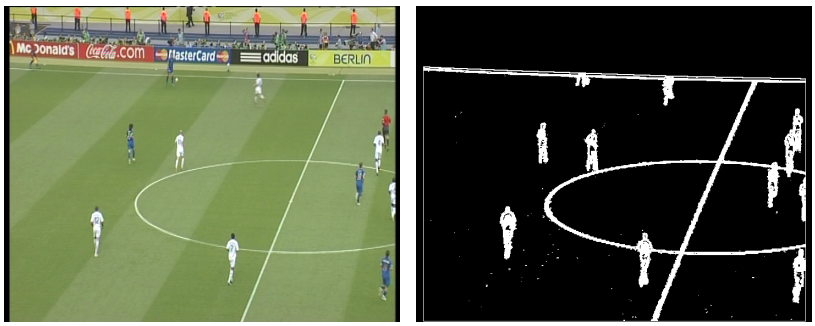


(a) Player at border (b) Detected foreground region

Figure 3.5: Players extending beyond the border of the playing field are handled in a special way. The bounding box of the region with high local variance intersecting the border (depicted in blue) is enlarged to match the projected height of a typical player (by the red rectangle).

variance image. We can make no assumptions about the appearance of the boards (especially as the displayed advertisements are often replaced or change frequently during a game) because they can even be the same color as the player

(see figure 3.5(a) for an example). To solve this problem, we enlarge the existing region to match the height of a typical player, if the line width at the outer lines exceeds the threshold analogously to the treatment of the court lines inside the playing field. The procedure is depicted in figure 3.5. For the final binary mask, the area under the enlarged bounding box is combined by a logical AND operation with the smaller region originating from the local variance if inside the smaller bounding box. The resulting mask can be seen in figure 3.6(d). These difficulties with players at the borders occur rarely in images captured by cameras with a high position and a steep viewing angle. Because this can be taken into consideration when control over the stadium set-up is given – in which case usually static cameras are employed –, this problem was not discussed for static cameras.



(a) Input (b) Local variance image

(c) Smart line erasure (d) Players at the border

Figure 3.6: Segmentation for dynamic cameras.

Morphological operations are applied as a final step on the binary mask to filter out noise induced by variations in illumination and the smart line erasure. The total process of an input frame is depicted in figure 3.6.

3.4 Broadcasted Material and Other Sources

Broadcasted material can be treated similarly to videos captured by dynamic cameras. In addition, cuts, action replays and slow motion episodes must be detected and the correct camera must be identified in the current episode. We rely on the work of Gedikli [88] to handle these non-trivial tasks.

Beside the plain image of the scene, broadcasted material provides a much richer source of information. Events, which each describe a special set of circumstances that could provide special insights into the game, can be detected in various ways. Recent research has focused primary on automatic detection in broadcasted television footage that exploits cinematic features [62, 153]. Motion vectors are used to find attack and defense situations in basketball [102]. In [265] Xu et al. give an overview of their work in semantic sports video analysis. They used audio keyword spotting and match reports in real-time from game logs on the web [266, 267]. Coldefy and Bouthemy [50] proposed an excited speech detection by modified, short-term energy in the audio channel. Babaguchi et al. [8, 9] detect events based on textual overlays (content description objects) and the transcript of speech data as subtitles (closed caption). Zhu et al. [278] extract route patterns (side-attack) and interaction patterns (dribbling-attack) in broadcasted soccer videos and webcasting text.

Several of these events give hints about player identity. The most obvious point is the announcement of the substitution of a player by another player of the same team. This information could be extracted from textual overlays as depicted in figure 3.7, from transcripts, audio commentary or real-time weblogs. Similar methods can be used to extract player identities for bookings or other penalties. Halin et al. [98] provide a technique for extracting text from overlays. The executor of a corner kick could be identified by its position in combination with the linguistic determination of the player's name.

(a) Substitution of a player (b) for a new substitute (c) or both on a single frame.

Figure 3.7: Substitutions signalised by overlays in broadcasted video can be detected as events and assist player identification.

In combination with ball tracking, penalties, passes, dribbling and shot

events can reveal the identity of the actors involved. However, a big problem is posed by the need of fine-grained temporal alignment. The same difficulties hold for audio keyword spotting in general, since it is not clear which player the commentator is talking about. Face detection, which is possible in close-ups only, can hardly be used in combination with a player's location. This marks both procedures as nearly impractical (but still possible) choices for player identification. Break detection, on the other hand, can be achieved more easily and is helpful for tracking because people other than players and the referee can enter the playing field solely in an adjourned game. In addition, events can act as a prior for positional identification based on the tactical player line-up. The certainty in player assignment at a kick-off, for example, is fairly high – in contrast to the assignment during a corner kick.

3.5 Player Localization

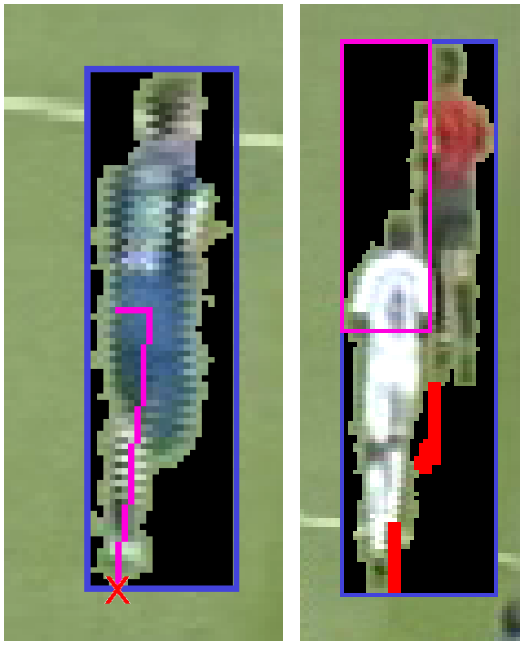
The task of localization consists in mapping of potential regions to locations in the image. So far, this transformation has mainly been done in a predefined deterministic fashion by computing agglomerative statistics of all positions inside the shape. Because the bounding box of the targets is primarily tracked in many projects, the locations of these targets can be easily deduced. Needham [179] utilizes learned shape templates to extract these bounding boxes. Alternatively, the mean shift method [117] has been applied to globally compute the positions (of all players) by identifying the modes of the region densities.

We extract the positions from regions containing single or multiple players differently. Firstly, the foreground regions are detected as connected components by following their external contours along region pixels in a 3×3 neighborhood. A single region found in that way can be described by so-called image moments: there are its centroid, its area, its rotation around the center and its minimum bounding box, which denotes the smallest rectangle containing the whole region. Because the shape of a human is not arbitrary, valid regions are constrained by their size as well as their area. We assume an average height of 1.80 meters and an aspect ratio of $\frac{1}{3}$ based on statistics of anthropometry. Considerably smaller areas are discarded or merged with player regions if they fall inside the bounding boxes of these to catch disconnected hands or feet.

Regions with appropriate area are assumed to contain a single player only. The centroid of such regions projected on the lower base vertex of the bounding box is taken as the position in image coordinates. This point can then be transformed to real world coordinates by the known homography of the recording camera. The localization for single players is depicted in figure 3.8(a).

As players often interact, their contours overlap and they appear as a single

big region. These merged regions are detected because they exceed the expected size of a typical athlete. The binary image of the big region is convoluted by a rectangular template of typical athlete size projected onto the image. This convolution computes the area of the region under the current template. For computational efficiency, integral images are used which reduce area computations to three additions and a single subtraction. Hypotheses for player positions are found as local maxima in the resulting convoluted image. Contour lines are fragmented in single saddle points with reasonable distance. This approach exploits the physical constraints induced by perspective occlusionges in a simple way. An example of this procedure is depicted in figure 3.8(b). Although this approach generates additional false positives, we prefer it over a more restrictive one since the tracking algorithm can better deal with surplus measurements than with missing positions.



(a) Single player (b) Multiple occluding players

Figure 3.8: Single players are localized as the lower intersection of the main axis (colored in pink) and the bounding box of the corresponding region (colored in blue). The red cross marks the computed position in the image, which is transformed to world coordinates in turn. Multiple occluding players form a single big region. Positions (colored in red) are extracted as local maxima of the convolution of the region with a rectangular template (colored in pink) representing the size of a typical player.

3.6 Conclusions

In this chapter we surveyed different kinds of sensors and their use for player tracking. With the focus on cameras, the main methods of foreground segmentation have been reviewed. These can be applied to segment players in sports video footage. We presented a novel approach based on the local variance image of a video frame, exploiting the homogeneity of the playing field. The foreground is segmented in this local variance image by background subtraction for static cameras and adaptive thresholding for dynamic cameras. We showed two different approaches of handling players who cross court lines by selective erasure of the lines and color background subtraction. The methods presented here are nonparametric in nature and exhibit real-time performance, while being fairly robust in handling different illumination conditions including shadows and reflections.

Extracting player locations from regions is handled differently for shapes containing single or multiple players. The number of players inside a region is estimated based on the expected area of a standard athlete. Shape-template matching is applied to extract potential positions of multiple players inside a large area.

Chapter 4

Multi-target Tracking

The potential player locations gathered in every frame can be filtered by exploiting temporal consistency to provide a better estimate of the current positions. Methods that try to find a good way to filter these data are defined as belonging to the multi-target tracking discipline. The computational problem of multiple target tracking consists in the estimation of target trajectories from cluttered and noisy measurements arriving over time at discrete time points. This task is illustrated in figure 4.1. Tracking typically includes a prediction and an update step. The former is based on a motion model of the targets to form a common ground with the current measurement scan. The latter combines the predicted state of the targets with associated measurements of the current scan.

In this chapter we introduce the Rao-Blackwellized Resampling Particle Filter (RBRPF) as a novel multi-target tracking method suitable for real-time performance. Our approach is designed to track a fixed number of targets because this is the case in almost all sports. We solve the data association problem, which is part of the update step, by sampling individual associations according to their likelihood. This approach reduces the tracking problem to several single-target tracking problems. Single-target tracking is achieved in turn by the popular Kalman filter, which constitutes an optimal estimator (in the sense of maximum a posteriori) and which has been used and verified extensively in the field. The Rao-Blackwellized Resampling Particle Filter belongs to the class of Monte Carlo Joint Probabilistic Data Association Filters (MCJPDAF). In addition to its special applicability for player tracking in sports video, it provides a general approach for multi-target tracking.

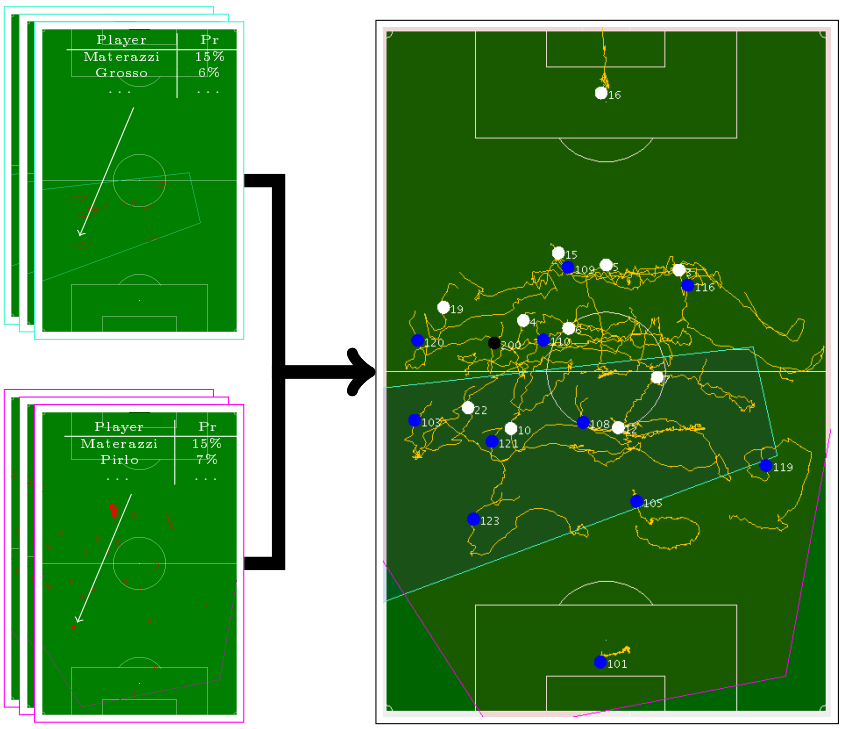


Figure 4.1: Multi-target tracking filters the cluttered and noisy measurement scans, which suggests potential player locations, to form consistent trajectories over time.

4.1 Related Work

Tracking has been researched since the 1950s, primarily for air and ocean surveillance by radar. Bar-Shalom and Fortmann defined "[t]racking [as] the processing of measurements obtained from a target in order to maintain an estimate of its current state [...]" [13]. Tracking predominantly follows the Bayesian approach as described in section 2.3.1. Chen gives a thorough tutorial on Bayesian filtering in [47]. The tracking approaches can be differentiated into single and multi-target tracking. Both fields are reviewed separately in the next sections.

4.1.1 Single-target tracking

Single-target tracking faces the problem of how to fuse different measurements with the predicted state of a single target to estimate its trajectory over time correctly. Arulampalam et al. [6] provide a detailed tutorial over the field.

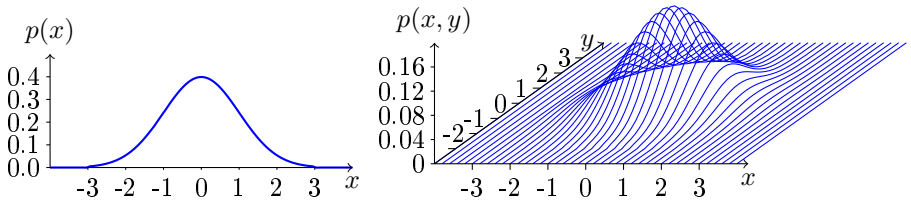
Kalman filter

The Kalman filter [123, 124] has been used extensively for single-target tracking and will continue to be used in the future. The algorithm constitutes an optimal maximum a posteriori estimator if the state prior and measurement noise follow a Gaussian distribution and the process as well as the measurement model are linear (see section 2.3.1 for an explanation of the models). An optimal Bayesian solution solves the problem of recursively calculating the exact posterior density; if an algorithm deduces this solution, it is called an optimal algorithm.

The multivariate Gaussian or Normal distribution is given by

$$p(x) = \mathcal{N}(x; m, V) = \frac{1}{\sqrt{|2\pi V|}} e^{-\frac{1}{2}(x-m)'V^{-1}(x-m)} \tag{4.1}$$

with mean $m = \bar{x}$ and covariance V. The univariate (one-dimensional) and bivariate (two-dimensional) standard normal distributions are depicted in figure 4.2, where the term "standard" implies that the mean equals zero in every dimension and the covariance matrix is the identity matrix. The Gaussian distribution has the nice property that it is closed under linear and affine transformations. The central limit theorem states that the sum of a sufficiently large number of independent random variables, each with finite mean and variance, approaches a Gaussian normal distribution in the limit.



(a) univariate standard normal distribution

(b) bivariate standard normal distribution

Figure 4.2: The normal distribution also called Gaussian with mean m and (co)variance V.

The Kalman Filter constitutes a set of recursive equations that can be differentiated in a predictive and an update step (see [260] for a good tutorial). The computation of the predicted state is reduced to a matrix product since the process model of equation 2.11 is assumed to be linear:

$$\hat{x}_k = F_k x_{k-1} + v_k \tag{4.2}$$

and

$$\hat{V}_k = F_k V_{k-1} F_k' + Q_k. \tag{4.3}$$

As the measurement model of equation 2.8 is also linear, it can be written as a matrix H_k, too. The state update equation reads as

$$x_k = \hat{x}_k + \left(\hat{V}_k H_k' \left(H_k \hat{V}_k H_k' + R_k\right)^{-1}\right)(z_k - H_k \hat{x}_k) \tag{4.4}$$

and the corresponding covariance update equation is

$$V_k = \hat{V}_k - \hat{V}_k H_k' \left(H_k \hat{V}_k H_k' + R_k\right)^{-1} H_k \hat{V}_k. \tag{4.5}$$

The second summand of the state update of equation 4.4 constitutes the filter gain multiplied with the innovation (or measurement residual). The filter gain controls the influence (innovation) of the measurements versus the prediction on the final estimate based on the proportion of their (un)certainties. Several (mathematically equivalent) versions of the covariance update equation 4.5 have been proposed showing differences in numerical stability and computational performance. The equations adapted for the inverse covariance are called Information Filter.

Extensions to the Kalman filter

Several extensions for the non-linear case exist: the Extended Kalman Filter (EKF) [118], which linearizes the model by Taylor expansion, and the Unscented Kalman Filter (UKF) [121], which transforms the so called sigma points and reconstructs the transformed Gaussian, are widely used. Figure 4.6 illustrates these methods. Iterated versions of these extensions exist to improve the robustness of the linearization.

For maneuvering targets, a multiple-model approach was originally proposed by Magill [169]. Bar-Shalom introduced the suboptimal Probabilistic Data Association Filter (PDAF) for handling clutter (measurements generated by noise rather than the target) and false alarms by a parametric and a nonparametric variant [15, 16]. The Optimal Bayesian Filter (OBF) [223] utilizes association probabilities based on the entire measurement sequence observed so far (in contrast to only the latest measurements for the PDAF). Suboptimal (N-scan-back) algorithms soften the dependence on former measurements by a Markov assumption of order N to decrease the high computational demand.

Grid-based methods discretize the state space into cells and utilize the Viterbi or Baum-Welch algorithm [197] to calculate the maximum a posteriori estimate of the path through these cells. They provide an optimal solution iff the underlying state space is really discrete in its nature. This is not the case for position tracking and these methods are rather inefficient if one wants to cover a large field with sufficient resolution.

Particle filter

The particle filter is a tracking method that can handle arbitrary process and measurement models and that approximates arbitrary probability distributions. It is also known as bootstrap filtering, CONDENSATION [116] or survival of the fittest. It has been successfully applied to various tracking tasks; its simple implementation as well as appealing speed and performance explain the wide usage in the field.

The filter instantiates a sequential Monte Carlo method that solves problems by statistically correct simulations. In *Sampling Importance Sampling* (SIS) particle filtering, the posterior probability density function is approximated by a weighted sum of N_p random samples x_k^i (known as particles)

$$p\left(x_k|z_{1:k}\right) \approx \sum_{i=1}^{N_p} w_k^i \delta\left(x_k - x_k^i\right) \tag{4.6}$$

with δ denoting the Dirac function which is one at the origin and zero elsewhere. The so called importance weights are normalized $\sum_i w_k^i = 1$ such that the approximation forms a proper pdf (c.f. equation 2.2).

The Particle Filter consists of a sampling and a filtering step. Particles for the next time step are sampled according to a proposal $q(.)$ called an importance density. The new weights depend on the choice of the importance density and are computed during the filtering step. The recursive equation for the weights up to a normalizing constant factor reads therefore as

$$w_k^i \sim w_{k-1}^i \frac{p\left(z_k|x_k^i\right) p\left(x_k^i|x_{k-1}^i\right)}{q\left(x_k^i|x_{k-1}^i, z_k\right)}. \tag{4.7}$$

If N_p tends to infinity, the approximation of equation 4.6 approaches the true posterior density in the limit.

Doucet [59] showed that the optimal importance density function that minimizes the variance of the true weights conditioned on x_{k-1}^i and z_k is

$$q_{opt}\left(x_k|x_{k-1}^i, z_k\right) = p\left(x_k|z_k, x_{k-1}^i\right) = \frac{p\left(z_k|x_k, x_{k-1}^i\right) p\left(x_k|x_{k-1}^i\right)}{p\left(z_k, x_{k-1}^i\right)}. \tag{4.8}$$

This optimal importance density would require sampling directly from the real distribution, making the weights redundant. It can hardly be achieved in practice. In the vast majority of applications in computer vision, the prior $p\left(x_k|x_{k-1}^i\right)$ is used as importance density leading to a weight update according to the likelihood $w_k^i \sim w_{k-1}^i p\left(z_k|x_k^i\right)$.

As can be seen in equation 4.6, the density at a specific point is calculated as the sum of the weights of all particles at the same point. The same density can be represented by a single particle with a high weight or multiple particles at

the same location with proportional lower weights. After a few iterations, the SIS particle filter suffers from the degeneracy phenomenon, where one particle will have a high and the rest only negligible weights. Resampling utilizes the mentioned relationship of the number and weights of particles to handle the degeneracy problem by eliminating particles with small weights and duplicating the ones with large weights. The resampling step of the SIR particle filter generates a new particle set $\left\{x_k^i\right\}$ by resampling (with replacement) N_p times from the current particle set and initializing the new weights with $\frac{1}{N_p}$. The SIR method is depicted in figure 4.3.

SIR($\left\{x_{k-1}^i, w_{k-1}^i\right\}_{i=1}^{N_p}, z_k$):

1. Draw $\left\{x_k^i\right\}_{i=1}^{N_p}$ from importance density $q\left(x|x_{k-1}^i, z_k\right)$
2. Calculate importance weights $w_k^i = w_{k-1}^i \frac{p\left(z_k|x_k^i\right)p\left(x_k^i|x_{k-1}^i\right)}{q\left(x_k^i|x_{k-1}^i, z_k\right)}$
3. Normalize weights $\tilde{w}_k^i = \frac{w_k^i}{\sum_{i=1}^{N_p} w_k^i}$
4. Resample with replacement $\left\{x_k^j\right\}_{j=1}^{N_p}$ from $\left\{x_k^i\right\}_{i=1}^{N_p}$, where $\Pr\left(x^j = x^i\right) = \tilde{w}_k^i$
5. Return $\left\{x_k^j, N_p^{-1}\right\}_{j=1}^{N_p}$

Figure 4.3: One iteration of the SIR particle filter.

The Mixture Kalman Filter (MKF) by Chen and Liu [46] is a special case of a particle filter that tracks mean and covariance of Gaussians for the target position by using Kalman filters inside each particle instead of the position itself. The technique of replacing the state by parameters of a model describing the entire state pdf is known as marginalization or Rao-Blackwellization [39]. Its use is founded on the Rao-Blackwell theorem, which states that an estimate based on sufficient statistics (an appropriate model) always improves the plain estimate in terms of its variance. Rao-Blackwellization often improves the computational efficiency of particle filters since (parts of) the sampling can be replaced by an analytical solution. Khan et al. [133] proposed a particle filter that was Rao-Blackwellized with an appearance model by computing probabilistic PCA.

4.1.2 Multi-target tracking as data association problem

The subfield of multi-target tracking was introduced by Sittler in 1964 [224]. In addition to multiple single-target problems, multi-target tracking predom-

inantly copes with the data association problem of associating measurements with the correct target. Data association algorithms are constructed according to either a deterministic or a probabilistic model. The former determines an association and treats it as if it were certain and the latter utilizes the probabilities of the different possible associations. Multiple single-target trackers are likely to coincide especially if interactions of targets occur. Most data association methods thus rely on some kind of exclusion principle. We will review the different multi-target tracking approaches in view of how they solve the data association problem.

Kalman filter based approaches

Bar-Shalom and Fortmann [13] survey the classical Kalman filter based approaches in multi-target tracking. Multiple Kalman filters are applied for soccer tracking in [17] resolving occlusions of multiple players by a rule-based method that recognizes differences between removable and intrinsic ambiguities. The Nearest-Neighbor Standard Filter (NNSF) associates targets to the closest measurement and was state-of-the-art until the early 1970s, but it is still used for tracking sport players, for instance in [268]. The Joint Probabilistic Data Association Filter (JPDAF) [12, 14] is an extension of PDAF to multiple targets by evaluating the probabilities of the joint association events. The Joint Likelihood Filter (JLF) was proposed as an advancement of the JPDAF for visual tracking [200] but is seldom used. The Multiple Hypothesis Tracker (MHT) [203, 204] lifts the Optimal Bayesian Filter to track multiple targets by constructing an association tree for the history. Because it suffers from even higher computational burdens as the OBF, N-backscan and pruning approaches or finding only the k-best assignments based on the Hungarian method [54] are used in facto as [89] in soccer, for example. All methods described so far consider only feasible associations. This means that measurements can be assigned to a maximum of a single target and that each target exhibits no or only a single measurement. Probabilistic Multiple Hypothesis Tracking (PMHT) [231, 230] joins the advantages of JPDAF and MHT, also allowing merged measurements for a single target, which are then selected by expectation maximization (EM). The selection of associations by the Viterbi algorithm, analogous to the grid-based methods, was proposed for ball tracking in soccer [271].

Particle filter based approaches

While the former methods are all based on Kalman filters, Monte Carlo approaches constitute the second class of multi-target tracking algorithms. Clustering of the particles circumvents the data association problem for soccer

[248, 77] or hockey players [182] by maintaining occluded targets as single objects. It thus results in multimodal tracking as Mixture of Particle Filters (MPF). Since identities are not preserved, identification of individuals after split is encouraged via AdaBoost [182] or learned decision trees [226]. Kristan et al. [142, 143] exploit local motion of the players in indoor sports computed by optical flow to separate the particle filters for each single target. The NNSF has been ported to particle filters by nearest neighbor association and applied to the sports domain in [36, 155]. A Monte Carlo variant of the JPDAF (MCJPDAF) replaces the underlying Kalman Filters of the JPDAF with Particle Filters for each target to handle Non-Gaussian distributions, the data association is sampled in the resampling step according to the joint probability and provides a fast way of finding the most likely assignments (c.f. [217, 127]). A joint Markov Random Fields (MRF) particle filter was proposed in [132]. This penalizes overlapping of particles for different targets. The Variational Particle Filter [119] utilizes the probabilistic data association scheme with the weak exclusion principle of PMHT and applies variational inference rather than sampling from the prior as the proposal distribution. A Particle filter with joint probabilities for the associations was proposed for soccer tracking by [90] handling the exclusion via the weights rather than by selection of the associations. Rao-Blackwellized Monte Carlo Data Association (RBMCDA) [213] employs the MCJPDAF idea modeling the individual particles as Gaussians.

MCMC based approaches

Monte Carlo Markov Chain (MCMC) [93] as another Monte Carlo approach is distinct from particle filtering by its underlying idea and gains in popularity recently. Instead of representing the posterior probability distribution by a number of samples in parallel, one sample is continuously changed by proposals and its trace approximates the distribution. A stochastic automaton is created where each transition inside the state space is accepted by a specially designed acceptance ratio such that the stationary distribution equals the posterior. The transitions (also called proposals) and the acceptance ratios must be ergodic, which means that every point in the state space can be reached, while wandering through the state space, and no dead-ends capturing the current state exist. Starting at an arbitrary state, the Markov Chain is usually run for a burn-in period to "forget" this initial state. The frequencies of the visited states after the burn-in form the desired probability distribution. Monte Carlo Markov Chain Data Association (MCMCDA) was proposed in [181]. The joint Markov Random Fields (MRF) particle filter was ported to MCMCDA in [134]. A real-time variant using Rao Blackwellization exists [135] and allows the tracking of

variable dimensions inside the MCMC framework. Bardet et al. [18] parallelized the originally sequential approach. MCMCDA was also used to track soccer players [162, 87].

Variable number of targets

Most methods for multi-target tracking have been extended to track not only a fixed but a variable number of targets. This has been achieved by assigning unmatched measurements to virtual targets and init new tracks (called birth) if the associated probability gives enough reason for their existence over time. If the probability for a target drops below a threshold, the target is removed from further tracking (called death).

4.2 Basic Idea

The Rao-Blackwellized Resampling Particle Filter (RBRPF) forms a recursive estimator of the complete formations including all player positions. The previous estimate is advanced to the time of the current measurement scan by predicting the locations according to a given motion model. Particles for the current estimate are gathered from the previous ones by sampling associations between the predicted formations and the current measurements at a rate proportional to the former weights and fusing the corresponding positions in an optimal way (max-likelihood). The probability densities are determined by the frequencies of the samples that resulted in the same association, as well as the likelihood of this association. The RBRPF is sketched in figure 4.4.

Stated more technically, the Rao-Blackwellized Resampling Particle Filter constitutes a SIR particle filter (c.f. figure 4.3) as joint tracker in the product multi-object state space. The filter is Rao-Blackwellized by tracking mixtures of Gaussians instead of the plain positions. Importance sampling is split into prediction (which is solved analytically), sampling of associations and fusion of predicted targets and measurements according to the sampled association (also solved analytically). A measurement is assigned to a single target at max, while the maximal number of measurements assigned to the same target can be constrained. The fusion of identity evidence is achieved via the association likelihood. Memoization and smart deterministic resampling improve the efficiency of RBRPF, while the use of negative information gains performance. The complexity of RBRPF is linear in the number of particles, targets and measurements. The Rao-Blackwellized Resampling Particle Filter algorithm is depicted in figure 4.5. We will detail the algorithm in the next sections.

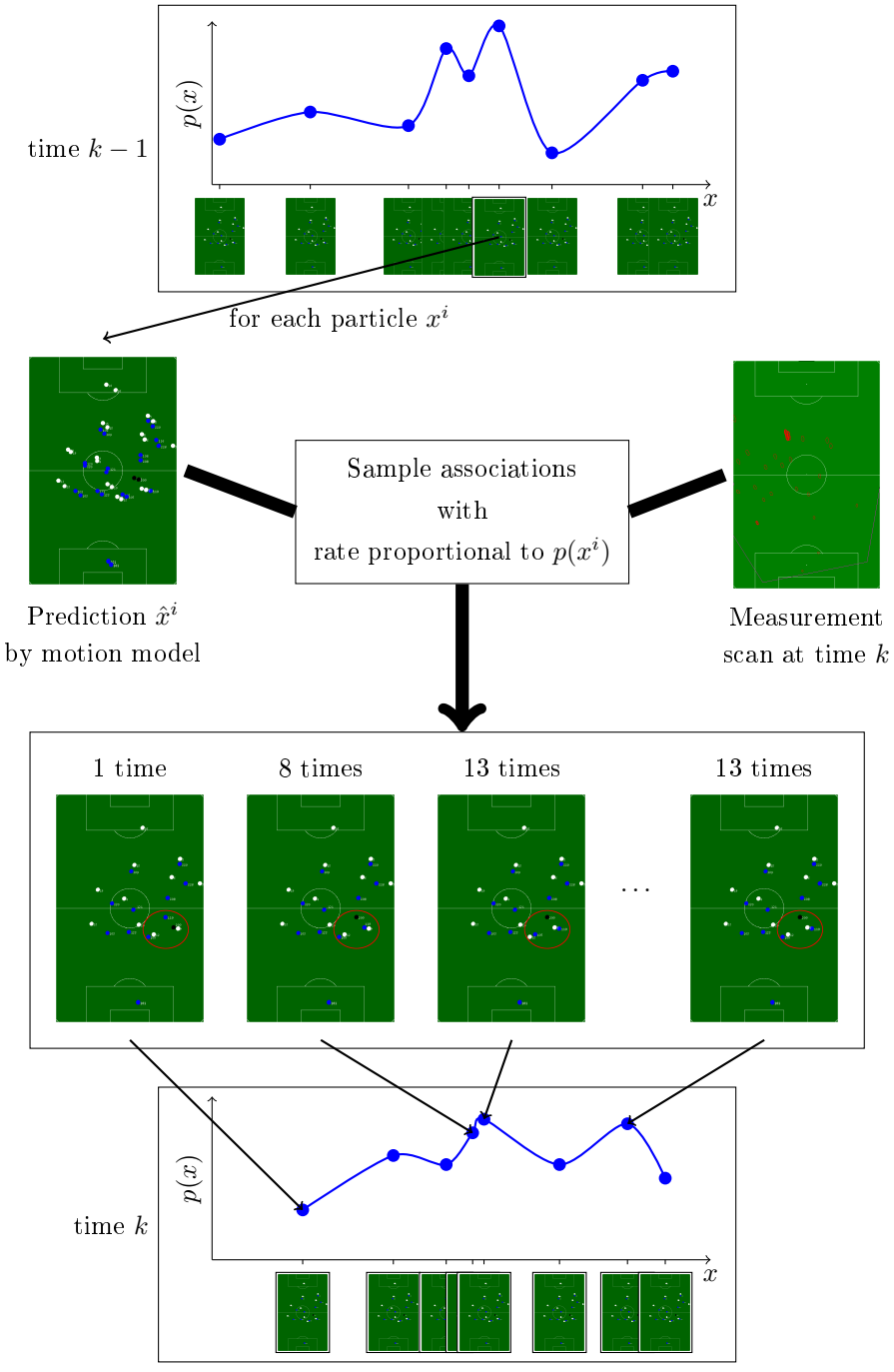


Figure 4.4: The Rao-Blackwellized Resampling Particle Filter (RBRPF) at a glance.

RBRPF($\{x_{k-1}^m, w_{k-1}^m\}_{m=1}^{N_p}, z_k$):

1. $i = 0$
2. FOR $m = 1 : N_p$
3. Draw prediction $\hat{x}_k$ analytically according to sec. 4.6.1
4. $\Gamma \leftarrow \emptyset$
5. FOR $j = 1 : \lceil N_p w_{k-1}^m \rceil$
   (Resampling according to sec. 4.5)
6. Draw association J_k according to sec. 4.6.2
7. IF $J_k \notin \Gamma$
8. $\Gamma \leftarrow \Gamma \cup \{J_k\}$; $i \leftarrow i + 1$; $N_i \leftarrow 1$
9. Draw x_k^i by fusing $\hat{x}_k$ and z_k given J_k
   according to sec. 4.6.3
10. Calculate unnormalized importance weight w_k^i
    according to sec. 4.7
11. ELSE
12. $N_i \leftarrow N_i + 1$
13. $N_p \leftarrow i$
14. Normalize weights $w_k^i \leftarrow \frac{N_i w_k^i}{\sum_{i=1}^{N_p} N_i w_k^i}$
15. Return $\{x_k^i, w_k^i\}_{i=1}^{N_p}$

Figure 4.5: One iteration of the proposed Rao-Blackwellized Resampling Particle Filter (RBRPF).

4.3 Assumptions

Our approach is designed for a fixed number of targets, since the number of players seldom varies during a game (opposed to their visibility) and in such cases, the change (e.g. due to penalty) is signaled and can be observed. For the replacement of players, the identity of the new target is given anyway. The restriction to a fixed number of targets does not imply that the tracked targets may be extended or reduced on the fly; it just states that the extension or reduction is not done automatically but must be initiated externally. The

proposed method can easily be extended to handle a variable number of targets as described in section 4.1.2 or [213], but this extension would be misleading in the sports domain, where the number of targets is known a-priori.

We assume that the individual state pdf of each player (his location) can be modeled sufficiently by a Gaussian distribution, not taking the ambiguity error into consideration. This is common practice and is presumed by most multi-target approaches (at least for the initialization). Additionally, only the two-dimensional position on the playing field is estimated since the 3D position can hardly be extracted from sports video images and would be estimated imprecisely.

Our approach is explained for a process model (or motion model as it is called in position tracking) that is linear Gaussian with constant velocity. Despite the fact that this is not true for the complex motions in sports, it is a sufficient approximation for ordinary frame rates of sports videos (around 25 frames per second) since the distance a player can travel in 40 milliseconds is reasonably small and can be approximated with a linear model. For cases where a non-linear model is essential, the proposed algorithm can easily be extended by the suboptimal approaches that are in use for the Kalman filter like EKF and UKF (c.f. figure 4.6). This extension can be done analogously to the linearization of the measurement model, which will be described.

The measurements are observed in sweeps (possibly multiple times per time step) and we expect no out-of-sequence measurements. These would have been dropped to preserve real-time processing anyway. The measurements of one sweep are assumed to be conditionally independent from other sweeps given the positions of the players. Our method states the weak exclusion principle of PMHT, where multiple measurements per target are allowed but each measurement of a sweep results from a single target only. This assumption is made by the most tracking algorithms to reduce the association space and holds – in our opinion – also in reality for visual sensors. The objection that interacting targets produce merged measurements due to imperfect segmentation as stated in [135] is not conclusive. It should be solved by improving segmentation because measurements that occlude other targets are still caused by a single target.

4.4 State Space

The state at time-point $k \in \mathbb{N}_0$ that is estimated by the proposed particle filter aggregates the individual states of all players j explicitly as the stacked vector

$$x_k = (x_{k,j})\ j = 1, \ldots, N_t \tag{4.9}$$

with $N_t \in \mathbb{N}$ denoting the total number of targets. The state of each player j is Rao-Blackwellized by supposing it to be normally distributed (see equation 4.1) around a mean m_j with covariance V_j

$$x_{k,j} \sim \mathcal{N}(m_{k,j}, V_{k,j}). \tag{4.10}$$

We track the two-dimensional position and velocity of each player on the playing field; the z-component of the true three-dimensional location in space is omitted for performance reasons. The state space for a single target can therefore be written as

$$m_{k,j} = (x_{pos}, y_{pos}, \dot{x}_{pos}, \dot{y}_{pos})' \in \mathbb{R}^4 \tag{4.11}$$

with a 4×4 real-valued symmetric positive (semi)definite covariance matrix $V_{k,j}$ (c.f. equation 2.18).

The posterior is approximated following the particle filter equation 4.6 by a set of $N_p \in \mathbb{N}$ weighted particles $\left\{x_k^i, w_k^i\right\}_{i=1}^{N_p}$. Although every player is assumed to be Gaussian, the posterior probability distribution of the particle filter describes a multimodal distribution that equals a mixture of Gaussians similar as for Mixture of Kalman Filters [46].

4.5 Resampling Step

We change the order of the typical SIR filter by bringing the resampling step forward. This is mathematically equivalent to the common case, but gives our approach a great gain in computational efficiency.

Our resampling method is similar to residual resampling [163] and replicates each particle deterministically N_i times (N_i can be zero) according to its weight with

$$N_i = \lceil w_{k-1}^i N_p \rceil. \tag{4.12}$$

Particles with larger weights thus allocate more particles in the next time step, while particles with small weights are discarded. Since the resulting number of resampled particles $\hat{N}_s = \sum_{i=0}^{N_p} N_i$ does not necessarily equal N_p, the proposed RBRPF constitutes a particle filter with a variable number of samples. This is rather unusual for SIR filters but there is no theoretical reason prohibiting doing so. The importance weights of the resampled particles are set equal to $\hat{N}_s^{-1}$. As we will see later in section 4.8 this will not be the final weights and particles since some of the samples can be joined again.

Independent of the proposed method, Särkkä et al. suggested in [213] future improvements for their RBMCDA approach, which are similar to ours:

> "By tuning the resampling algorithm and possibly changing the order of weight computation and sampling, accuracy and computational efficiency of the algorithm could possibly be improved [73]. An important issue is that sampling could be more efficient without replacement, such that duplicate samples are not stored. There is also evidence that in some situations it is more efficient to use a simple deterministic algorithm for preserving the N most likely particles. In the article [195] it is shown that in digital demodulation, where the sampled space is discrete and the optimization criterion is the minimum error, the deterministic algorithm performs better."

4.6 Sampling Step

The innovation of the proposed RBRPF algorithm is the choice of the importance density which differs substantially from the commonly used prior $p\left(x_k|x_{k-1}^i\right)$. The proposed importance sampling is composed of a prediction, an association and a fusion step. After the last estimate is projected to the time of the current measurement sweep, associations J between the measurements and predicted positions are sampled proportional to their likelihood. The resulting position sample is evaluated as the minimum variance fusion of the sampled associations.

This approach can be mathematically derived. Based on the total probability theorem of equation 2.12 and the assumptions of section 2.3.1, we can transform the optimal importance density of equation 4.8 to

$$q_{opt}\left(x_k|x_{k-1}^i, z_k\right) = \int p\left(\hat{x}_k^i|x_{k-1}^i\right) p\left(J_k|\hat{x}_k^i, z_k\right) p\left(x_k|J_k, \hat{x}_k^i, z_k\right) dJ_k \qquad (4.13)$$

with associations J_k and predicted state $\hat{x}_k^i$. The predicted state $\hat{x}_k^i$ is often referred to as the a priori state.

The integral of equation 4.13 is actually a sum since the associations constitute a discrete and finite set. We propose an approximative importance proposal density defined on the optimal fusions according to the associations J_k only

$$q\left(x_k|x_{k-1}^i, z_k\right) = \sum_{J_k} p\left(\hat{x}_k^i|x_{k-1}^i\right) \Pr\left(J_k|\hat{x}_k^i, z_k\right) \delta\left(x_k - \arg\max_x p\left(x|J_k, \hat{x}_k^i, z_k\right)\right), \qquad (4.14)$$

where δ denotes the Dirac function. This is reasonable since the probabilities for fusions other than the optimal one would be lower and therefore approximately negligible.

4.6.1 Prediction step

The first step of sampling from $p\left(\hat{x}_k^i | x_{k-1}^i\right)$ constitutes the prediction of the states of all particles i according to a constant velocity model. Since this model is linear Gaussian and Gaussians are closed under affine transformations, the sampling according to the predicted state distribution can be solved analytically.

The constant velocity model assumes no acceleration, but since the velocity can undergo small changes over time t, it is modeled by the mutually independent white zero-mean Gaussian noises $\tilde{v}(t) \sim \mathcal{N}(0, \tilde{q})$ and $\breve{v}(t) \sim \mathcal{N}(0, \breve{q})$ for the 2-dimensional case as

$$\dot{x}_{pos}(t) = \dot{x}_{pos}, \tag{4.15}$$

$$\ddot{x}_{pos}(t) = \tilde{v}(t), \tag{4.16}$$

$$\dot{y}_{pos}(t) = \dot{y}_{pos}, \tag{4.17}$$

$$\ddot{y}_{pos}(t) = \breve{v}(t). \tag{4.18}$$

A noise $v(t)$ is called white iff it has zero mean $E[v(t)] = 0$ and its value at a specific time is statistically independent of the value at any other time also written as $E[v(t)v(\tau)] = q\delta(t-\tau)$, i.e. it is completely unpredictable.

We process measurement sweeps observed at discrete time points with the time difference $\Delta t_k = t_k - t_{k-1}$ between the (not necessarily equidistant) time points k and $k-1$. The process model of equation 2.11 for each player j can now be written as

$$\hat{m}_{k,j}^i = F_k m_{k,j-1}^i + v_{k-1} \text{ with } F_k = \begin{pmatrix} 1 & 0 & \Delta t_k & 0 \\ 0 & 1 & 0 & \Delta t_k \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{4.19}$$

where hat denotes the predicted state. The white process noise v_{k-1} between $k-1$ and k is written as

$$v_{k-1} = \begin{pmatrix} \int_0^{\Delta t_k} (\Delta t_k - \tau)\, \tilde{v}\,(t_{k-1} + \tau)\, d\tau \\ \int_0^{\Delta t_k} \tilde{v}\,(t_{k-1} + \tau)\, d\tau \\ \int_0^{\Delta t_k} (\Delta t_k - \tau)\, \breve{v}\,(t_{k-1} + \tau)\, d\tau \\ \int_0^{\Delta t_k} \breve{v}\,(t_{k-1} + \tau)\, d\tau \end{pmatrix}, \tag{4.20}$$

but fortunately it needs not to be computed explicitly since it is zero-mean and can therefore be discarded for the prediction.

The covariance matrix evolves to

$$\hat{V}_k = F_k V_{k-1} F_k + \begin{pmatrix} \frac{\Delta t_k^3}{3} & 0 & \frac{\Delta t_k^2}{2} & 0 \\ 0 & \frac{\Delta t_k^3}{3} & 0 & \frac{\Delta t_k^2}{2} \\ \frac{\Delta t_k^2}{2} & 0 & \Delta t_k & 0 \\ 0 & \frac{\Delta t_k^2}{2} & 0 & \Delta t_k \end{pmatrix} \tilde{q} \tag{4.21}$$

with the so called power spectral density $\tilde{q}$ as a constant factor. The second summand of equation 4.21 constitutes the covariance $E\left[v_{k-1}v'_{k-1}\right]$ of the process noise based on the covariances of the acceleration noises with $\tilde{q} = \breve{q}$. This additive covariance grows exponentially with Δt because it models the increasing uncertainty in the position over time, when no evidence is given.

As already mentioned in section 4.3, the prediction step could be adapted for other, also non-linear motion models with e.g. the unscented transformation of UKF to compute the predicted Gaussian state $\hat{x}_k$ given the previous one (c.f. figure 4.6).

4.6.2 Association step

Associations are sampled according to the likelihood $\Pr\left(J_k|\hat{x}^i_k, z_k\right)$ which is based on consistency of motion and identity.

We model associations

$$J_k : \{1, \ldots, N_{z_k}\} \to \{0, 1, \ldots, N_t\} \tag{4.22}$$

as function from the N_{z_k} measurement indices of the current sweep at time k to their assigned target index or 0 if not assigned. We denote

$$\Im_k : \{0, 1, \ldots, N_t\} \to \wp\left(\{1, \ldots, N_{z_k}\}\right) = \left(J_k\right)^{-1} \tag{4.23}$$

as the inverse mapping from target indices to their assigned observations for convenience.

Independent measurement likelihoods

Direct sampling of a total association J_k is difficult, since the enumeration of every possible association J_k is infeasible because the number is exponential in the number of measurements (there are $(N_t + 1)^{N_{z_k}}$ possible associations). If we look at an individual association for a measurement $z_{k,l} \in z_k$, we can enumerate all $N_t + 1$ possible individual assignments easily as $z_{k,l}$ can be clutter viz. a false alarm or assigned to one of the targets. The probability of an association factors to individual associations assuming conditional independence of individual associations given predicted state and measurements

$$\Pr\left(J_k|\hat{x}_k, z_k\right) = \prod_{l=1}^{N_{z_k}} \Pr\left(J_k(l)|\hat{x}_k, z_{k,l}\right). \tag{4.24}$$

Based on Bayes' rule, the independence of associations $J_k(l)$ and predicted positions $\hat{x}_k$, the probability for an individual assignment can be converted to

$$\Pr\left(J_k(l) = j|\hat{x}_k, z_k\right) = \frac{p\left(z_{k,l}|\hat{x}_k, J_k(l) = j\right)\Pr\left(J_k(l) = j\right)}{\sum_{j=0}^{N_t} p\left(z_k|\hat{x}_k, J_k(l) = j\right)\Pr\left(J_k(l) = j\right)}. \tag{4.25}$$

Identification

The probability of an individual association of measurement l and target j is denoted by $\Pr(J_k(l) = j)$. It refers to the likelihood of l being identified as a measurement of j. The different player identification modules (c.f. section 2.2) are assumed to identify the measurements independently or rather to provide independent probability distributions for the identities. According to the total probability theorem 2.12, the probability of an association is given by summing the probabilities gathered by all different identification sources s as

$$\Pr(J_k(l) = j) = \sum_s \Pr(J_k(l) = j|s) \Pr(s). \tag{4.26}$$

If we assume $\Pr(s)$ to be constant for all sources s, $\Pr(s)$ inserted in equation 4.25 cancels such that the likelihood of an individual association including the sensor fusion of different identity information sources s reads as

$$\Pr(J_k(l) = j|\hat{x}_k, z_k) = \frac{p(z_{k,l}|\hat{x}_k, J_k(l) = j) \sum_s \Pr(J_k(l) = j|s)}{\sum_{j=0}^{N_t} p(z_{k,l}|\hat{x}_k, J_k(l) = j) \sum_s \Pr(J_k(l) = j|s)}. \tag{4.27}$$

Player assignment

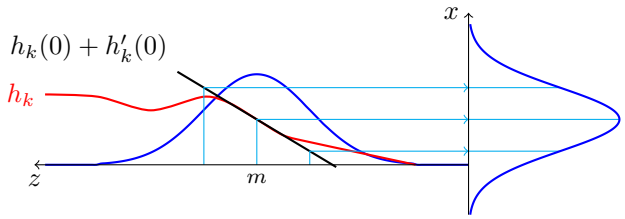
The probability for a data association between player j and an observation l according to the kinematic model is independent of the predicted positions of the other players. Measurements and a priori states are connected by the measurement model of equation 2.8. Since predicted state and measurement are Gaussian and the measurement model is assumed to be linear Gaussian, the pdf of a measurement given its assigned position can be computed analytically as

$$p\left(z_{k,l}|\hat{x}_k^i, J_k(l) = j\right) = \mathcal{N}\left(z_{k,l}; H_{k,l}\hat{m}_{k,j}^i, H_{k,l}\hat{V}_{k,j}^i H_{k,l}{}' + R_{k,l}\right) \tag{4.28}$$

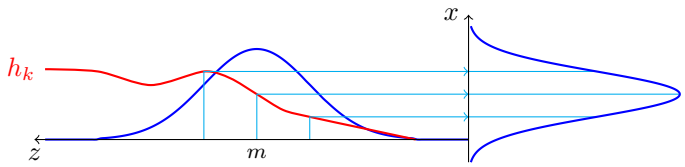
with measurement model $H_{k,l} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$ and $R_{k,l}$ as measurement noise covariance of $z_{k,l}$.

The right side of equation 4.28 turns out to be the likelihood function of the Kalman filter (c.f. [13]) used in one way or the other by all Kalman filter based multi-target tracking approaches for data association. The measurements $z_{k,l}$ of equation 4.28 only provide information about the position and not the velocity because we process frames which are discretized in time. To include measurements of the velocity, $H_{k,l}$ would be formed by the identity matrix.

Although the likelihood of an association for a single measurement and a single target is stated in equation 4.28 for linear Gaussian measurement models, its calculation scales to the non-linear case easily. In that case, position $\breve{z}_{k,l}$ and



(a) First order Taylor approximation (EKF) [118]



(b) Unscented Transformation (UKF) [121]

Figure 4.6: The most common approaches to propagate a Gaussian pdf by a nonlinear function. Depicted is the one-dimensional case propagating from the z-axis to the x-axis by the nonlinear function h_k (images are adapted from [89]).

covariance $\breve{R}_{k,l}$ of the measurement l result from the non-linear projection of the original observations into the two- or four-dimensional target state space. This can be achieved by applying the known measurement model h_k to the positions and covariances as proposed by one of the methods which extend the Kalman filter to the non-linear case. The two most common approaches EKF and UKF are depicted in figure 4.6.

Since our measurement model is given by the homography from image to playing field coordinates with distortion and this transformation is non-linear, we use the Unscented Transformation because it is fast and robust and "[i]t is founded on the intuition that it is easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function or transformation" (see Julier and Uhlmann [121]). The following sigma points s are transformed by the nonlinear function:

$$\begin{aligned} s_0 &= \breve{z}_{k,l} & W_0 &= \tfrac{\kappa}{N_{\breve{z}_{k,l}}+\kappa}, \\ s_i &= \left(\breve{z}_{k,l} + \left(N_{\breve{z}_{k,l}} + \kappa\right)\breve{R}_{k,l}\right)_i & W_i &= \tfrac{1}{2(N_{\breve{z}_{k,l}}+\kappa)} \text{ and} \\ s_{i+N_{\breve{z}_{k,l}}} &= \left(\breve{z}_{k,l} - \left(N_{\breve{z}_{k,l}} + \kappa\right)\breve{R}_{k,l}\right)_i & W_{i+N_{\breve{z}_{k,l}}} &= \tfrac{1}{2(N_{\breve{z}_{k,l}}+\kappa)}, \end{aligned} \tag{4.29}$$

where we follow Julier and Uhlmann and set $\kappa = 3 - N_{\breve{z}_{k,l}}$ because the measurements are assumed to be Gaussian. These sigma points are transformed by the non-linear measurement function h_k

$$\mathring{s}_i = h_k\left(s_i\right). \tag{4.30}$$

The transformed mean which is inserted into equation 4.28 is computed as

$$z_{k,l} = \sum_{i=0}^{2N_{z_{k,l}}} W_i \mathring{s}_i \tag{4.31}$$

with covariance

$$R_{k,l} = \sum_{i=0}^{2N_{z_{k,l}}} W_i \left(\mathring{s}_i - z_{k,l}\right)\left(\mathring{s}_i - z_{k,l}\right)'. \tag{4.32}$$

Clutter

Measurements are assigned to no target with probability $p\left(J_k(l) = 0|\hat{x}_k, z_k\right)$ assuming they were generated as clutter. If no informative clutter model for the localization module is known, a uniform distribution over the total measurement space with volume $\mathcal{Z}$, which is independent from the predicted state and measurement position, can be taken

$$p\left(J_k(l) = 0|\hat{x}_k, z_k\right) \approx |\mathcal{Z}|^{-1}. \tag{4.33}$$

Despite the fact that this uniform clutter model usually deviates from the truth, since the probability for a false measurement increases around the real target positions, it gives more or less good results.

The clutter likelihood $p\left(J_k(l) = 0|z_k\right)$ functions as a soft gating: because of the normalization in equation 4.25 and the exponential decrease of target association probabilities of equation 4.28 on the distance, it decreases the probability for far targets to be sampled.

Modeling of multiple measurements

As can be seen from the definition of associations in equation 4.22, we assume the same weak exclusion principle like the PMHT: multiple measurements can be assigned to one player, but a measurement is exclusively assigned to a single target. The number of multiple measurements inside a single sweep that are associated with a single target is usually bounded. This restriction can be modeled in different ways dependent on the sensor characteristics.

Prior to the sampling of associations for particle i, the maximal number of associations $\bar{N}^i_{k,j} \geq |\Im_k(j)|$ for each target j is drawn according to a model which constrains the multiplicity of the associations. We prefer to sample the maximal number instead of the actual one, contrary to what is usually done. The typical approach of drawing the actual number directly cannot be easily achieved correctly in a sequential process, because it causes several problems in respect to the data association induced. The number of associations cannot be sampled in isolation from each target because the total number of associations

must match the number of available measurements $\sum_{j=0}^{N_t} |\Im_k(j)| = N_{z_k}$, but also the number of measurements in the vicinity of the target j should have an influence on the actually sampled number. This forms a complex pdf from which one cannot sample directly given a uniform distribution which is mostly the only distribution provided by common random generators.

In the following, we propose a sequential sampling scheme excluding targets from being assigned multiple times during the sampling process. Let A_k^i denote the set of all targets that can be assigned to an additional measurement of the current sweep. This set is assigned to all targets initially for each particle $A_k^i = \{1, \ldots, N_t\}$. If a target j was drawn to be associated according to equation 4.27, it is excluded from further associations according to the proposal probability

$$q\left(j \notin A\right) = 1 - \frac{\Pr_j\left(n = |\Im_k(j)|\right)}{1 - \sum_{k=0}^{|\Im_k(j)|-1} \Pr_j\left(n = k\right)} \tag{4.34}$$

where $\Pr_j\left(n = k\right)$ denotes the probability that target j is assigned to exactly k measurements. After each exclusion, the association proposal density of equation 4.27 for the following measurements of the current sweep is normalized, omitting the removed target in the denominator

$$\Pr\left(J_k(l) = j | \hat{x}_k, z_k\right) = \frac{p\left(z_{k,l} | \hat{x}_k, J_k(l) = j\right) \sum_s \Pr\left(J_k(l) = j | s\right) \delta_{j,A}}{\sum_{j=0}^{N_t} p\left(z_{k,l} | \hat{x}_k, J_k(l) = j\right) \sum_s \Pr\left(J_k(l) = j | s\right) \delta_{j,A}} \tag{4.35}$$

with $\delta_{j,A} = \begin{cases} 1 & \text{if } j \in A \\ 0 & \text{if } j \notin A \end{cases}$.

Several plausible models can be regarded to constrain the number of associations per target. The Kronecker delta function constitutes a model for a maximum threshold

$$\Pr\left(n = k\right) = \delta\left(k - \max\right) = \begin{cases} 1 & \text{if } k = \max \\ 0 & \text{if } k \neq \max \end{cases}. \tag{4.36}$$

If the exclusion principle of JPDAF holds, i.e. targets can be assigned to one measurement at max, a maximum threshold model with max = 1 should be used. For an unconstrained number of associations per target, the threshold is set to a high number $\max \to \infty$. Alternatively, the Poisson distribution

$$\Pr\left(n = k\right) = e^{-\lambda} \frac{\lambda^k}{k!} \tag{4.37}$$

is frequently suggested for this task. If one wants to differentiate the probabilities of one detection at all (p_d) and multiple measurements, the following function can be used

$$\Pr\left(n = k\right) = \begin{cases} 1 - p_d & \text{if } k = 0 \\ p_d p_{sd}^{k-1} \left(1 - p_{sd}\right) & \text{if } k > 0 \end{cases}, \tag{4.38}$$

where p_{sd} adjusts the probability for multiple associations by a binomial distribution. In general, an arbitrary probability distribution defined on $\mathbb{N}_0$ can be chosen as a model for the number of associations $\Pr(n = k)$ (except $\delta(k - 0)$, which makes no sense anyway).

Because the model distribution must be evaluated on integer numbers only and these numbers are bounded below

$$\min\left(|\mathcal{Z}|,\ \text{minimal } k \text{ such that } \sum_{i=0}^{k} \Pr(n = k) = 1\right),$$

a look-up table for the exclusion probabilities can be precomputed and used for computational efficiency.

Unfortunately, the set A is updated sequentially, violating the independence assumption of the individual associations in equation 4.24. If the number of associations per target is constrained, the order in which the individual associations are drawn influences the sampling probability. To reduce the impact of violating the independence assumption on the importance density, the ordering of the measurements of one sweep is shuffled uniformly at random preliminary to each individual association step if multiplicity is constrained.

Sampling

Taken together, we sample a total association J_k^i by sequentially sampling an individual association $J_k^i(l)$ for each of the N_z measurements according to the proposal probability of equation 4.35. This is achieved by drawing a number $u_{k,l}$ uniformly at random in $(0, 1)$ and taking the minimal j such that

$$u_{k,l} \sim \mathbb{U}(0, 1) \leq \sum_{i=0}^{j} Pr\left(J_k(l) = i | \hat{x}_k, z_k\right). \tag{4.39}$$

The individual associations $J_k^i(l)$ are combined to form the total association J_k^i afterwards.

4.6.3 Fusion step

Given the sampled association J_k^i, the predicted player positions $\hat{x}_k$ must be fused with the assigned observations optimally in the sense of minimal variance to result in samples for the current state

$$x_k^i = \arg\max_x p\left(x | J_k^i, \hat{x}_k^i, z_k\right). \tag{4.40}$$

The Kalman update was shown to be an optimal solution for this problem [13], and it can be applied individually for each player j as

$$m^i_{k,j} = \hat{m}^i_{k,j} + \hat{V}^i_{k,j} H'_{k,\Im^i_k(j)} \left(H_{k,\Im^i_k(j)} \hat{V}^i_{k,j} H'_{k,\Im^i_k(j)} + R_{k,\Im^i_k(j)} \right)^{-1} \times \\ \times \left(z_{k,\Im^i_k(j)} - H_{k,\Im^i_k(j)} \hat{m}^i_{k,j} \right), \tag{4.41}$$

with $H_{k,\Im^i_k(j)}$ denoting the linear measurement model (4.28) as stacked matrix of $H_{k,l}$, $z_{k,\Im^i_k(j)}$ as stacked vector of $z_{k,l}$ and $R_{k,\Im^i_k(j)}$ as diagonal matrix of measurement covariances $R_{k,l}$ for all $l \in \Im_k(j)$. The covariances are updated as

$$V^i_{k,j} = \left(\left(\hat{V}^i_{k,j} \right)^{-1} + H'_{k,\Im^i_k(j)} \left(R_{k,\Im^i_k(j)} \right)^{-1} H_{k,\Im^i_k(j)} \right)^{-1}. \tag{4.42}$$

4.7 Filtering Step

After the sampling of new particle states according to the important density of equation 4.14, the weights for these particles must be set appropriately by equation 4.7. We assume that the sampled association J^i_k, which led to x^i_k, constitutes the only reasonable association given the sample and the measurements, therefore $p\left(J^i_k | x^i_k, z_k\right) = 1$, and equation 4.7 can be rewritten as

$$w^i_k \sim w^i_{k-1} \frac{p\left(z_k | x^i_k, J^i_k\right) p\left(J^i_k | x^i_k\right) p\left(x^i_k | x^i_{k-1}\right)}{q\left(x^i_k | x^i_{k-1}, z_k\right)}. \tag{4.43}$$

The posterior pdf of each target is conditionally independent given J^i_k and we can factorise the numerator such that

$$w^i_k \sim w^i_{k-1} \frac{p\left(J^i_k | x^i_k\right) \prod_{j=1}^{N_t} p\left(z_{k,\Im^i_k(j)} | x^i_{k,j}, J^i_k\right) p\left(x^i_{k,j} | x^i_{k-1,j}\right)}{q\left(x^i_k | x^i_{k-1}, z_k\right)}. \tag{4.44}$$

The likelihood of the association J^i_k given x^i_k is determined by the constraints on the multiplicity of assignments only

$$p\left(J^i_k | x^i_k\right) = \prod_{j=0}^{N_t} \Pr_j \left(n = |\Im^i_k(j)| \right) \tag{4.45}$$

assuming the independence of each target and clutter.

The prior $p\left(x^i_{k,j} | x^i_{k-1,j}\right)$ can be evaluated based on the predicted state $\hat{x}^i_k$ via the likelihood function as

$$p\left(x^i_{k,j} | x^i_{k-1,j}\right) = \mathcal{N}\left(m^i_{k,j}; \hat{m}^i_{k,j}, V^i_{k,j} + \hat{V}^i_{k,j}\right). \tag{4.46}$$

The measurement likelihood can be calculated in an analogue way (c.f. equation 4.28) as

$$p\left(z_{k,\Im^i_k(j)} | x^i_k, J_k\right) = \mathcal{N}\left(z_{k,\Im^i_k(j)}; H_{k,\Im^i_k(j)} m^i_{k,j}, H_{k,\Im^i_k(j)} V^i_{k,j} H^{i,\prime}_{k,\Im^i_k(j)} + R_{k,\Im^i_k(j)}\right)^{\lambda} \tag{4.47}$$

with $\lambda = \frac{1}{|\Im_k^i(j)|}$ and $z_{k,\Im_k^i(j)}, H_{k,\Im_k^i(j)}$ as well as $R_{k,\Im_k^i(j)}$ defined in the same way as for equation 4.41. The exponent λ for the likelihood function is due to the compensation of different dimensionalities of $z_{k,\Im_k^i(j)}$ for different associations J_k^i.

4.8 Implementational Remarks

This section specifies issues that should be considered when the proposed tracking algorithm is implemented as a computer program. Several improvements have been used to enhance the performance of the RBRPF method.

4.8.1 Memoization and smart resampling

Since $p\left(\hat{x}_k^i | x_{k-1}^i\right)$ is conditionally independent given the state x_{k-1}^i, the prediction step can be executed once for all N_i replicates of the previous particle x_{k-1}^i that have been resampled according to equation 4.12. The importance density which was used for sampling the associations (c.f. section 4.6.2) is stored and reutilized for filtering in equation 4.44. These memoizations save computation time and improve the efficiency of the proposed RBRPF.

To save computational time in future iterations of the RBRPF, the N_i sampled, discrete associations J_k^i are checked for equality and particles with identical associations are joined to a single particle. The state of this particle is trivially set to the state of the combined particles, while its importance weight is computed as the number of the combined particles times the weight of one of these particles. This is done before the fusion step 4.6.3 to avoid redundant computations. The number of clutter measurements should be compared first to detect if two associations differ, to avoid unnecessary comparisons.

4.8.2 Parallelization

The proposed algorithm is well suited for parallelization because all particles can be independently sampled and filtered. The individual associations can be drawn in parallel if the multiplicity of associations for a single target is unconstrained; otherwise an appropriate partition of the measurements is presupposed.

4.8.3 log-space and restricted codomain

Because the likelihoods can be very small, all probabilities are computed in log-space to avoid numerical problems. Because the probabilities must be added

frequently for normalization (especially in equation 4.25), we propose a fast version for summing two numbers that are given in log-space

$$\log\left(e^a + e^b\right) = \begin{cases} a + \log\left(1 + e^{b-a}\right) & \text{if } a > b \\ b + \log\left(1 + e^{a-b}\right) & \text{if } a \leq b \end{cases} \tag{4.48}$$

This saves the computation of one power operation by an extra of one comparison, two additions and a subtraction and reveals an average saving of 20% time for this operation. We also use a fast version of log based on look-up tables which needs also 20% less computation time than the standard operation.

Due to constraints inherent in the domain, we bound the position and velocities of all players to reasonable values so that players are assumed to be located inside the playing field and velocities cannot exceed the speed of a 100 m world class runner ($10\frac{m}{s}$); the covariances are bound accordingly.

4.8.4 Final estimate

The final estimate which serves as output of the method should be selected as the particle with maximum probability. Although it is common to choose the weighted mean of all or clustered subsets of the particles, the inherent multimodality in the design of the proposed method based on the sampling of associations would lead to the so called ghost phenomenon. The ghost phenomenon [31] refers to the appearance of a ghost target at the mean of the different modes of the distribution, where the target is known not to be for sure. Induced by memoization and the smart resampling procedures, the particles already represent clusters somehow.

4.8.5 Negative information

Up to now, we have described how to exploit all available evidence given by spatial measurements. On the other hand, the absence of observations carries information as well. This type of information is called negative information and has been exploited for tracking already. For example, Patterson et al. [186] utilized it in tracking based on GPS:

> "In particular, most buildings and certain outdoor regions are GPS dead-zones. If signal is lost when entering such an area, and then remains lost for a significant period of time while the GPS device is active, then one can strengthen the probability that the user has not left the dead-zone area."

We take advantage of negative information (especially for tracking in broadcasted videos) in two ways.

Its first usage is based on the assumption that players are occluded only for a fairly short time due to interactions. If a player is thus not assigned to measurements obtained by a certain camera over a reasonably long period of time, the probability that this player is outside the visible area of this camera rises. If the probability exceeds a predefined threshold, the player is pushed to the closest point outside of the polygon which describes the visible area and its velocity is set to zero. Since the time a player is not detected also depends on the quality of vision-based segmentation, we utilize this kind of negative information very conservatively with a high probability threshold.

Secondly, if a player has left the visible area of a camera for a reasonable time, he is assumed to remain outside the visible area if no measurements suggest otherwise. Such players are repeatedly pushed to the closest point outside of the observed area until they can be assigned to measurements again. This approach is especially useful for panning cameras that do not capture the total playing field as is the case in broadcasted soccer games.

4.8.6 Runtime analysis

We give a rough worst case complexity analysis of the RBRPF algorithm, which is depicted in figure 4.5. One iteration of the algorithm needs in the worst case

$$\mathcal{O}\left(N_t N_p c_p + N_p \left(N_z N_t c_a + \left(N_t + N_z\right) c_f + \left(N_t + N_z\right) c_w\right) + N_p c_n\right) \tag{4.49}$$

steps with

c_p constant steps for the prediction of a single target,

c_a constant steps for the computation of the likelihood function for a single measurement and a single target (for association sampling),

c_f constant steps for the fusion of a single target with a single measurement,

c_w constant steps for the computation of the likelihood function for a single measurement and a single target (for filtering) and

c_n constant steps for normalization of a single weight.

The RBRPF algorithm is linear in the number of particles, measurements and targets (or players respectively) because the runtime complexity can be written as

$$\mathcal{O}\left(N_p N_t N_z\right). \tag{4.50}$$

For this reason, it constitutes a well scalable multi-target tracking method.

4.9 Demarcation to State-of-the-art

Our RBRPF approach shares most characteristics with other recent Rao-Blackwellized multi-target tracking methods as there are the RBPF by Särkkä, Vehtari and Lampinen [213] and the RBMCMCDA method by Khan, Balch and Dellaert [135]. Contrary to RBRPF, JPDAF and MHT are exponential in the number of measurements and targets and can be used for real-time tracking only if a rather low number of measurements per time-step or small gating thresholds are considered. The following section describes the similarities and differences between the proposed RBRPF and RBPF as well as RBMCMCDA.

4.9.1 RBPF

The Rao-Blackwellized Particle Filter (RBPF) [213], which extends the Rao-Blackwellized Monte Carlo Data Association (RBMCDA) method [212], applies a SIR particle filter to track multiple target positions. The filter is Rao-Blackwellized by tracking associations instead, assuming that individual data associations provide a sufficient model for the positions. They process one measurement at a time, instead of processing measurement *sweeps* at once as we do. Their importance density $p\left(c_k|z_{1:k}, c_{k-1}^i\right)$ (with association $c_k \approx J_k(l)$) for individual associations is analogue to equation 4.25. The target states are deduced analytically from the sampled association, similar to our fusion step in section 4.6.3 but with a single measurement only. Constraints on the multiplicity of associations for one target are modeled in a different way, namely by a mth order Markov chain on single associations. Since the importance density as well as the weight update differs from ours, the likelihood of the measurements given the sampled single association and the prior of the target is used instead of equation 4.44. They apply adaptive resampling triggered by the effective number of the particles.

RBMCDA suffers from a theoretical drawback. The authors describe the RBMCDA process in [212]:

> "Measurements are processed one at a time in sequential fashion instead of parallel fashion. The sequential and parallel update schemes are mathematically equivalent".

This statement is deceptive since sequential and parallel *update* schemes are equivalent but sequential and parallel *processing* are not. The sampling in a parallel fashion is independent of the ordering of the measurement sequence in contrast to the sequential one. We will proof this fact by showing a simple counterexample.

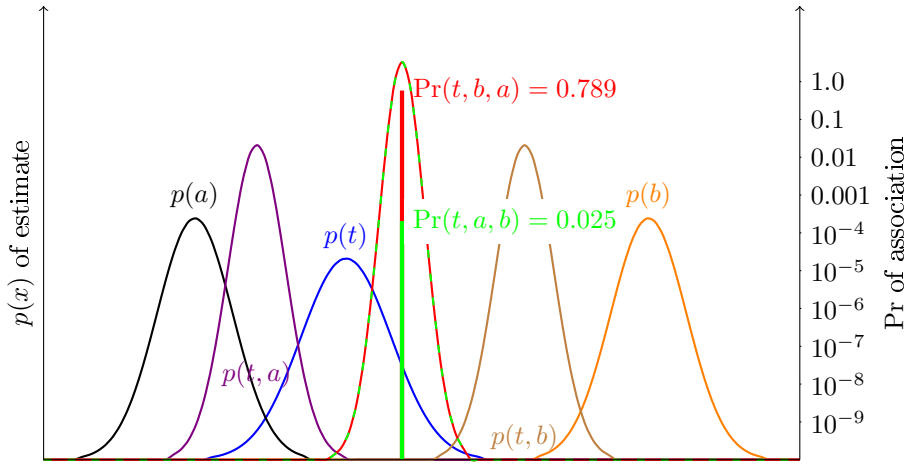


Figure 4.7: Example depicting the relevance of the order of intermediate fusions for the sampling probabilities. The fused Gaussian pdf of t with a followed by b is equivalent to the fusion of (t, b) with a, so the Gaussian curves of (t, a, b) and (t, b, a) coincide. However, the probabilities that these associations have been sampled differ greatly, as can be seen from the discrepancy of the bars at the mean of each estimate, which depict the probability for that estimate to be sampled with the corresponding probability value displayed on the right in an exp-scale.

The example is one-dimensional for convenience and contains just one target $t \sim \mathcal{N}(x; 0, 0.3)$ and two measurements $a \sim \mathcal{N}(x; -1, 0.25)$ and $b \sim \mathcal{N}(x; 2, 0.25)$ of a single sweep for simplicity. The clutter probability $P(J(a) = 0) = P(J(b) = 0) = 1.25 \cdot 10^{-6}$ was set comparatively low. Figure 4.7 shows the state distributions for the target t, the measurements and the different intermediate fused estimates with labels beneath the modes of each distribution. One can see that the fused pdf of t and a followed by b is equivalent to the fusion of (t, b) with a (the Gaussian curves of (t, a, b) and (t, b, a) coincide). But the probabilities that these associations have been sampled differ greatly. One can imagine that a target Gaussian fused with a measurement with very low uncertainty is rarely associated with another measurement afterwards. In the figure, a bar at the mean of each estimate depicts the probability for that estimate to be sampled with the corresponding probability value displayed on the right in an exp-scale. Fusing immediately after the association, the association of the target with both measurements is significantly more unlikely to be sampled if the measurement sequence is a followed by b, $(P((t, a, b)) = 0.025)$ than sampling associations of

the measurement sequence in the reverse order ($P((t,b,a)) = 0.789$):

$$\begin{aligned} &P((t,a,b)) = P((t,a,b)\,|\,(t,a)) = P(a|t)\,P(b|\,(t,a)) \\ &\neq P(b|t)\,P(a|\,(t,b)) = P((t,b,a)\,|\,(t,b)) = P((t,b,a)) \end{aligned} \tag{4.51}$$

Associating all measurements first and then fusing the assigned measurements with the corresponding targets at once (as done in our RBRPF approach) results in the same target posteriors sampled with the (correct) probability independently of the association sequence.

$$\begin{aligned} &P((t,a,b)) = P((t,a))\,P((t,b)) = P(a|t)\,P(b|t) \\ &= P(b|t)\,P(a|t) = P((t,b))\,P((t,a)) = P((t,b,a)) \end{aligned} \tag{4.52}$$

The dependence on the ordering of the measurement sequence is even more severe if the targets are constrained to be assigned to a single measurement at max. The RBPF approach behaves almost deterministically in some cases, omitting likely associations with a high probability. Imagine two targets t_1 and t_2 and two measurements a, b on a straight line with the same covariances, where the measurements a, b have the same distance as t_1, t_2 and t_1 is placed in the midst of a and b. Figure 4.8 depicts this scenario. If we assume no clutter and the strong exclusion principle of PMHT, the RBPF method would assign $J(a) = t_1, J(b) = t_2$ with the same probability as $J(a) = t_2, J(b) = t_1$ if the sequence is b followed by a while these associations are obviously not equally likely.

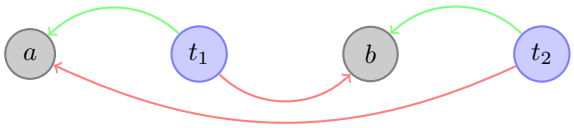


Figure 4.8: Thought experiment showing a pitfall for RBPF which samples the highly probable association of measurements a and b (depicted in green) to targets t_1 and t_2 with the same frequency as the second but unlikely association (depicted in red).

In our approach, shuffling the order of all measurements of a sweep prevents this pitfall, resulting in a much higher probability of $J(a) = t_1, J(b) = t_2$. This solution is not applicable for the RBPF approach due to its design as one measurement at a time, where only a single order can be chosen (even if this is done at random). One could argue that the order of measurements is random by nature, but this does not hold for most of the applications, since the order of measurements is predetermined by the sensor. For instance in computer vision, measurements are read typically top-down from left to right.

4.9.2 RBMCMCDA

The Rao-Blackwellized Markov Chain Monte Carlo Data Association approach [135] uses an Auxiliary Variable Sampling particle filter in which a separate Markov Chain is run for each of the Rao-Blackwellized particles. The same Rao-Blackwellization scheme is applied as in our approach, so a single state is formed by the combined Gaussians of each target. Also, their importance sampling focusing on optimally fused states is similar to equation 4.14.

On the other hand, there are many differences. Instead of separate Kalman updates, the optimal samples for a given association are found by solving a linear least squares problem with updating and downdating for efficiency (which results in the same solution). The different associations are visited during the Markov Chain, which is constructed following the Metropolis Hastings (MH) algorithm, which samples proposals with an acceptance ratio according to the likelihood ratio of the proposed state versus the current one. The different kinds of proposals are 1) the auxiliary variable proposal, which jumps between the particles and the different Markov Chains respectively, and 2) the edge proposal, which adds or removes an individual association, producing a new clutter or a newly assigned measurement. All types of associations are permitted, including merged measurements which are assigned to several targets. No constraints on the multiplicity of associations for a single target can be modeled, but these restrictions could possibly be integrated into the likelihood term of the assigned measurement proposal. Additionally, the authors claim that multiple associations are penalized automatically due to higher deviance of the measurements to their assigned target, which in turn results in a lower likelihood. Measurement gating is used as a heuristic to reduce the selection of possible associations. Interactions are modeled explicitly by common covariances of interacting targets. However, two targets are assumed to be independent if their distance exceeds a predefined threshold. The Markov Chains are initialized by nearest neighbor association and run for a predefined burn-in period ($4,000$ steps in their case) for mixing the chain, ensuring independence from the initial association.

The computational demand of the Markov Chain burn-in period restricts the RBMCMCDA to a small number of particles (one or six in their case) for real-time tracking. Thus, only a weak multimodality of the posterior can be tracked efficiently, in contrast to our approach. Their design of interactions by interfering covariances of the targets' states seems odd, since typically the motions of interacting targets are not aligned. At least this assumption does not hold for interacting athletes of different teams in sports, if the attacker intentionally tries to avoid the defending opponent. Since RBMCMCDA is heavily based on sequential Markov chain sampling, it is hard to be parallelized,

although an approach has been published recently [18]. We will experimentally compare the RBMCMCDA approach with ours in the next section.

4.10 Evaluation

We conducted three experiments for the proposed Rao-Blackwellized Resampling Particle Filter. The first experiment was a simulation demanding the ability to handle a large number of measurements as well as targets. The second experiment was taken from the sports domain where positions of basketball players were observed by laser range finders without any identity information. The third one called for tracking ants and was selected for the purpose of comparison with the RBMCMCDA method [135].

4.10.1 Simulation

To investigate the ability of the proposed method of tracking a high number of targets with multiple measurements through clutter, we adopted a simulation similar to the one described in [107]. Hundred targets are initialized to positions, which are uniformly distributed in $[-2000; 2000]^2$, and velocities drawn from the distribution $\mathcal{N}\left(0; 10^2\right)$. The targets are tracked for 100 measurement sweeps with the time between measurement sweeps set to 1. Measurements are taken of the true targets' positions, while each position measurement has an independent error which is distributed according to $\mathcal{N}\left(0; 20^2\right)$. The number of measurements generated by one single target is Poisson distributed with $\lambda = 3$. Clutter is drawn according to a Poisson with $\lambda_c = 100$, uniformly distributed over the whole tracking area $\mathcal{M} = [-4000; 4000]^2$. The last point in time of an exemplary simulation is shown in figure 4.9.

We initialized our tracker with the true target positions and zero velocity; the uncertainty in the target state was set to $100.0I_4$. The power spectral density of the process noise was set to $\tilde{q} = 1.0$. We used $N_{max} = 50$ particles to track the hundred targets.

Assignment	Single only	Multiple
Failures	37.01	8.21
Time(ms)	473.5	561.1

Table 4.1: Tracking results for the simulation experiment.

We counted tracking as a failure if the tracked target position differed by more than 100.0 from the true target position at the last point in time. We

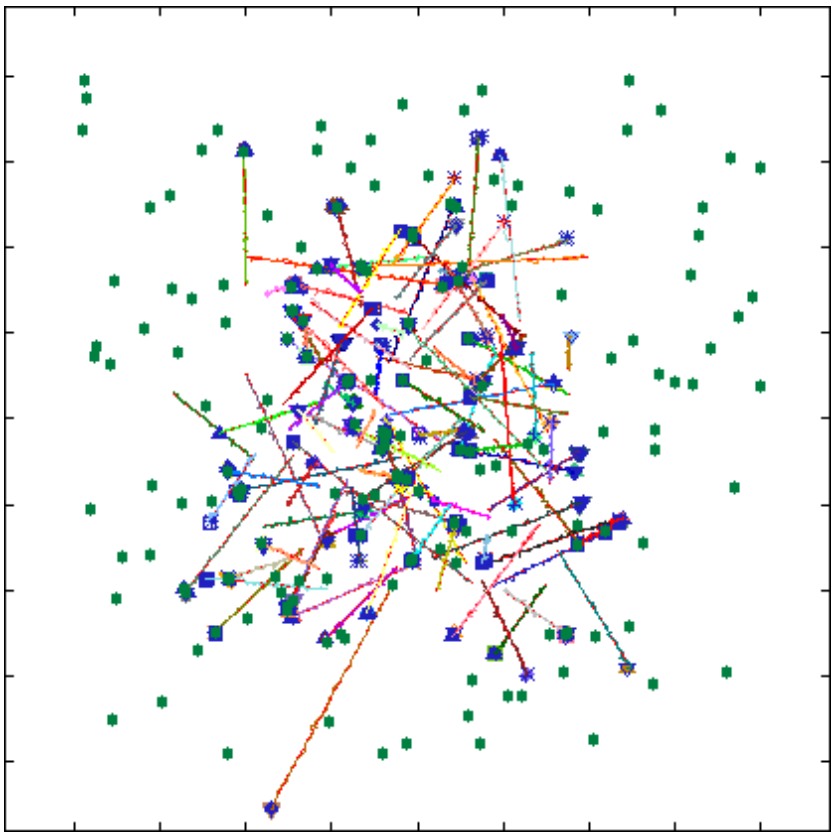


Figure 4.9: Tracking of hundred simulated targets which generate multiple measurements in clutter.

tracked the targets in a first run assuming single assignments only and the second run assumed the correct Poisson distribution. Table 4.10.1 depicts the mean number of failures during 100 simulation runs. Failed tracking is mostly due to very close initial target positions getting swapped or misled by clutter at the beginning. One can see the higher robustness due to the incorporation of more measurements and hence more information. Figure 4.10 graphs the median distance of the tracked targets to their true position with 0.25 and 0.75 quantiles for a single simulation run tracking with multiple measurements. This error remained within the measurement generation distribution $\mathcal{N}\left(0; 20^2\right)$ as desired.

The original simulation of [107] generated 400 tracks without clutter. Unfortunately, Horridge and Maskell [107] did not provide tracking errors but computation time only. We also conducted the same experiment, but tracking seemed futile since the measurement density was very high, as well as accompanied by

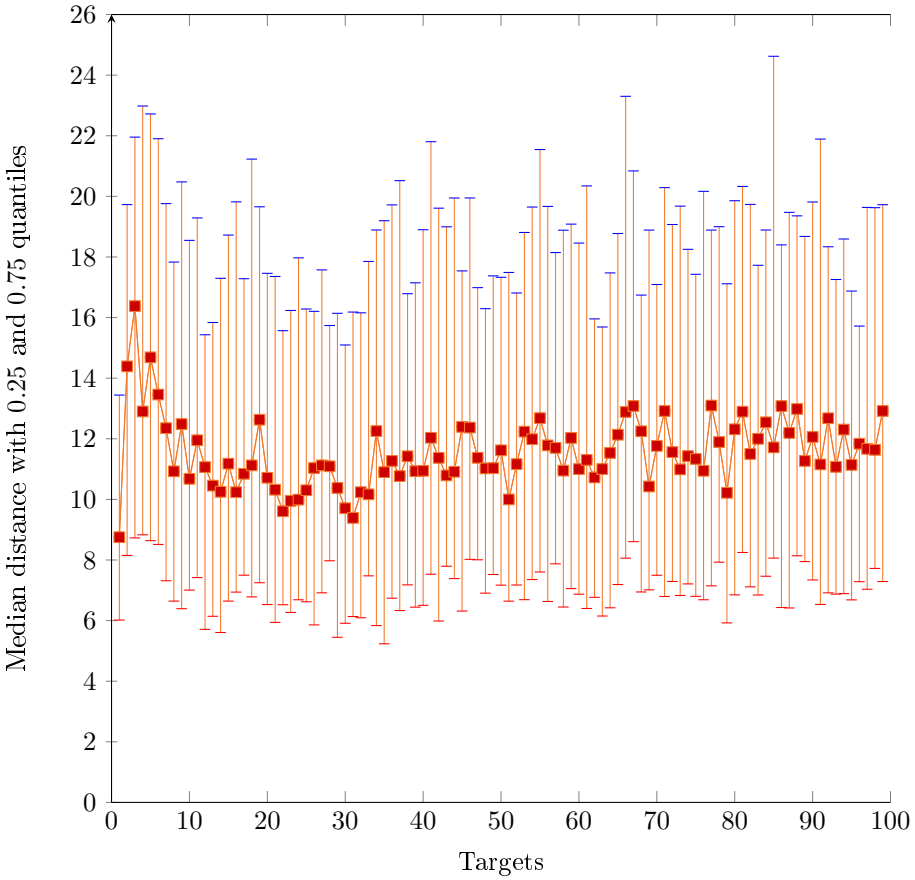


Figure 4.10: Median distance of the tracked targets to their true position for a single simulation run tracking 100 targets with multiple measurements in clutter.

a high uncertainty of each measurement, so arbitrary traces could be supported by observations resulting in low tracking performance.

4.10.2 Basketball

At the Georgia Tech BORG Lab, Balch, Dellaert and Starner are investigating algorithms for automatically tracking and modeling the behavior of multi-agent systems. The Laser Tracking Project also conducted an experiment that tracked a 4-on-4 basketball game over 22 minutes with eight players. Measurements were gathered using four SICK LMS291 laser range finders. "Each of these laser range finders scans, in half degree increments, an arc of 180 degrees, out to a range of up to 80 meters." (see [75]). The basketball court is covered by placing the laser range finders in the center of each surrounding vertex. Sometimes, players leave the observed area to get the ball from outside and can therefore not be observed

in that period of time. Occlusions are a big problem since laser scanners cannot perceive agents that are blocked by other agents or stationary objects. One of the lasers was knocked over at around 18 minutes after the beginning of the game. After the laser was reset for the rest of game, it was noticeably unaligned. Some hard scenes of the evaluated basketball game are depicted in figure 4.11. The laser data are published on `http://www.kinetrack.org`.

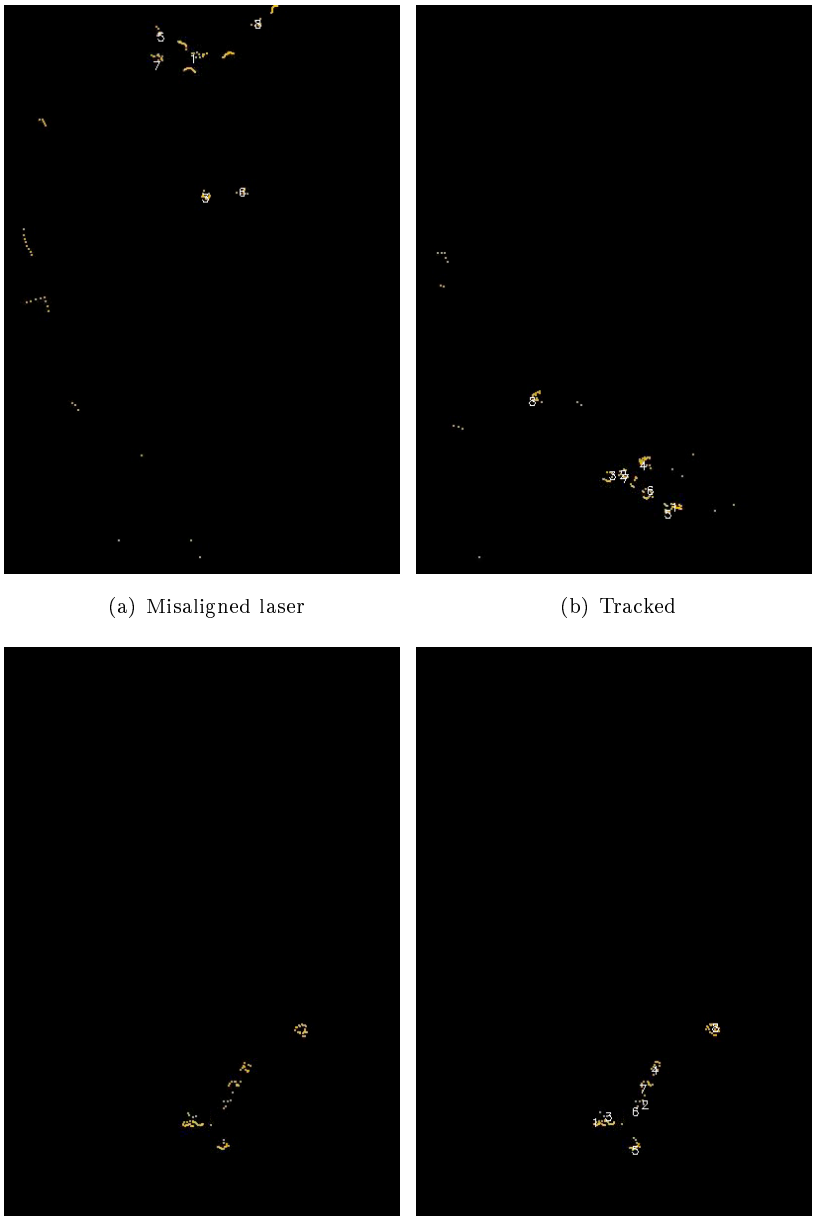
Feldman et al. [74, 75] applied clustering to these data to form trajectories of a variable number of targets. They planned to include identity information by RFID tags worn by each athlete. However, the authors did not track the identities since RFID measurements turned out to be unsuited for this task [10]. The ratio of the correctly detected *number* of targets was used to measure the performance of their approach.

We used a video of the background-subtracted laser data for the evaluation of RBRPF. The video contains 22 minutes or 81774 frames recorded with 37.5 Hz. Measurements were extracted by color thresholding. Around 1000 measurement points were detected in a single frame of size 400×576 pixels, yielding about 100 measurements per target. Since no identity hints are available, the tracking is exclusively based on the motion (model) of the targets.

Due to the lack of ground truth provided, we gathered these data ourselves by manually marking the players in each video frame image with the mouse pointer. We thereby encountered situations in which a target left the field of view for several frames and re-entered the scene at a different position. Since these cases violated our assumption of total visibility, failures are reported without the 49 failures that were due to re-entrances. Failures have been counted if the estimated position differed from the ground truth at least 30 pixels in Euclidean distance.

We set the parameters as follows: $N_{max} = 50$ particles, a very low clutter probability of $p\left(J_k(l) = 0|z_k\right) = 1.0 \times 10^{-11}$ due to the characteristics of the laser range finders, $\lambda = 100$ as an expected number of measurements per target for the Poisson constraint, $\tilde{q} = 0.1$ as growing factor for uncertainty and $v_{max} = 20$, both deduced from frame rate and typical human velocity thresholds; the measurement covariance matrix was defined as $V_z = I$ due to the high accuracy of the laser scans. Computation times were measured on a 2.5 GHz Quad-Core PC.

We encountered 1234 failures for the whole video, which equals a rate of 98.49% correctly tracked frames. The average time to process a single frame was 81.8ms and therefore not real-time. To cope with this deficiency, we down-sampled the images by halving width and height and tracked the video again with an adapted parameter $\lambda = 37$. This procedure decreased the computational demand to an average time of 22.5ms or a frame rate of 44fps, which



(a) Misaligned laser (b) Tracked

(c) Occlusion (d) Tracked

Figure 4.11: The laser data provide challenging scenes for every multi-target tracking method due to the inaccurate motion model in combination with the lack of separability of the players inherent in laser range data.

therefore provides a real-time tracking system for these data, because they were recorded with 37.5fps. The linear runtime complexity of the RBRPF is attested to empirically since the runtime was quartered according to quartering

the measurements. Happily, the failures also decreased slightly to 1209 failures, resulting in a correct tracking rate of 98.52%.

The failures which arose were primarily due to the inaccurate motion model in combination with the lack of separability of the players. In addition, the misalignment of one laser about one fifth of the time disturbed the tracking process by adding additional noise.

4.10.3 Ants

In [135] Khan et al. tested their proposed RBMCMCDA tracker on a challenging ground truth sequence of twenty ants in a small container. The image data and ground truth are available online at `http://www.kinetrack.org`.

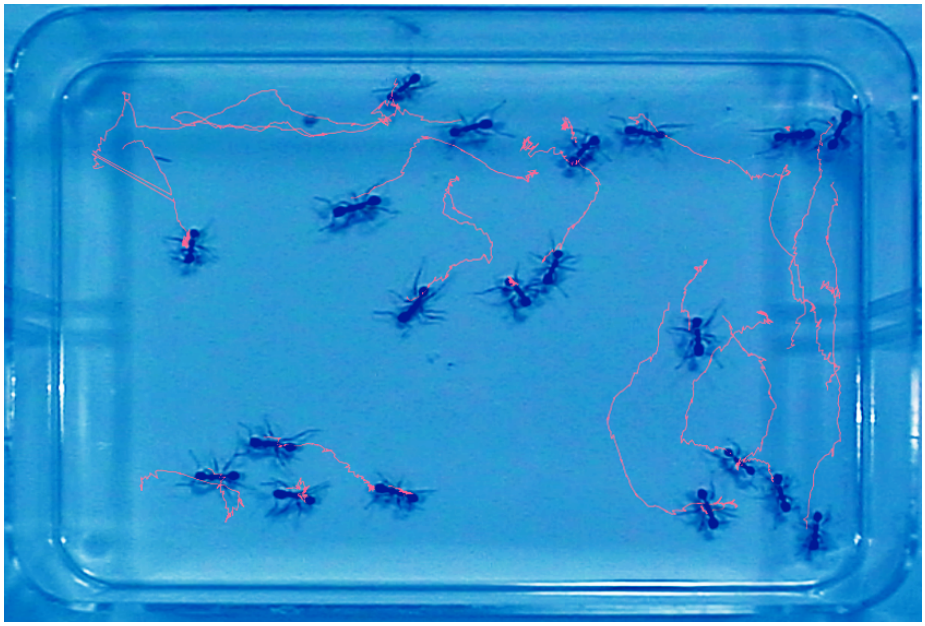


Figure 4.12: Tracking twenty visually similar ants through 10,400 frames with frequent interactions. Traces of the ants are shown as orange lines.

The ants that were to be tracked to gain insights in social behavior of insects are about 1 cm long and move as quickly as 3 cm per second, frequently interacting with up to five or more ants in close proximity. The test sequence presents a substantial challenge for any multi-target tracking algorithm and was selected for comparison purpose. One frame of the sequence is depicted in figure 4.12. The sequence consists of 10,400 frames recorded at a resolution of 720×480 pixels at 30 Hz. We used the same simple thresholding procedure of the blurred and downsampled video as [135] to obtain the measurements. The original images were blurred and downsampled twice to obtain an image that

was 180×120 pixels. Pixels with the following YUV ranges were considered as detections: $1 < Y < 75$, $122 < U < 128$, and $128 < V < 135$. The x, y locations on the smaller images were then scaled up to the original 720×480 image.

We used the same parameters as given in [135]. Target motion was modeled using a constant velocity model with time step $\Delta t = 0.033$ and $\tilde{q} = 32$. The initial covariance was set to $V_0 = 32I_{4N}$ and the measurement noise was set to $R = 32I_{4N}$. All positions and covariances are specified in pixels. We evaluated the proposed RBRPF with the same number of particles to provide a maximum of accordance to the experiment of Khan et al. Failures are counted when an estimated target position deviates from the ground truth position by more than 60 pixels. After a failure, all of the targets are reset to ground truth position and tracking is resumed.

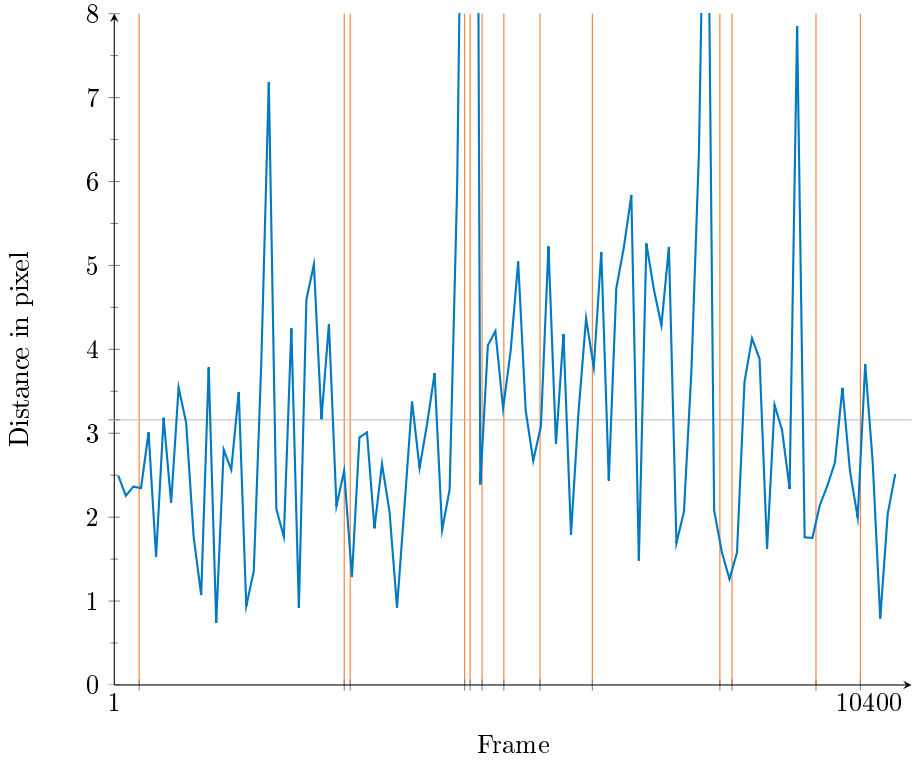


Figure 4.13: Average distance of tracked ants to groundtruth. The red vertical lines depict failures of the tracking process.

The number of failures detected on the ground truth sequence for the RBMCMCDA tracker with different numbers of particles and our tracker are shown in table 4.2. Results are also listed for a delayed version of our algorithm, which returns the estimated positions after a delay δ. The positions are taken from

Algorithm	Failures	Runtime
RBMCMCDA [135] $N = 1$	24	23.03 ± 0.87 fps @ P4-M 3Ghz
RBMCMCDA [135] $N = 6$	21	8.75 ± 0.55 fps @ P4-M 3Ghz
RBRPF $N_{max} = 6$	19	8.38 ± 1.5 fps @ P4-M 1.6GHz 40.68 ± 1.0fps @ Dual Core 2.5GHz
RBRPF $N_{max} = 6$ with delay $\delta = 4$	13	8.38 ± 1.5 fps @ P4-M 1.6GHz 40.76 ± 1.0fps @ DualCore 2.5GHz

Table 4.2: Experimental results for tracking ants through 10,400 frames.

the precedent particle of the current estimate. This method reduced the failure because it takes more information into account. The results in table 4.2 show the supremacy of our approach in accuracy, although the failure rate of both approaches is so low (99.77% correct tracking!) that this difference is not essential for the application.

We measured the runtime by the average frame rate in frames per second (fps) including image processing time. With current standard hardware, our method is able to track the twenty ants faster than real-time (40 fps) and with low failure rate. The delay reduces the number of failures even further while maintaining the frame rate at 40 fps. Our algorithm exhibits higher quality in tracking than the state-of-the-art tracker of [135]. This performance is achieved twice as fast, as the same time is needed for tracking but on a CPU with half the GHz (unfortunately, we had not the same processor on hand).

Contrary to [135], we do not allow merged measurements because these result mostly from the target in front occluding the target in back and may mislead the tracker. We restricted the number of detections of one target by a Poisson distribution, yielding less (possibly wrong) associations. Speed-up is achieved because we directly sample the associations instead of running a Markov chain, allowing a better constriction to necessary computations by memoization without the need for uninformative burn-in steps. The average distance over all ants is depicted in fig. 4.13. Analogous distance graphs have been published for the RBMCMCDA approach in [135]. The distance for tracking without delay obviously differs only minimally to the ones with delay, because both approaches use Kalman filters to predict and update target states. The mean for the average tracking error, at 3.16 pixels, is very low regarding the systematic error caused by downsampling to one fourth of the original resolution.

We also conducted experiments on a second ant dataset of [135] in which ants move on two glass layers. Khan et al. provide 16 image sequences. These video clips have been preprocessed in the same way as above to extract measurements but with different thresholds for the YUV ranges ($39 < Y < 101$, $116 < U <$

125, and $128 < V < 136$). The RBMCMCDA approach could track 12 of the 16 demanding sequences successfully with parameters $\Delta t = 0.1$, $V_0 = 32I_{4N}$, $\tilde{q} = 4$ and $R = 150I_{4N}$ but failed on sequences 5, 8, 12 and 14. Our approach could also handle 12 of the 16 sequences using the multiplicity constraint of equation 4.38 with $p_{sd} = 0.14$ but failed on 3, 8, 12 and 16 instead. With $p_{sd} = 0.4$, our method also tracked through sequence 16 successfully. All sequences include longer partial or full occlusions or sudden changes in direction and velocity, which makes it difficult for every tracker, assuming a constant velocity model. On an average, about 40 fps could be achieved on the Core 2 Duo with 2.5 GHz, emphasizing the real-time capability of RBRPF.

4.11 Conclusions

In this chapter we have proposed the Rao-Blackwellized Resampling Particle Filter as a novel approach for probabilistic real-time multi-target tracking, solving the problem of building consistent estimates of trajectories from noisy, cluttered measurements. RBRPF constitutes a Rao-Blackwellized SIR particle filter where importance sampling is solved partially in an analytic manner and partially by drawing associations based on predictions and measurements. We explicated the theoretical foundations and assumptions of the proposed algorithm in detail, justifying our decisions based on the characteristics of sports videos and mathematical conclusiveness. The approach is designed to track multiple targets of similar appearance, which is a challenge for every multi-target method. RBRPF is suitable for real-time performance, thanks to the linear runtime complexity which is due to Rao-Blackwellization, memoization and smart resampling. Constraints on the multiplicity of single target associations can be stated in a natural (mathematical) way and are integrated seamlessly, while the method – in contrast to others – is independent on the order of the measurement sequence of one sweep. We demarcated our approach against the state-of-the-art both from a theoretical and an empirical perspective, as it outperforms current multi-target tracking methods. The performance of RBRPF was evaluated on several demanding applications, proving its effectiveness and real-time capability.

Chapter 5

Position-based Identification

A typical game in team sports consists of structured and intentional motions of players who can be dichotomized in two teams. Because the motions of a team are synchronized and every player is allocated a specific role, the spatial positions can provide evidence about the identities of the corresponding players. The computational problem considered here is identification based on positions. It is illustrated in figure 5.1. We distinguish between initialization without a-priori knowledge and re-identification during the game. The tactical line-up is often the only information about identities that is available beforehand, and it offers a potential source for initializing the tracking module. During the game, plenty of estimated spatial data labeled with identifiers are made available as the main tracking output. Models of the players can be learned for the purpose of re-identification, exploiting these positions as training data. We assume that – together with the spatial information – their team affiliation is known as well. This information can be extracted from the video frames by appearance as described in section 6.2. At kick-off time, the positions themselves reveal their team affiliation, because the rules of the game typically permit all players of the same team to stay within their half of the field.

After a review of related work, we will discuss several methods for automatic initialization based on the tactical line-up. Further, we will investigate identification methods based on incrementally learned models of player positions. The different approaches are evaluated using real soccer games. The focus of this chapter is the exploitation of spatial data solely for the purpose of identification, while chapter 8 presents a discussion about tactic analysis in general, which sees models of positions themselves as one main subject of interest.

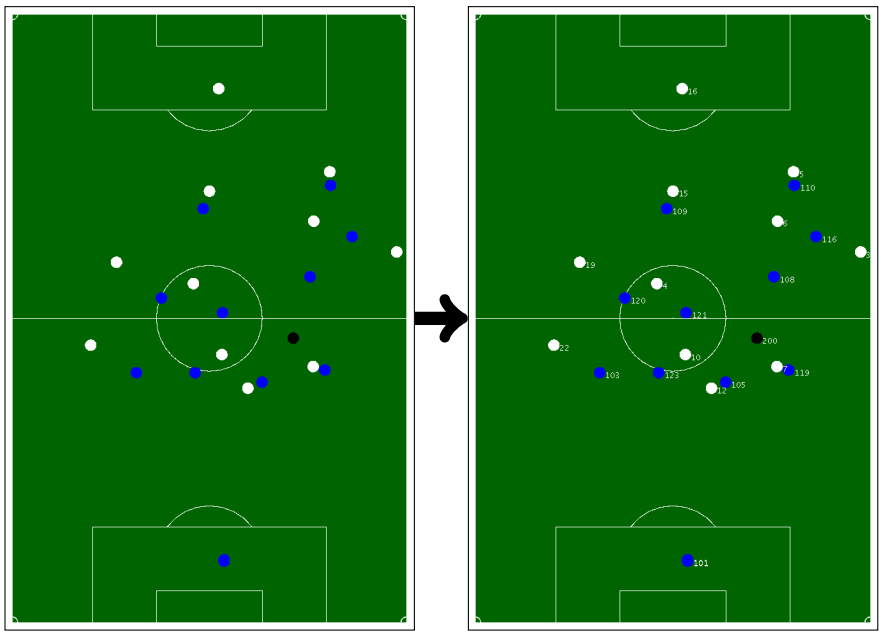


Figure 5.1: This chapter investigates the computational problem of assigning identifiers to plain positions which are already tagged with their team affiliation.

5.1 Related Work

Despite the advanced research in tracking methods, not much analysis of snapshots of athlete positions has as yet been done.

Visser et al. [250] classify formations in simulated RoboCup games. The bounding box of all players of one team is discretized into a grid and binary indicators are fed as input vectors to an Artificial Neural Network for training. Ramos and Ayanegui [199] build topological structures of player positions based on triangular planar graphs. Despite the fact that both approaches aim at detecting formations instead of identifying players, they have the potential to be expanded for that purpose.

Identification methods based on spatial data have been published in dissertations only. Needham [179] trains a Gaussian Mixture Model (GMM) for each player based on simulated data for 5-on-5 soccer. Identification is achieved by a graph algorithm. A fully-connected graph between the unknown players and the identities is generated, where the vertices are weighted by the probabilities based on the GMMs. Multilevel recursive bisection is applied to this association graph to extract valid and unambiguous labeling with approximate optimal probability. Vector Quantization was also applied on position and velocity data to model behavior as states of graphical models like HMMs, but "initial ex-

periments have shown that [they did] not have enough data to do this" [179]. Altruistic Vector Quantization (AVQ) was proposed by Johnson [120] of the same group to learn behavior models, but the approach has been evaluated on trajectories of pedestrians only. Intille proposes consistent player labeling in [115] based on logical rules (Horn clauses). A blackboard technique is used to infer the consequences of the rules, which denote the assignment of locations to identities. The inferred hypothesis for an association that covers the most players is selected as the final one. The approach was evaluated on American football with complex role models. However, the choice of the rules is essential for the algorithm and Intille states: "Encoding the rule sets is a laborious process". Further, he recommends a probabilistic framework: "Rules that softly weight relative evidence – modeling some important dependencies between related rules and evidence – seems necessary to make the algorithm practical given noisy trajectory data" [115].

5.2 Identification based on Tactical Lineup

We propose a novel method for labeling player positions with identities based on tactical line-up information. The method is suited not only for initialization at kick-offs but also for re-initialization after cuts in broadcasted video. The comparison of estimated labels with real identities forms an interesting analytical tool for sports scientists and coaches on its own and provides a measurement method for compliance of a team with the decreed line-up.

5.2.1 Tactical lineup

Besides manual submission, the tactical lineup can be gathered automatically from the web or from the broadcasted video itself. Websites providing live tickers of a game usually offer visualization of the formations as well. Because encoding is mostly based on HTML, the corresponding web page content needs to be parsed to extract relative or absolute positions of the players of both teams. Names and jersey numbers can often be captured from the same site. A general approach for extraction is not possible, because web pages vary highly in presentation and in code, but learning methods for easy parser generation by marking the fields of interest exist (e.g. [37]). Figure 5.2 depicts an example for the presentation of the formations based on HTML tables taken from `http://www.kicker.de`.

Figure 5.3(b) shows an example for the broadcasted lineup for a single team. Each broadcaster uses different visualizations, one for each type of tournament and season. Often semi-transparent overlays are used, which complicates the

(a) Visualization.

```
⋮

<table style="background: ...;
width: 579px; height: 1050px; ..."
summary="Taktische Aufstellung">
 <tbody>
  <tr>
   <td>
    <div style="position: absolute;
width: 103px; left: 233px; top: 5px;
text-align: center;">
     <img class="bild_s" style="width:
44px; height: 52px;" src="..." alt="">
     <br>
     <a class="link" href="...">Butt
(2)</a>
    </div>
    <div style="position: absolute;
width: 103px; left: 15px; top: 113px;
text-align: center;">
     <img class="bild_s" style="width:
44px; height: 52px;" src="..." alt="">
     <br>
     <a class="link" href="...">Lucio
(3)</a>
    </div>
    <div style="position: absolute;
width: 103px; left: 179px; top: 113px;
text-align: center;">
⋮
```

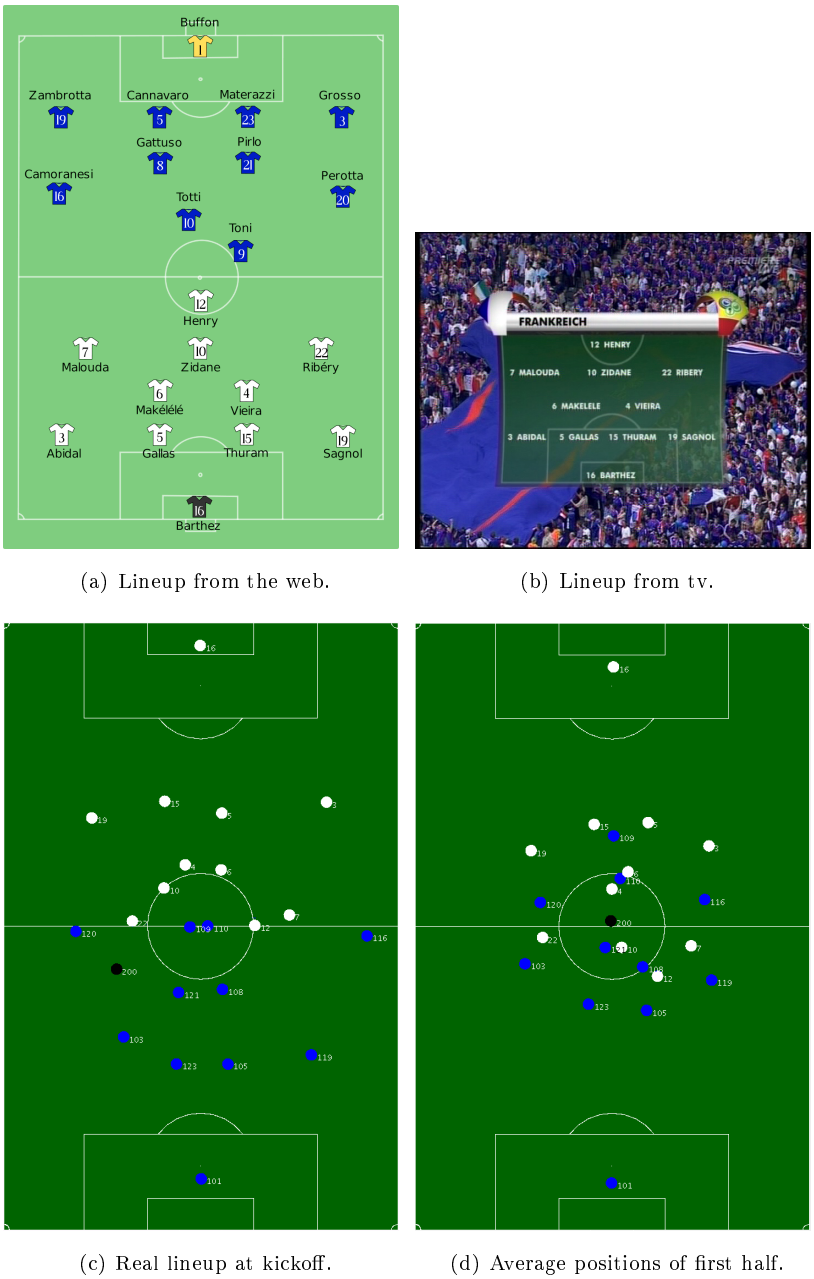
(b) HTML code.

Figure 5.2: Example for a website (`http://www.kicker.de`) providing the tactical lineup for a Bundesliga soccer game.

extraction task. The same methods used for other overlays can be applied to extract player names (c.f. [98]) or numbers (c.f. [29, 273]). The real player positions at the kickoff and their mean taken over the first halftime is shown in figure 5.3 for comparison.

5.2.2 Relative distance of associations

Assuming that positions of all players and their team affiliation is available, the problem of labeling consists in assigning these positions to identifiers, such that the distance of the resulting association to the tactical lineup is minimal with respect to a given distance measure. This measure should reflect the similarity



(a) Lineup from the web.

(b) Lineup from tv.

(c) Real lineup at kickoff.

(d) Average positions of first half.

Figure 5.3: Tactical lineups for the final game of the soccer world championships 2006.

of the labeling to the real (but unknown) identities. To keep notation simple, an association of elements between two ordered sets of positions is given by the ordering of these sets; for instance, the first position of a set A is associated with the first position of B. If a mapping of positions to player identities is known for one of these sets as it is the case for the tactical lineup, the mapping can be easily transferred to the other set.

The absolute positions gathered from formation images are neither very accurate nor reliable, as information is carried mainly by the relative positioning of the players to each other. Therefore, we propose the following symmetric distance $d(A, B)$ based on relative constraints for an ordered set of positions A and B of same size $|A| = |B| = N_t$

$$d(A, B) = \sum_{j=1,\ldots,N_t} c_j\,(A, B) \tag{5.1}$$

with

$$\begin{aligned} c_j\,(A, B) = {} & \alpha_l\,(|\{(x, y) \in A | y < y_j\}| - |\{(x, y) \in B | y < y_j\}|) \\ & + \alpha_r\,(|\{(x, y) \in A | y > y_j\}| - |\{(x, y) \in B | y > y_j\}|) \\ & + \alpha_f\,(|\{(x, y) \in A | x < x_j\}| - |\{(x, y) \in B | x < x_j\}|) \\ & + \alpha_b\,(|\{(x, y) \in A | x > x_j\}| - |\{(x, y) \in B | x > x_j\}|)\,. \end{aligned} \tag{5.2}$$

The proposed distance measures the compliance with weighted relative constraints for each player j. The constraints compare the number of players relative to each other and are induced by the given formations A and B. There are four different constraints: to the left l, to the right r, in front of f and behind b. For flexibility, the constraints are weighted by weights α describing their rigidity. Note that the numbers of players in one direction and its opposite are only bounded by the total number of players from above, but they do not have to match it $|\{(x, y) \in A | y < y_j\}| + |\{(x, y) \in A | y > y_j\}| \leq |A| - 1$ and $|\{(x, y) \in A | x < x_j\}| + |\{(x, y) \in A | x > x_j\}| \leq |A| - 1$.

5.2.3 Searching

The association problem can be solved by searching for the permutation of the input positions with minimum distance to the ordered set induced by the tactical lineup. The distance is computed according to the previous section. For an exhaustive search, all 11! possible associations would have to be listed and sorted. Since this is too costly, approximate gradient descent methods like simulated annealing could be used; Russell and Norvig [211] give an introduction in directed search methods. Unfortunately, the branching factor for the association search is fairly high $\binom{11}{2} = 55$, if the state space is traversed by

swapping two individual associations. A good heuristic for selecting the most promising assignments would be needed for acceptable runtime performance. Additionally, gradient descent methods typically require a continuous (and ideally convex) rating function for reasonable performance. Because this is not the case, searching provided bad and unreliable results in preliminary experiments. We thus abstained from further research in this direction and do not present the results here.

5.2.4 Sorting

The choice of the relative distance measure of section 5.2.2 permits computation of the association by sorting the positions with subsequent deterministic labeling. The formation is usually given in the form of lines as goalie, defense line, defensive midfielder, offensive midfielder and striker. We exploit this structure by sorting the positions by their x-coordinate first, partitioning the list by taking the number of players building each line and sorting these subvectors by their y-coordinates. A one-to-one mapping of position indices to identities is computed by tracing the identities of the positions of the given tactical line-up, which have been processed in the same way beforehand. This mapping is used for the final labeling of the unassigned positions. The lineup of France at the final of the FIFA world championships 2006 as depicted in figure 5.3 would result in the labeling according to figure 5.4.

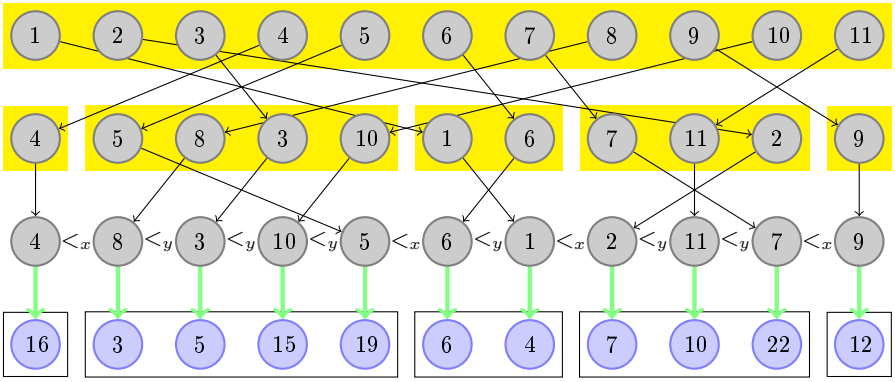


Figure 5.4: Hierarchical sorting of input positions (at the top row) down to association with the tactical lineup (at the bottom row), where the IDs are taken from figure 5.3 and lines are boxed.

Another approach consists in sorting the positions primarily according to their y-coordinates and ordering them by their x-coordinates if the distance between the y values is below a given threshold θ_{XY}. The association procedure

remains the same as for the other sorting approach.

5.2.5 Graph matching

We can transform the association problem to the problem of finding a min-weight bipartite graph matching of input positions A to the positions induced by the tactical lineup B. This is advantageous because fast algorithms for the graph matching already exist. The transformation is defined such that the matching provides the solution of the minimization problem which is imposed by the distance measure between B and any permutations of A.

We review the necessary terminology of graph theory in a nutshell: a weighted graph $G = (V, E, w)$ is called complete bipartite, if there exists a partition of nodes $V = A \cup B$ with $A \cap B = \emptyset$ and edges $E = A \times B$ with weights $w : A \times B \to \mathbb{N}_0$. A matching is a subset of edges $M \subset E$, such that every node is connected with at most one other node $\forall v \in V.|\{u|(u,v) \in M\}| \leq 1$. The size of a matching is written as $|M|$, denoting the number of edges in M. A Maximum Matching is a matching $M_{\max}$ with the maximum number of edges such that $\forall M.|M| \leq |M_{\max}|$ (which is not an unique property). A matching is called perfect, if every vertex is connected once $\forall u \in V.\exists v \in V.(u,v) \in M \vee (v,u) \in M$. The weight of the matching M is the sum of the weights of edges in M, $w(M) = \sum_{(x,y)\in M} w(x,y)$. Within the context of matchings, a path is called alternating, if its edges alternate between M and $E \setminus M$. An alternating path is called augmenting, if both endpoints of the path are not connected in M (also called free). A complete bipartite graph and a perfect matching is exemplified in figure 5.5.

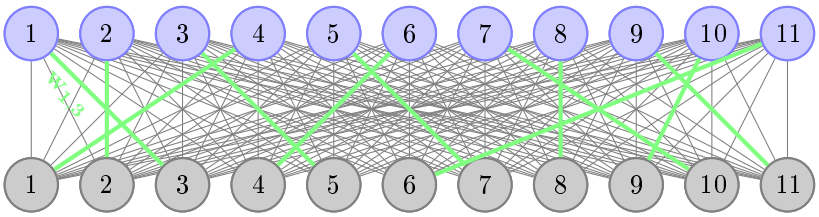


Figure 5.5: A maximum matching (depicted in green) of the complete bipartite graph (depicted in gray).

The Hungarian method proposed by Kuhn [145] solves the problem of finding a max-weight maximum matching in polynomial time $\mathcal{O}\left(|V|^3\right)$ by reducing it to a search for a feasible vertex labeling l and a perfect matching M in its equality subgraph G_l. A vertex labeling is a function $l : V \to \mathbb{N}_0$ and is called feasible if the combined labels always exceed the true weights $\forall (x,y) \in E.l(x) + l(y) \geq$

$w(x, y)$. The equality graph $G_l = (V, E_l)$ of a vertex labeling l contains all edges, where the labeling matches the true weights, or stated formally

$$E_l = \{(x, y) | x \in V \wedge y \in V \wedge l(x) + l(y) = w(x, y)\}.$$

The algorithm is depicted in figure 5.6. It starts with a feasible labeling l, which is divided into lx and ly by domains X, Y, and a matching M in E_l (step 1). As long as the matching is not perfect (step 2), an augmenting path for M in E_l is searched (step 9&10). Therefore, an augmenting tree (represented by S and T with edges from G_l), initialized with an unmatched vertex (step 3&4), is repeatedly expanded by adding alternating edges of E_l (step 11). If the tree forms an augmenting path, the matching M is set to the flipped augmented path (step 14), since this path contains one edge more than the augmenting path itself; in the next step a new augmenting tree will be created. If no augmenting path exists and the tree cannot be expanded (step 5), l is improved to l' such that $E_l \subset E_{l'}$ (step 6-8). Since either the size of M or E_l is increased in every iteration, the process must finally terminate. The complexity is reduced to $\mathcal{O}(|V|^3)$ by using buffering slacks $slack_j = \min_{i \in S} (l(x) + l(y) - w(x, y))$ (for step 5).

To compute the min-weight maximum matching with the Hungarian method, the weights have to be inverted (and shifted to be non-negative) as

$$w(x, y) = \max_{k,l} (w'(k, l)) - w'(x, y). \tag{5.3}$$

There are also modified versions of the Hungarian method, which find a specific number of best associations. Cox and Miller [54] proposed a method for finding the k-best assignments in polynomial time that is linear in k.

For the purpose of assigning input positions based on the tactical line-up, the weights are initialized with the difference of the relative constraints for position i, which was extracted from the input, to position j, which was deduced from the tactical lineup

$$w'(i, j) = c_j(\grave{A}, B) \text{ with } \grave{A} = (1\,2\,3\,4\,5\,6\,7\,8\,9\,10\,11)^i\, A. \tag{5.4}$$

The equation utilizes the repeated application of the permutation cycle which is denoted by $(1\,2\,3\,4\,5\,6\,7\,8\,9\,10\,11)$ for the correct mathematical notation. As the sum of these weights and therewith the individual constraints are minimized by the min-weight maximum matching, the distance between the tactical lineup B and the input positions A, which were reordered according to the resulting matching, is minimal as well.

HungarianMethod($\{w_{xy} \in \mathbb{N}_0\}_{y=1..N}^{x=1..N}$):

1. Init $M = \emptyset$, $lx_x = \max_y (w_{xy}), ly_y = 0$
2. WHILE $|M| < N$
3. Select $r \in \{x | \forall y.(x,y) \notin M\}$
4. Init augmenting tree rooted in r: $S = \{r\}, T = \emptyset$
5. IF $\bigcup_{x \in S} \{y | lx_x + ly_y = w_{x,y}\} = T$
6. $a = \min_{x \in S, y \notin T} (lx_x + ly_y - w_{xy})$
7. $lx_x = lx_x - a \quad \forall x \in S$
8. $ly_y = ly_y + a \quad \forall y \in T$
9. Select $y \in \{y | y \notin T \wedge x \in S \wedge lx_x + ly_y = w_{xy}\}$
10. IF $\exists z.(z,y) \in M$
11. $S = S \cup \{z\}, T = T \cup \{y\}$
12. Goto step 5
13. ELSE
14. $M =$ inverted augmenting path starting from y in M
15. Goto step 2
16. return M

Figure 5.6: The Hungarian method finds the max-weight maximum matching for a given bipartite graph, which was generated as a transformation of the assignment problem.

5.3 Identification based on Previous Positions

The tracking system estimates the locations of all players at every frame. To make use of these data, they have to be compressed in an adequate representation that is suited for the task of re-identification. In contrast to dimension reduction, the data have to be reduced in terms of their number.

5.3.1 Modeling positions by their mean

The law of large numbers from probability theory states that the average of the results running the same experiment a large number of times becomes closer to the expected value as more trials are performed. The experiment in our case consists in drawing the position of each player from its distribution by localizing it with the tracking system. We assume that the spatial distribution of each player is the same for the whole game (despite its reflection around the center after each halftime). Although this assumption does not hold in practice because athletes usually adapt their play to their competitors, it is a fairly reasonable approximation.

The average position $\overline{m}_{1:l,j}$ of player j is computed as the sample mean of the positons of the most likely estimates provided by the tracking system up to the past time l

$$\overline{m}_{1:l,j} = \frac{1}{l} \sum_{k=1}^{l} m_{k,j}^{\arg\max_i p\left(x_k^i | z_{1:k}\right)}. \tag{5.5}$$

An example for the average positions is depicted in figure 5.3(d).

Taking the probability distribution of positions as unimodal, the Hungarian method constitutes the method of choice for labeling the input such that the overall distance from input to expected values is minimized. The weights of the bipartite graph are thus given as

$$w(i,j) = \left\| (x_i, y_i) - \overline{m}_{1:l,j} \right\|. \tag{5.6}$$

Alternatively, the pdf of the positions can be modeled by a normal distribution with the same mean but additional unbiased sample variance

$$\overline{V}_{1:l,j} = \frac{1}{l-1} \sum_{k=1}^{l} \left(\arg\max_i p\left(x_k^i | z_{1:k}\right) - m_{k,j}^{\arg\max_i p\left(x_k^i | z_{1:k}\right)} \right)^2. \tag{5.7}$$

The weights for the Hungarian method are changed to the Mahalanobis distance of each input position i to the learned Gaussian distribution j

$$w(i,j) \sim \sqrt{\left((x_i, y_i)' - \overline{m}_{1:l,j}\right)' \overline{V}_{1:l,j}^{-1} \left((x_i, y_i)' - \overline{m}_{1:l,j}\right)}. \tag{5.8}$$

In practice, these distances take small values and must thus be scaled, since the Hungarian method requires the weights to be nonnegative integers. For speed-up, we compute the squared Mahalanobis distance instead, omitting the expensive square root operation, while still minimizing the same distance.

5.3.2 Learning vector quantization of player positions

The modeling of the players' spatial distribution by their expected value (mean) has some shortcomings. Contrary as one might understand, the expected value

of a distribution does not always coincide with the most probable one. Regarding multimodal distributions, this even makes only the exceptional case. In this section, we assume the spatial distribution of a single player to be multimodal. Therefore, we apply vector quantization, which extracts multiple prototype vectors intending to match the modes of the underlying distribution.

Related work

Different methods for extracting the modes from training data have been proposed (c.f. section 6.1.3). In this section, we will focus on artificial neural networks which follow the competitive learning paradigm.

Growing Neural Gas (GNG) was introduced by Fritzke [82] as an incremental and unsupervised neural network for vector quantization including a similarity graph. The prototype vectors are represented as neurons with an attached weight vector $\mathbf{w}$. Starting with two neurons, GNG grows in regular time intervals λ up to a maximum size θ. Connections between neurons are created by topology preserving Competitive Hebbian Learning [171] and form the similarity graph. Only the neuron that is closest to the input (also known as the best matching unit and abbreviated BMU) and its direct topological neighbors are updated towards each input signal leading to lower time complexity than comparable approaches (e.g. Self-Organizing Maps [139] or Neural Gas [172]), where the total network is updated. All used learning parameters of GNG are constant and enable the handling of infinite input streams. The complete algorithm is depicted in figure 5.7. The exponential decay of past values in the update step favors samples from the recent past over initial values, depending on the learning rates ϵ_b and ϵ_n. This behavior supports an adaption to (slowly) changing distributions and is therefore well suited for approximating the spatial distribution of the players.

GNG-U has been introduced as a variant of GNG for non-stationary data distributions [83]; it deletes neurons according to their utility of approximating the target distribution. This modification allows estimating prototypes for fast changing distributions. Kyan and Guan proposed Self-Organized Hierarchical Variance Map (SOHVM) [146] as another incremental network. It showed better accuracy than GNG because of the higher complexity (nodes include also the sample covariance computed by incremental PCA). Unfortunately, SOHVM lacks the capability of online processing due to declining parameters.

LateGNG

For the purpose of re-identification, prototypes must be built for the probability distribution of each player's location. Growing Neural Gas would constitute the

GNG-Learn-PDF($p(\mathbf{x}), \epsilon_b, \epsilon_n, \gamma, \eta, \lambda, \theta$):

1. time variable $t = 1$
2. initialize neuron set
   $\mathcal{K} = \{a, b\}$, $e_a = e_b = 0$, $\mathbf{w}_a \sim p(\mathbf{x})$, $\mathbf{w}_b \sim p(\mathbf{x})$
3. initialize connection set $\mathcal{E} \subseteq \mathcal{K} \times \mathcal{K} = \emptyset$
4. LOOP
5. Draw $\mathbf{x}_t \sim p(\mathbf{x})$
6. find winner $r = \arg\min_{n \in \mathcal{K}} \|\mathbf{x}_t - \mathbf{w}_n\|^2$
7. find second winner $s = \arg\min_{n \in \mathcal{K} \setminus \{r\}} \|\mathbf{x}_t - \mathbf{w}_n\|^2$
8. GNG-Update($\mathbf{x}_t, t, r, s, \epsilon_b, \epsilon_n, \gamma, \eta, \lambda, \theta$)
9. $t = t + 1$

GNG-Update($\mathbf{x}, t, r, s, \epsilon_b, \epsilon_n, \gamma, \eta, \lambda, \theta$):

1. increment error of r: $e_r = e_r + \|\mathbf{x} - \mathbf{w}_n\|$
2. connect r with s: $\mathcal{E} = \mathcal{E} \cup \{(r, s)\}$
3. $age_{(r,s)} = 0$
4. increment the age of all edges connected with r
   $age_{(r,n)} = age_{(r,n)} + 1 \quad (\forall n \in \mathcal{N}_r \setminus \{s\})$
5. remove old connections $\mathcal{E} = \mathcal{E} \setminus \{(a, b) | age_{(a,b)} > \gamma\}$
6. delete all nodes with no connections
   $\mathcal{K} = \mathcal{K} \setminus \{n | \forall k \in \mathcal{K}.(n, k) \notin \mathcal{E} \wedge (k, n) \notin \mathcal{E}\}$
7. update r and its direct topological neighbors $\mathcal{N}_r$:
   $\mathbf{w}_r = \mathbf{w}_r + \epsilon_b \cdot (\mathbf{x} - \mathbf{w}_r)$, $\mathbf{w}_n = \mathbf{w}_n + \epsilon_n \cdot (\mathbf{x} - \mathbf{w}_i) \quad (\forall n \in \mathcal{N}_r)$
8. IF $t \mod \lambda \equiv 0 \wedge |\mathcal{K}| < \theta$
9. find neuron q with greatest error $q = \arg\max_{n \in \mathcal{K}} e_n$
10. find neighbor f of q with $f = \arg\max_{n \in \mathcal{N}_q} e_n$
11. new node l: $\mathcal{K} = \mathcal{K} \cup \{l\}$, $\mathbf{w}_l = \frac{1}{2}(\mathbf{w}_q + \mathbf{w}_f)$, $e_l = \delta \cdot (e_f + e_q)$
12. adapt connections $\mathcal{E} = (\mathcal{E} \setminus \{(q, f)\}) \cup \{(q, n), (n, f)\}$
13. $e_q = (1 - \delta) \cdot e_q$
14. $e_f = (1 - \delta) \cdot e_f$
15. decrease all errors $e_n = \eta \cdot e_n \quad (\forall n \in \mathcal{K})$

Figure 5.7: Growing Neural Gas (GNG) training algorithm for data pdf $p(\mathbf{x})$.

method of choice for this task. But if the GNG algorithm is applied continuously to the positions in an online manner, one encounters the unwanted behavior of overshooting the prototypes over their expected values. Decreasing the adaption parameters ϵ_b and ϵ_n cannot solve the problem, since this phenomenon is inherent to the GNG algorithm because the value of the current BMU is more strongly influenced by more recently learned data.

To overcome this problem, we propose LateGNG as a variant of GNG, which applies a delayed (or late) update of the weights. If the same neurons are repeatedly determined as best and second best matching units, the input vectors are accumulated and the cumulative update is deferred until a different neuron pair is selected. The LateGNG algorithm is depicted in figure 5.8. This change successfully enables LateGNG to cope with uniformly continuous input streams. Figure 5.9 illustrates the different behavior of GNG and LateGNG for a simulated signal. In contrast to GNG, the nodes of LateGNG are able to approximate continuous distributions in a stable way. As an additional benefit, this variant also exhibits better runtime performance than the original GNG since some computations are omitted if the same BMU is selected.

Identification

LateGNG provides a neural network that can learn and classify arbitrary data online. For the purpose of identification based on individual positions, a single LateGNG model is learned for each player based on the estimates of the tracking system during the game. To retrieve an association for unknown positions, we apply the Hungarian method to find an association that minimizes the distances of the unlabeled positions $\mathbf{x}_j$ to the corresponding best matching units of the LateGNG models of each player i

$$w(i,j) = \min_{n \in \mathcal{K}_i} \|\mathbf{x}_j - \mathbf{w}_n\|^2 . \tag{5.9}$$

To also model the dependencies between the player positions, we learned the complete formations with LateGNG. A single formation is represented as the stacked vector of player positions ordered by their identifier. Because this agglomerated information constitutes the training data, the network learns the spatial interdependence of the team members as well. We expect this model to capture more details of the players' positions compared to the other models, which take only uncoupled player positions into account. The optimal associations with respect to each prototype formation are determined again by the Hungarian method. However, the association corresponding to the overall minimum matching weight of all prototypes is taken as the final association.

LateGNG-Learn($\mathbf{x}, \epsilon_b, \epsilon_n, \gamma, \eta, \lambda, \theta, \lambda_a$):

1. IF $|\mathcal{K}| < 2$
2. $\quad \mathcal{K} = \mathcal{K} \cup \{n\}$ with $\mathbf{w}_n = \mathbf{x}$
3. $\quad t = |\mathcal{K}|, t_a = 1, \mathbf{x}_{acc} = \mathbf{x}, s_{last} = r_{last}, r_{last} = n$
4. ELSE
5. find winner $r = \arg\min_{n \in \mathcal{K}} \|\mathbf{x} - \mathbf{w}_n\|^2$
6. find second winner $s = \arg\min_{n \in \mathcal{K} \setminus \{r\}} \|\mathbf{x} - \mathbf{w}_n\|^2$
7. IF $r = r_{last} \wedge s = s_{last} \wedge t_a < \lambda_a$
8. $\quad \mathbf{x}_{acc} = \mathbf{x}_{acc} + \mathbf{x}$
9. $\quad t_a = t_a + 1$
10. ELSE
11. $\quad \mathbf{x}_{acc} = t_a^{-1} \mathbf{x}_{acc}$
12. $\quad$ IF $t_a = \lambda_a \wedge |\mathcal{K}| < \theta$
13. $\quad\quad$ Add new node l: $\mathcal{K} = \mathcal{K} \cup \{l\}$ with $\mathbf{w}_l = \mathbf{x}_{acc}$
14. $\quad\quad$ add connections $\mathcal{E} = \mathcal{E} \cup \{(r_{last}, l), (l, s_{last})\}$
15. $\quad$ ELSE
16. $\quad\quad \epsilon_b' = 1 - (1 - \epsilon_b)^{t_a}$
17. $\quad\quad \epsilon_n' = 1 - (1 - \epsilon_n)^{t_a}$
18. $\quad\quad$ GNG-Update($\mathbf{x}_{acc}, t, r, s, \epsilon_b', \epsilon_n', \gamma, \eta, \lambda, \theta$)
19. $\quad t_a = 1, r_{last} = r, s_{last} = s, \mathbf{x}_{acc} = \mathbf{x}$
20. $t = t + 1$

Figure 5.8: Late Growing Neural Gas (LateGNG) is a variant of GNG that handles uniformly continuous input streams.

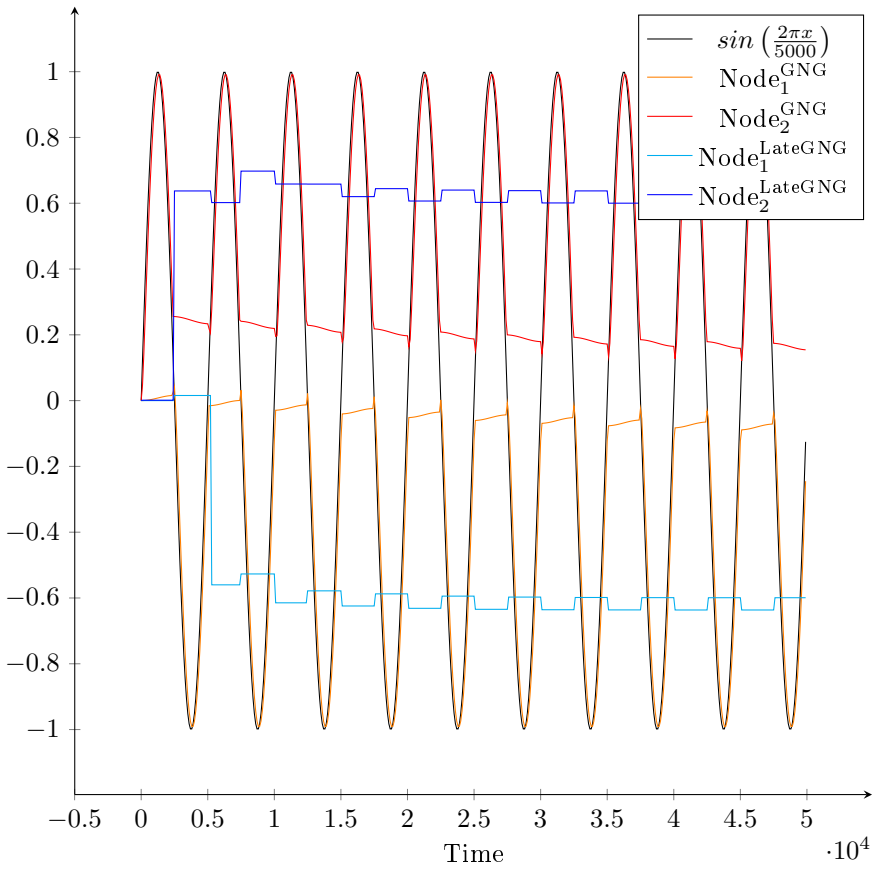


Figure 5.9: Comparison of GNG and LateGNG quantizing a one-dimensional sinoidal signal. The graph depicts the values of the input and of each node of GNG and LateGNG over time. Best matching units of GNG tend to follow the signal and overshoot the right cluster center. LateGNG handles this problem by delaying the update resulting in a stabilized behavior with better approximation of the underlying data distribution.

5.3.3 Assigning and learning partial data

Since absolute spatial information is provided by models learned on previous positions, they can be utilized for the assignment of partial data as well. Partial data occur frequently if only a single camera is used to provide the total input of the system, only capturing a fraction of the playing field. In this case, the estimates of the positions of unseen players are also inaccurate; therefore the quality of the models that are trained using these data is reduced as well.

The Hungarian method requires the same number of nodes in each part of the bipartite graph. If the number of positions does not match the number

of players, artificial nodes must be generated for the association graph to balance these numbers. The weights of edges starting at these artificial points are initialized with the maximum weight such that the min-weight matching still provides a solution to the original minimization problem. The assignment of positions that exceed the targets by number can be handled analogously by artificial nodes on the target side with zero weighted connections to all input nodes.

One could also ask for the identifiers of a subset of the players only, assuming that the identities of the rest are known. This scenario occurs when an unknown player comes into sight of the single camera that is capturing the game. This single player must be initialized with an identifier, while the other visible players are already assigned. This problem can be solved in the case of independent player models by applying the Hungarian method to the subset of unknown players and remaining identities or models respectively. Obviously, solving this minimization needs less computational time than the uninformed case. Special care must be taken for models representing complete formations, because these models cannot be reduced in their dimensionality. First, the model that fits the formation, which is induced by the known identities, best must be searched. Given this model, the best association of the unknown players can be found in the same way as in the case of independent player models.

If not all players are visible all the time, player models must provide the possibility for learning from partial data. This limited visibility is given almost permanently for broadcasted video. Because a simple motion model is a feasible approximation for the real movement of players only if restricted to short time windows, the estimated locations for the unseen players become quickly inaccurate and are therefore not suited for learning. Hence, higher order models like formations are not qualified for this scenario, but the representations for individual players like mean, Gaussian or GNG and LateGNG with multiple prototypes can still be utilized, while only the models of players in sight are updated.

5.3.4 Support of probability distributions

The methods proposed so far provide the single best association of the positions. However, to make use of this information in the tracking process, a probability distribution over identities is needed for each position (or measurement respectively). Obviously, the Dirac function at the classified association can be used to form a proper pdf, but it is too restrictive for most cases. Although this procedure seems to provide the only choice for models based on relative distance, alternative methods can be applied for absolute models.

If the player models have been learned separately for each athlete, the probabilities can be deduced directly from the normalized distances to these models. In the case of vector quantization, only the distances to the corresponding best matching unit are taken into account.

For complex models like complete formations, the distributions cannot be achieved that easily because the associations are not independent of each other. Instead, a Monte Carlo Markov Chain (abbreviated as MCMC) sampler based on the Metropolis-Hastings algorithm must be applied to turn an arbitrary distance metric into a probability distribution.

The MCMC algorithm is designed to sample from arbitrary distributions that can be evaluated easily at given points, but which may not be known in total. The method sequentially provides its internal state as a next sample. The state space is traversed pursuant to proposals that change this state locally if accepted with an acceptance ratio α. The probability distribution can be obtained by the reciprocal relative frequencies of the sampled states. The Metropolis-Hastings algorithm is depicted in 5.10. A so called burn-in phase, where the samples are excluded from frequency counting, is assumed to be needed for the algorithm to be independent of the initial state. As a key property of the MCMC method, the target pdf π must be known only up to a normalizing factor and must be evaluated solely on sampled states, which avoids the infeasible listing of the complete state space.

In our case, the complete formations are taken as state space and the reciprocal distance to the given or learned model forms the un-normalized probability distribution. The proposals for the Metropolis-Hastings algorithm denote changes in the ordering of the given positions or the swapping of single associations uniformly at random. The probability distribution for a single position can be computed as the frequency of the assigned identities inside the sampled sequence.

5.4 Evaluation

We evaluated the different models on the spatial data of 21 games of the first Bundesliga during the 2008/09 season played by the FC Bayern München team. Prior to the evaluation of the proposed methods, we define a performance measure to compare the algorithms and provide a worst case algorithm as the basis of this comparison.

Metropolis-Hastings($p(x), N, N_{\text{burn-in}}$):

1. Draw $x_0 \sim p(x)$ from prior density
2. FOR $i = 1 : (N + N_{\text{burn-in}})$
3. Draw $x' \sim q(x|x_{i-1})$
4. Draw $u \sim \mathcal{U}(0,1)$ uniformly at random
5. $\alpha(x', x) = \begin{cases} \min\left(\frac{\pi(x')q(x,x')}{\pi(x)q(x,x')}, 1\right) & \text{if } \pi(x)q(x,x') > 0 \\ 1 & \text{otherwise} \end{cases}$
6. IF $u < \alpha(x', x_{i-1})$
7. $x_i = x'$
8. ELSE
9. $x_i = x_{i-1}$
10. return $\{x_i\}_{i=N_{\text{burn-in}}}^{N+N_{\text{burn-in}}}$

Figure 5.10: Metropolis-Hastings algorithm to sample from an arbitrary pdf.

5.4.1 Performance measure

We select the cumulative probability function $\Pr(x \leq k)$ for the occurrence of failures as performance measure. This measure provides the probability that the algorithm in question results in less or equal than k failures on the average. The higher this probability for smaller values of k the better the performance of the algorithm, since it results in fewer failures on the average.

5.4.2 Random association as worst case algorithm

The algorithm, which assigns each position to a player randomly, provides the theoretical lower bound for the identification performance. Every algorithm that performs worse than this random algorithm should be rejected. The performance of the random method can be evaluated analytically by looking at relative frequencies. The probability that a random association will result in exactly k failures is given as the ratio of the number of associations with k failures and the total number of associations. A single association can be represented by a specific ordering (or permutation) of the numbers in $1 \ldots N_t$ with the total number of players N_t. The total number of associations is thus $N_t!$. The number of associations with exactly k failures can be evaluated as follows: we

select k indices out of the N_T indices that form the correct association and count the number of permutations of this k-element set that leave no element fixed, thus really resulting in exactly k failures. These derangements of length k are given by the subfactorial of k written as $!k$ (see [32, p. 118]). The subfactorial $!k$ can be computed recursively by $!0 = 1, !1 = 0, !k = (k-1)\,(!(k-1)+!(k-2))$ as proven by Euler [66] or can be evaluated as $!k = k! \sum_{i=0}^{k} \frac{(-1)^i}{i!}$. Putting the parts together, the cumulative probability for exactly k failures in a random association is

$$\Pr(x \leq k) = \sum_{i=0}^{k} \frac{!i\binom{n}{i}}{n!} = \sum_{i=0}^{k} \frac{\sum_{j=0}^{i} \frac{(-1)^j}{j!}}{(n-i)!}. \tag{5.10}$$

5.4.3 Initialization performance

In this section, we investigate the ability of the different non-learning approaches to provide good initialization for the tracking module. We concentrate on identifying all players at kick-offs by approaches based on the tactical lineup only. Potential methods are 1) sorting by formation lines, which first sorts the positions longitudinally and afterwards, the partitioned lines laterally, 2) minimizing the relative distances by the Hungarian method and 3) sorting by position, which sorts mainly on the y-coordinate but on the x-coordinate inside a given threshold, which was set to $\theta_{XY} = 3.0$ meters.

Figure 5.11 depicts the probabilities of having fewer than x failures for each approach. Kick-offs at the beginning of a halftime and an average of all kick-offs, including the ones during the games after a goal, are shown separately.

It turned out that methods which are based solely on relative spatial information outperform the sorting approach that is based on absolute positions. This method also has the disadvantage of demanding the threshold parameter θ_{XY} to be set, while the optimal value in respect to performance varies for every game and is hard to guess. Sorting by formation lines exhibited the best identification performance while providing a fast and easy-to-implement algorithm.

The performance of every approach exceeds the uninformed strategy of random association by far. Despite this fact, one would expect a higher overall identification rate since the positioning of the players at kick-off time should match the tactical line-up. This assumption obviously does not hold: the players performing the kick-off are so close to each other, as they typically never are, and the wing midfielders of the team with ball possession stand literally on the midline – and therefore on the same y-coordinate as the strikers – to speedily advance into the opponent's half after the opening whistle. Additionally, the lineups extracted from broadcast or the web are inaccurate or even incorrect: A 4-4-2 diamond formation, also written as 4-1-2-1-2, is often displayed as an ordinary 4-4-2, which causes assignment errors in the center midfield. The lineup

of the second halftime may also be different from the first: besides substitutions, red cards and the resulting reduction of the number of players on the field causes a reorganization in play. Taking into account that we cannot make hard assumptions on the accuracy of the input model, the proposed coarse methods show fairly good performance.

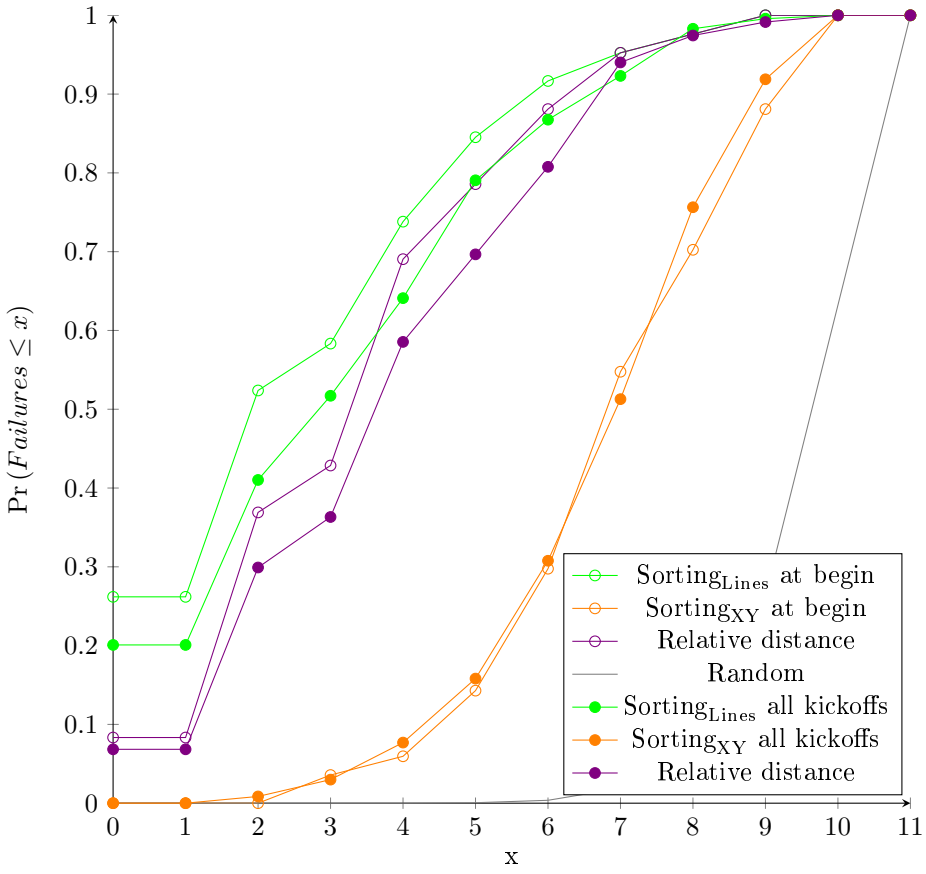


Figure 5.11: Mean failure rates of different approaches for initialization based on tactical lineups only. Visualized is the cumulative probability function which evaluates the probability that the number of failures is less than or equal to x. Higher values on the left state better performance.

5.4.4 Overall identification performance

We evaluated the adequacy of all methods, this time also including the learned models, for both halftimes of all 21 games separately for each team. In order to give a comprehensive view of the performance, we show the mean, the best

and the worst performances in figures 5.13, 5.14 and 5.15, respectively. The mean is taken as an average over all halftimes; best performance depicts the single halftime, where the algorithm under inspection encountered the highest probability of no failures. The halftime with the worst performance is selected analogously. All learning models have been trained on the first 2000 frames and evaluated for the rest of the game. Parameters for GNG have been set empirically to $\epsilon_b = 0.05$, $\epsilon_n = 0.005$, $\theta = 30$ $\delta = 0.1$, $\eta = 0.9$, $\gamma = 100$, with the additional parameter for LateGNG set to $\lambda_a = 15$.

As expected, all methods outperformed the random algorithm by far. The learned models of formations by LateGNG showed the best performance of all models throughout. This is not surprising since these models incorporate the most information about the players' locations, including the dependencies between them. The second best approach is sorting according to the lineup, which had fairly robust results. Despite its simplicity and use of relative relations only, it outperforms the other learning models clearly on the average. It is noteworthy that LateGNG, when learning player positions, showed a very good best performance compared to the others, but only medium quality on the average. One can see that LateGNG outperforms GNG in all cases, which emphasizes the effectiveness of our enhancement of the original algorithm.

The individual position models applying the Euclidean distance dominated the Mahalanobis distance significantly, which is astonishing since it is based on less knowledge. We explain this behavior with sideward overfitting of the Gaussian models: positions of the players are estimated badly and the variance degenerates to a narrow ellipse in a specific direction, if the game started mostly on one specific side. Figure 5.12 depicts an example of this situation.

The best classification rate with zero failures in 40% of the time is promising, but the average case is lower, at around 10%. This is due to the variability in the play inherent to sports. Soccer teams try to change their tactics if they are not successful, therefore increasing the problem of automatic re-identification. Substitutions and conjunct changes in the assignment of roles degrade the identification rate – especially in second halves of the games. Sometimes the interchange of players is part of a team's tactics, with the aim being to confuse the opponent as well as any automatic identification approach. The number of wrong associations in our evaluation strongly depends on the conformity of the play with either the tactical lineup or the play at the beginning of the game. If the learning approaches are applied continuously, better results can be expected.

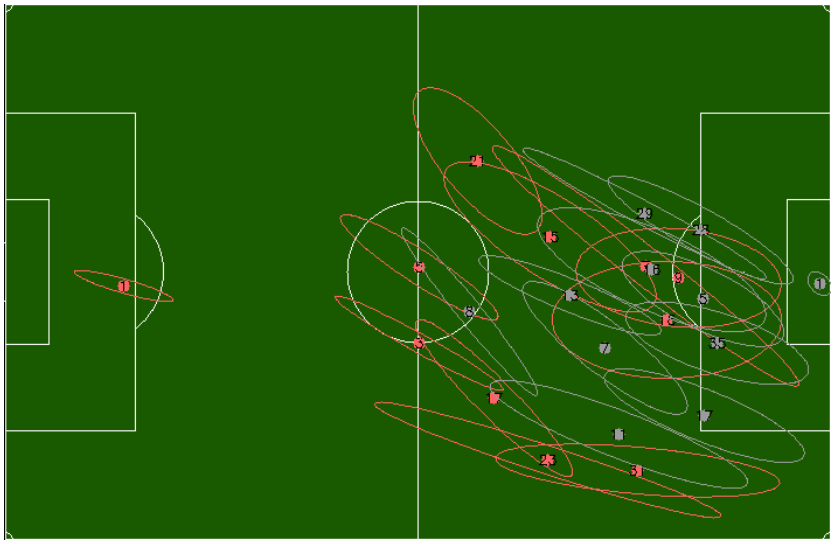


Figure 5.12: Gaussians with 1-sigma contours for all players gathered from the first 2000 frames of a second halftime of a single game. The covariances are very narrow, resulting in bad performance for identification.

5.4.5 Identification performance during the game

This section investigates identification performance with a focus on changes from a temporal perspective. For the sake of clarity, only the performance of the best algorithm (LateGNG learning formations) is shown in detail for the halftime where this method performed best. The average number of failures during a time window of 140 frames is depicted in figure 5.16. Four different types of events are visualized as vertical lines of different colors.

One can see clearly that the algorithm performs badly when corner kicks take place. This is not surprising because the formation of the players in such scenarios differs strongly from the normal play. Also, the identification performance is reduced during fouls or free kicks respectively.

The detection rate stays below four failures during the first ten thousand frames. In the middle of this halftime, the play gets more aggressive, as seen by the high number of fouls. As the behavior of the team changes, the identification method makes more failures. A low identification rate can be seen as an indicator for the change of play and could therefore also be used for analyzing tactics. We will look at this idea in more detail in section 8.2.

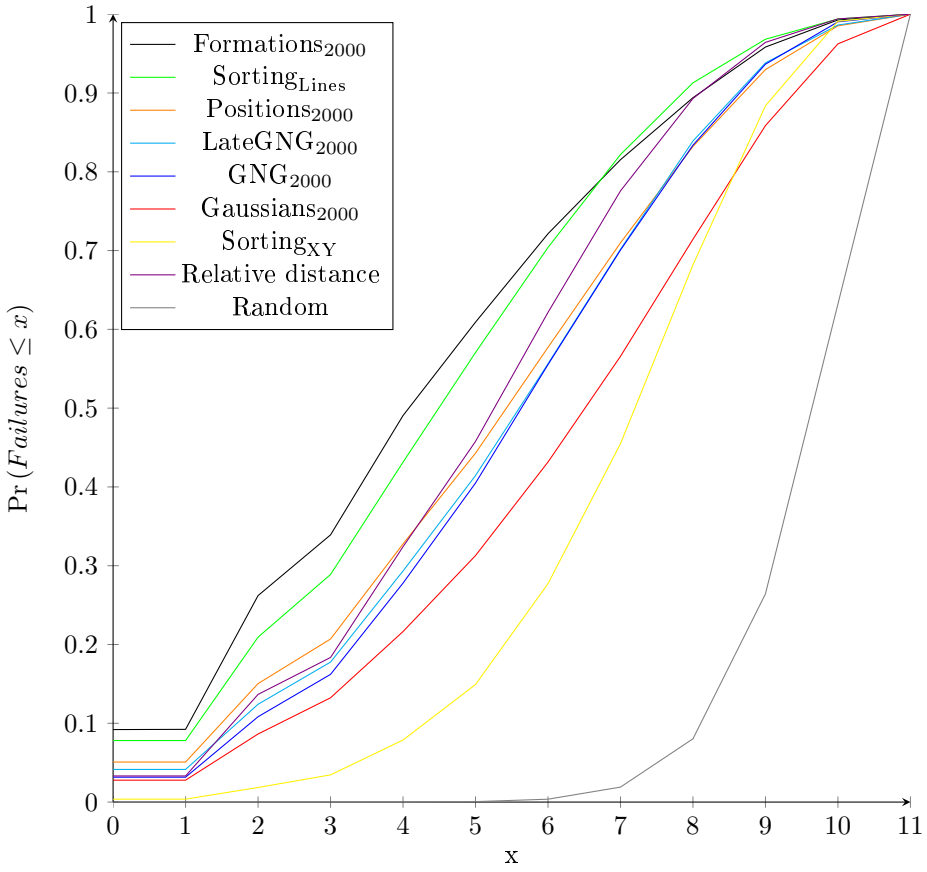


Figure 5.13: Mean failure rates of the different approaches for all games. Visualized is the cumulative probability function, which evaluates the probability that the number of failures is less than or equal to x. Higher values on the left state better performance.

5.4.6 Identification performance according to roles

The identification performance is different for each player role. For evaluation, we selected two games played by FC Bayern München with an identical lineup. The team played a 4-4-2 diamond formation, which is a common formation in soccer today. 4-4-2 diamond denotes the formation of playing with four defenders, a defensive and three offensive midfielders and two forwards/strikers.

The classification rate is enumerated for each player in the upper part of table 5.1 for one game and in the lower part for the other game. One can see that although the club played against different teams, the identification rates are nearly the same. Besides the $\text{Sorting}_{\text{XY}}$ strategy, all approaches show

Player role	F	S_L	P	LGNG	GNG	G	S_{XY}	R
goalkeeper	98%	97%	96%	98%	98%	84%	96%	97%
left fullback	67%	53%	58%	65%	53%	51%	3%	55%
left center back	70%	56%	56%	70%	54%	42%	41%	50%
right center back	78%	67%	58%	77%	64%	41%	7%	55%
right fullback	81%	71%	66%	77%	64%	47%	14%	71%
defensive midfielder	22%	33%	41%	30%	30%	34%	28%	34%
left midfielder	64%	62%	50%	57%	52%	49%	12%	62%
central midfielder	27%	42%	35%	33%	33%	29%	17%	45%
right midfielder	52%	55%	56%	34%	37%	27%	27%	56%
left forward	30%	31%	30%	32%	23%	22%	32%	24%
right forward	36%	39%	37%	31%	39%	34%	49%	31%

Player role	F	S_L	P	LGNG	GNG	G	S_{XY}	R
goalkeeper	94%	94%	94%	94%	94%	88%	92%	94%
left fullback	79%	78%	68%	82%	65%	38%	35%	79%
left center back	79%	73%	65%	80%	59%	42%	37%	74%
right center back	83%	73%	67%	83%	57%	62%	31%	71%
right fullback	85%	77%	74%	81%	63%	64%	41%	77%
defensive midfielder	45%	38%	33%	33%	51%	30%	31%	39%
left midfielder	82%	66%	67%	79%	50%	22%	26%	67%
central midfielder	39%	41%	30%	33%	22%	21%	32%	42%
right midfielder	77%	63%	67%	77%	48%	38%	39%	66%
left forward	47%	18%	30%	47%	36%	32%	22%	15%
right forward	53%	24%	32%	51%	38%	16%	35%	21%

Table 5.1: Comparison of the different approaches for each player role of a 4-4-2 diamond soccer formation during two games with the same lineup but different opponents. The approaches are abbreviated as follows: Formations_{2000} (F), $\text{Sorting}_{\text{Lines}}$ (S_L), Positions_{2000} (P), LateGNG_{2000} (LGNG), GNG_{2000} (GNG), Gaussians_{2000} (G), $\text{Sorting}_{\text{XY}}$ (S_{XY}), Relative distance (R).

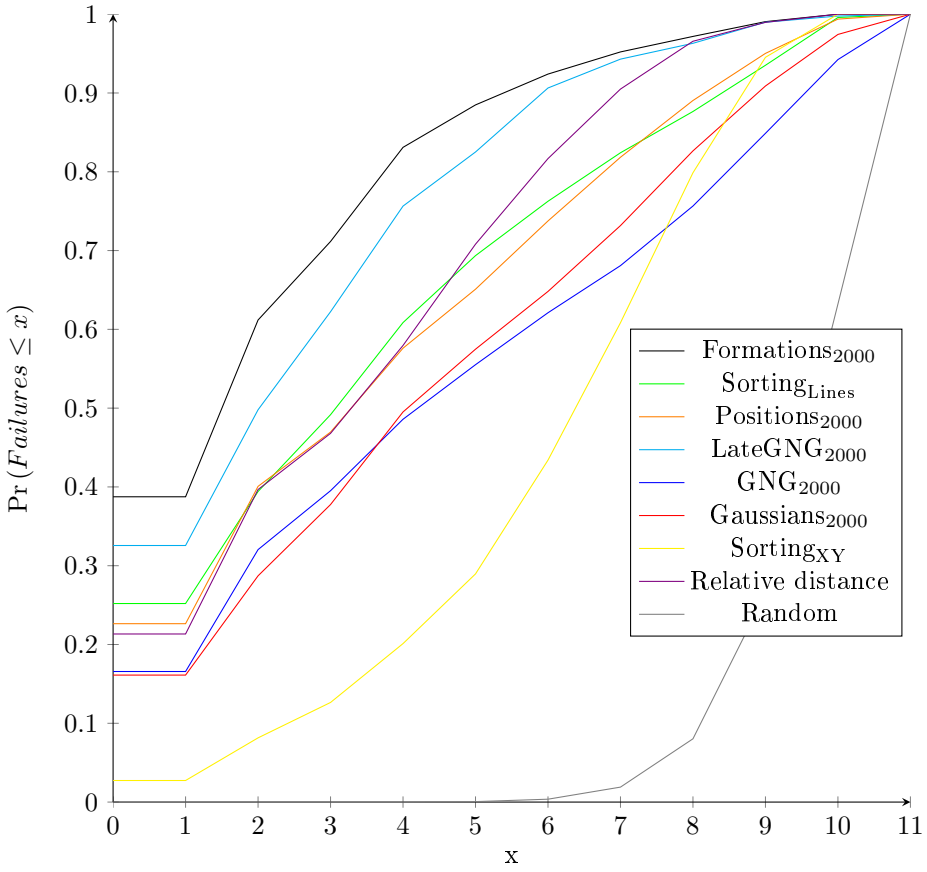


Figure 5.14: Failure rates of the different approaches for the halftime in which each algorithm performed best. Visualized is the cumulative probability function which evaluates the probability, that the number of failures is less than or equal to x. Higher values on the left state better performance.

similar distributions, implicating a systematic error induced by play rather than method.

The goalkeeper was identified correctly throughout, because he is for the most part allocated the most distinctive role. The defensive line also seems to be more easily assignable. This is reasonable since defending roles are stricter in terms of responsibility for a spatial area. The higher classification rates for the left and the right midfielder can be explained by the same rationale. Low identification of the central midfielders might be due to a less strict role or could also be caused by their special additional roles as the taker of corners for the offensive and the team captain for the defensive midfielder respectively. The lower rates of the forwards can be explained by the higher flexibility of their

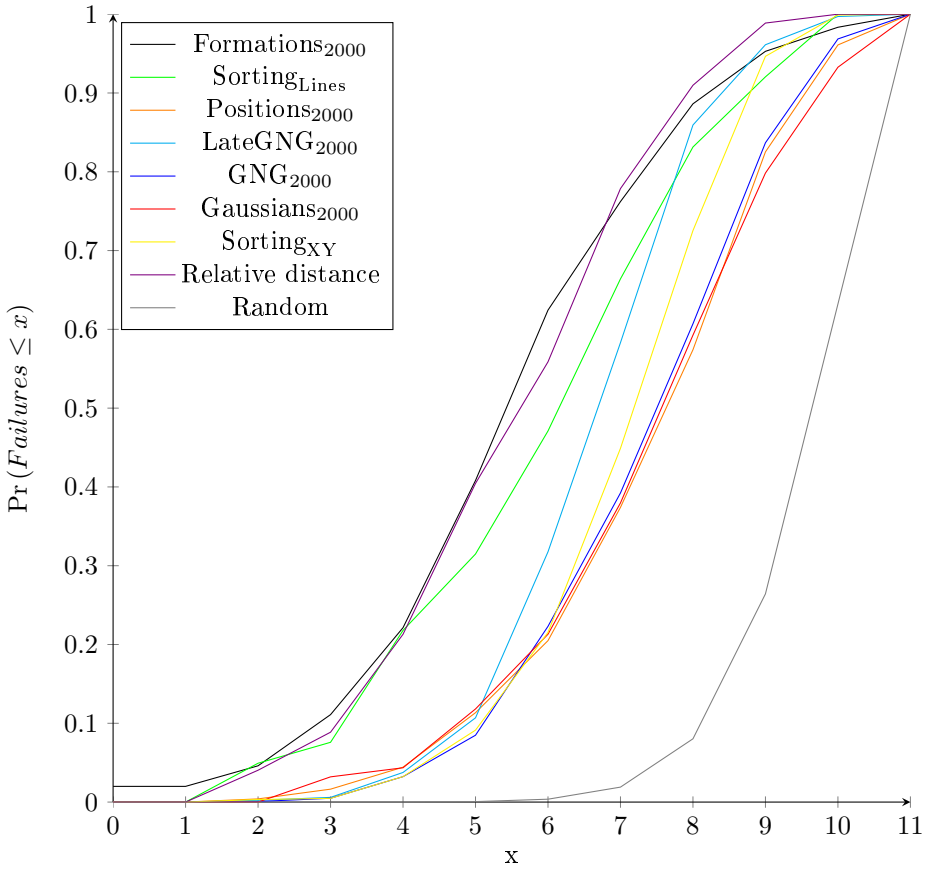


Figure 5.15: Failure rates of the different approaches for the halftime, in which the algorithm performed worst. Visualized is the cumulative probability function which evaluates the probability that the number of failures is less than or equal to x. Higher values on the left state better performance.

roles and a more frequent mix-up between them.

5.4.7 Runtime requirements

All methods exhibit real-time performance. The different runtimes for assigning and learning the positions of a single frame and a single team are depicted in table 5.2. Computational time was evaluated on an AMD 2.5GHz processor and averaged over a complete halftime. One can see that LateGNG computes an assignment twice as fast as GNG and learns four times faster than GNG. This is due to the deferred update, which bundles computations and results in fewer prototypes while increasing accuracy.

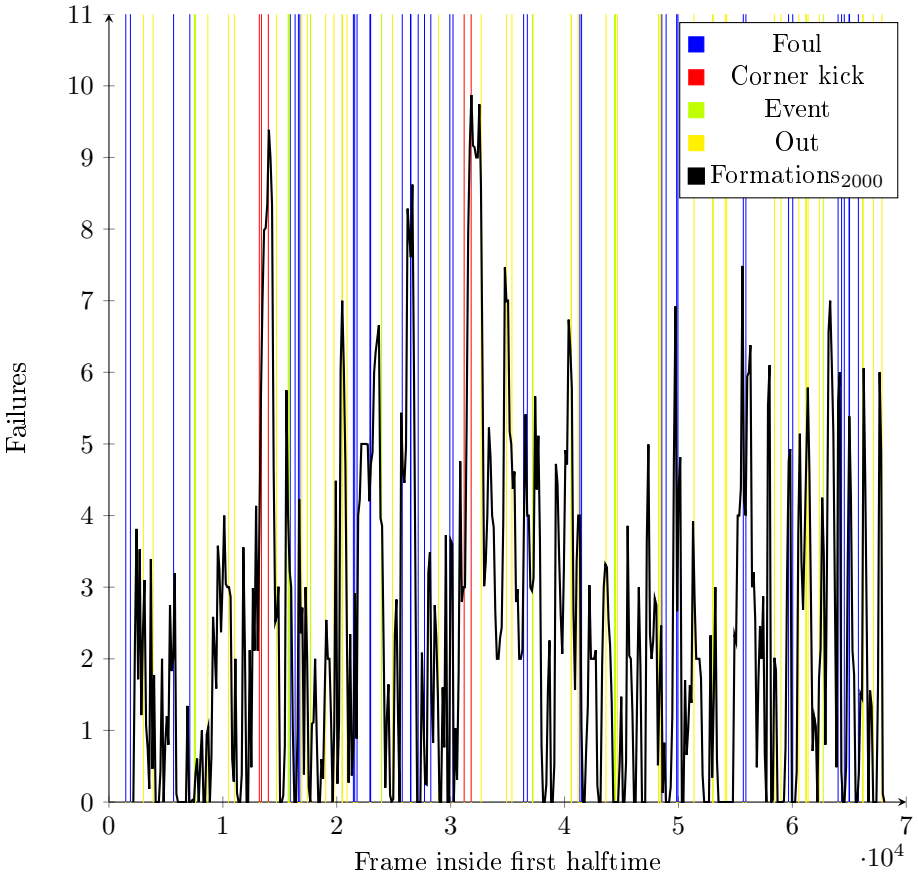


Figure 5.16: Identification rate of learned formations from a temporal perspective, also showing breaks of the game. The graph depicts the moving average of the failures during a time window of 140 frames.

5.5 Conclusions

In this chapter, we proposed several innovative methods to re-identify players based on spatial information only. The computational task is not trivial, since spatial variability is inherent to tactics in sports. The tactical lineup extracted from other information sources like the World Wide Web can be useful for initializing the multi-target tracking module. We applied two main approaches to solve the identification problem: special sorting for relative positions and minimizing the difference of the total association to a predefined model according to some distance measure. The minimization is solved using the Hungarian method, after transforming the problem into a search for a min-max weight graph matching. Different distance measures for models with rising complexity

Approach	Assignment	Learning
$\text{Sorting}_{\text{XY}}$	0.009ms	-
$\text{Sorting}_{\text{Lines}}$	0.009ms	-
Relative distance (R)	0.049ms	-
Positions_{2000} (P)	0.086ms	0.002ms
Gaussians_{2000} (G)	0.074ms	0.002ms
GNG_{2000} (GNG)	0.198ms	0.028ms
LateGNG_{2000} (LGNG)	0.105ms	0.007ms
Formations_{2000} (F)	0.100ms	0.029ms

Table 5.2: Runtime comparison of different methods for identification by position. Times are given for a single frame and a single team and were evaluated on a AMD 2.3GHz processor.

have been investigated. Inaccurate lineups motivated the use of learning approaches. These complex models are therefore learned online and incrementally in real-time by aggregation or by the use of growing self-organizing networks. We enhanced state-of-the-art Growing Neural Gas (GNG) towards continuous (ordered) data and demonstrated the improvement of the resulting LateGNG empirically.

Evaluation of real games revealed that most players could be identified quite well, while learned formation prototypes provided the best model for doing so. In addition to formation learning, special sorting presents a competitive algorithm that is very fast and easy to implement. According to our experiments, this relative sorting method is preferable for initializing the tracker based on a tactical lineup when no data could have been learned so far. Evaluation revealed that corner as well as free kicks decrease identification performance. Also, the more defensive the player role, the more easily the player can be assigned.

Our scientific contributions consist in novel approaches for identifying players using relative sorting based on the tactical lineup, by transforming the search for the best association according to a given model to a search for the min-max weight graph matching solved by the Hungarian method and by proposing LateGNG as a novel extension of Growing Neural Gas towards an online vector quantization method for continuous data.

Chapter 6

Appearance-based Identification

Appearance denotes the characteristic outward aspect of a person. It is composed of shape, which denotes the spatial extension, and texture, which represents the physical characteristics of the surface. In addition, the individual change of these features over time can be subsumed under the concept. Standard cameras capture solely two-dimensional projections of the appearance; these appear as colored regions in the image and constitute the exploitable visual information at a specific time. The computational task of this chapter is to extract the appropriate probability distributions for the assignment of identities to single regions of a specific frame or frame sequence. The task is visualized in figure 6.1.

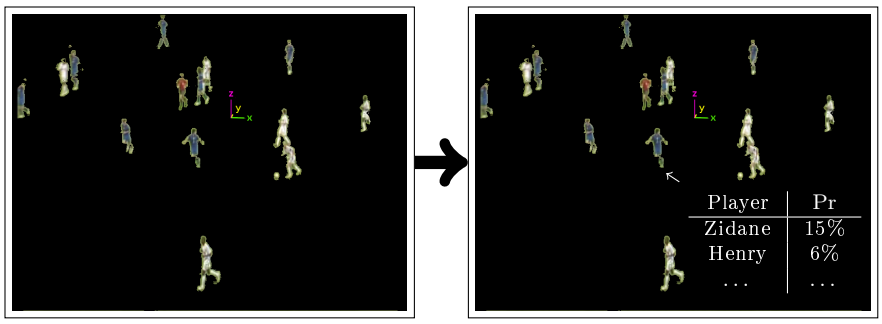


Figure 6.1: Potential player regions are labeled with the probability distribution for their identity assignment.

While localization was merely based on shape, identification relies heavily on color and texture. Since images contain only partial views of the underlying appearance, we propose building models for each player incrementally, hence

agglomerating these data over time. In this chapter, we discuss different models of increasing complexity for identification purposes, while attention is paid to the real-time capabilities of the investigated methods.

6.1 Related Work

After a review of published work for identification of athletes in sports videos, the literature for identification in general is surveyed according to the categorization of Wang and Suter [257].

6.1.1 Face and jersey number recognition in sports

Re-identification of players at a distance is based on recognition of exclusive features like faces, jersey numbers or gait. Yang et al. [272] find a specific person in broadcasted video by classifying the transcript and utilizing face detection. Bertini et al. [29] identify soccer players in close-ups based on face, jersey number and overlay recognition, where player names and faces are correlated automatically by the system. They apply an Adaboost face detection; recognition is achieved by matching SIFT descriptors. Jersey number recognition as well as the interpretation of superimposed text captions is achieved by optical character recognition (OCR) applied to maximally stable extremal regions. Faces are learned given the extracted textual cues (name or jersey number) in a supervised way. Although they report acceptable results, the problem of correlating these close-ups with the player position is apparent for the systematic identification of tracked players.

Ye et al. [273] recognize jersey numbers on several types of sports video by voting for the tracked number after a dozen frames using k-Nearest Neighbor classification. Andrade et al. [4] transform each frame into a picture tree as a Region Adjacency graph (RAG) to track players for the purpose of extracting their shirt numbers on broadcasted soccer videos.

The above-mentioned papers report good results for the identification during close-ups and selected scenes. Typical scenes of a recorded match, however, provide material that is more difficult to handle, because faces and jersey numbers occasionally occupy just a few pixels, are heavily contorted or only visible for a short period of time. Exclusive features may not be found if the resolution of a single player drops below a certain threshold and the total player region is represented by only a limited number of pixels.

6.1.2 Identification of humans at a distance

We follow the framework of Wang and Suter in [257] to review the work on re-identifying humans in image sequences at a distance. Therefore, the identification process is composed of three basic steps: feature extraction, dimension reduction and classification. These steps are reviewed in more detail in the next sections.

Feature extraction

An appearance is qualified by texture, shape and its change over time (named as motion). Features represent specific aspects, agglomerations or abstractions of the plain image regions that could be helpful to reveal the correct identity. The selection of adequate features has a great impact on later identification performance.

Color supplies the first feature for re-identification because this information is directly encoded in the image. A color denotes characteristic wavelengths of visual light. Many different color spaces have been proposed for representation until now (see [221] for an overview). Some general-purpose color models are RGB, YUV and HSV, but also task-specialized ones as the adapted hybrid color space [247] exist. Colors are represented mostly as three-dimensional byte vectors with 256 distinct values per channel.

Shape seems to be an unsuited feature for the purpose of identification as shapes may be nearly identical at a distance, regardless of the corresponding identity. It does, however, serve as prior information for other features. Shape can be described as a connected and closed region inside the image or its contour respectively. These regions are detected by splitting the foreground, which could have been extracted aided by one of the various foreground segmentation methods mentioned in section 3.3, into several connected components by repeated region growing [11]. Representations for shapes are manifold: they range from basic binary masks, run-length encoded regions and skeletons [109] to silhouettes which are also known as contour models. These contours are described occasionally by polar coordinates, snakes, B-splines, projections onto the bounding box [64], point distribution models or active shape models [53]. Wang et al. provide a survey of the different representations for shapes [255, 256], while a more comprehensive overview, including recent work, is outlined by [176].

In the context of computer vision, texture refers to the distribution of colors inside an area or a so-called pattern. It combines spatial and color-coordinated information and is usually represented by histograms or bags of image patterns (see f.i. [226]). Often, statistics are gathered only at special points like corners or edges. SIFT [165] or SURF [21] are popular features for texture following

that approach.

Time-varying appearances are modeled in two ways: on the one hand, a model of shape variation is combined with a model of texture (or color) variation as done by active appearance models [52] or, on the other hand, several specific appearance regions are tracked forming a complex body model [198].

6.1.3 Dimension reduction

Because the features for appearance constitute a huge amount of data that must be processed and classified for every point in time, several dimension reduction techniques have been applied to reduce the computational demand of classification.

Histogram analysis is a basic method that is frequently used to quantize a high dimensional space. Histograms constitute relative or absolute frequencies of the data that are matched to a selection of bins. A general way to define these bins is achieved by a uniform discretization of the data space. More sophisticated methods for selecting appropriate bins are based on a distance measure. The binning is called quantization and is often done via nearest neighbor classification to prototype vectors. These vectors can be defined by a set of predefined examples also called templates or determined in an unsupervised way. Cluster analysis builds such representative positions in the high dimensional space (which are called clusters) annotated with additional model parameters based only on a training set. Clustering techniques state a wide research area in unsupervised learning: hierarchical, agglomerative and iterative methods exist. Vector quantization denotes a variant of clustering that assigns data to mutually exclusive prototypes using the winner-take-all principle. A variety of methods for vector quantization have been proposed as there are the Lloyd algorithm (better known as k-Means clustering) [164], Gaussian Mixture Models (GMM) extracted by Expectation Maximization (EM) [56], Self-organizing Maps (SOM) [139] or Neural Gas (NG) [170] with its variant Growing Neural Gas (GNG) [82].

Other approaches try to directly extract an information preserving transformation of the training data to a lower dimensional space. Exponents of these are the Principal Component Analysis (PCA) [243], Non-negative Matrix Factorization (NMF) [148, 149], isometric feature mapping (ISOMAP) [237] and Local Linear Embedding (LLE) [208] (with no intention to be exhaustive).

In addition, polar histograms [126] and histograms extracted at predefined rectangular sub-regions [182, 89] encode texture with a reduced dimensionality by also taking spatial information into account.

6.1.4 Classification

We refer to classification as the automated mapping of dimensionally reduced features extracted from a foreground region to player labels. Although most methods in the area of supervised machine learning are suited for this task, the need for online, real-time classification eminently reduces the number of choices. Hulten and Domingos [110] postulate an online data-mining system to provide constant time and memory per incoming datum, single scan of data, availability of a usable model at any time, equivalence to an ordinary data-mining algorithm, and adaption of time-varying concepts, while preserving unchanging concepts. In the field of image recognition, only nearest neighbor classification as well as decision trees have been applied to real-time online learning and classification until now.

The k nearest neighbor classification (k-NN) approach determines the resulting class by majority voting of the k closest labeled templates or examples to the date in question. If $k = 1$, the k is omitted and the method is just called nearest neighbor classification. Odd ks usually outperforms even ones due to voting without draws. Online learning is achieved by substituting or updating templates during execution. The method is simple to implement but costly if the number of templates and/or dimensions are high. The costs of finding the closest template can be softened by constructing efficient search structures or can be precluded by the use of effective dimension reduction as a preceding preprocessing step. Kirstein et al. [137] applied k-NN to image prototypes acquired by vector quantization of a visual hierarchical model for the purpose of online object recognition in real-time.

Decision trees represent undirected acyclic graphs with simple conditions at inner nodes and class labels at leaves (c.f. [211]). Classification is done by top down traversal beginning at the root, taking the decision for the left child if the condition holds for the record in question and the right child otherwise; the class label at the ultimately reached leaf provides the result. Decision trees provide fast methods if they are balanced, because only a small number of decisions must actually be evaluated to classify a record. A tree is constructed by recursively selecting the conditions or features that are best suited for discriminating the data for classification based on entropy or the Gini measure. Typically, trees are pruned based on statistics afterwards to reduce model size and avoid overfitting.

Several approaches building decision trees in an online manner have been proposed in the literature. Incremental Decision Tree Induction (ITI) [245] is based on efficient restructuring of a given decision tree to incorporate the new training examples as they arrive. The Very Fast Decision Tree algorithm (VFDT) [85] builds upon the Hoeffding tree algorithm. The split conditions are

selected based on sub-sampling of a stationary data stream. The number of sufficient examples is estimated by exploiting the Hoeffding inequality to achieve a probabilistic bound on the accuracy of the constructed tree. VFDT exhibits higher accuracy than ITI, according to [85]. The Incremental On-Line Information Network (IOLIN) [49, 48] incrementally updates an Information Network (also called Info-fuzzy Network). This multi-layered network is constructed in order to test the Mutual Information (MI) between the input and target attributes. The operations for updating the network include checking the validity of conditions, examining the replacement of the last layer and adding new layers as necessary. IOLIN shows superior accuracy to VFDT, according to [49].

Song et al. [226] learned and applied decision and regression trees for classifying image patches to identify soccer players. Regular non-incremental decision-tree-learning algorithms were utilized to learn stubs from limited training data. These stubs are constantly discarded and spawned according to their prediction accuracy. The persistent forest that incorporates all these stubs forms the incremental classification model. Unfortunately, no insights concerning the runtime of this approach are given in the paper [226], so the real-time criterion remains questionable. Due to the intricacy inherent in player re-identification, this method has problems distinguishing players of the same team as stated in [226]:

> For the SCEPTRE [soccer] dataset, we discovered that when some similar appearance players merged or split, our method might fail due to the similar score map obtained by the classifiers. Majority of the failed tracking in this dataset was caused by this condition.

6.2 Identification based on Color

There are typically five different appearances in ball games: the dress of both teams, both goalies and the referee. The official FIFA rules [76] for soccer state:

> The two teams must wear colours that distinguish them from each other and also the referee and the assistant referees. Each goalkeeper must wear colours that distinguish him from the other players, the referee and the assistant referees.

These appearances refer mainly to the players' colors, which is the case in most types of sports with opposing teams. Although individual players of the same team wear the same colors, they may still differ in hair and skin color or by the footgear worn. Each video image is typically represented as a matrix of color values, so the direct use of this information for the purpose of identification is

obvious. The colors at the player regions selected by the localization module of section 3.5 are the feature which is investigated in this section. These colors are quantized into histograms which constitute the individual player models for subsequent identification. We review quantization methods and histograms, before our approach for identification by color is explained in detail.

6.2.1 Color quantization

Because the number of three-dimensional color values inside a player region is on the order of ten thousands, an effective and efficient dimension reduction technique is necessary to reduce this information. As an additional advantage, this process enables the resulting model to be invariant against affine transformations and minor changes in lighting conditions. Color quantization is the process of reducing a large set of colors to a small palette.

The popularity algorithm was invented in 1978 by Tom Boyle and Andy Lippman for this task and published in [104]. The method simply selects the k most frequent colors as the color palette. Octree quantization [91] is a fast and fairly accurate algorithm that builds an octree data structure of the colors. Octrees are undirected acyclic graphs, where every inner node has eight children that are indexed by the bits of the three-dimensional color at the corresponding level. The leaves are labeled with the frequency of traversal; the trees are pruned by repeatedly collapsing the sub-trees with the lowest frequencies until the number of leaves is smaller than the given k. In 1997, Denis Lee provided the DL3Quant algorithm as source code distribution only (it was probably developed as part of IBM's QBIC system [80]): the algorithm applies quantization on the upper two bits of each color band and reduces the resulting bins by repeatedly merging the two bins, which approximately induces the minimum error compared to the other merges, until the desired grade of quantization is achieved. Median cut [104] is a hierarchical clustering method that splits the color space recursively into rectangular partitions at median planes. Dekker proposed NeuQuant [55] as a competitive color quantization approach based on Self-ordering Maps. The Lloyd algorithm (k-Means) [164] is widely used for color quantization and iteratively moves k cluster centers in an expectation maximization manner until a stopping criterion (usually a threshold on the movement of the clusters) is matched. Scheunders gives a comparison of k-Means and competitive learning approaches (CL) applied to color image quantization in [214]. Hautamäki surveys several color quantization algorithms in his master thesis [103].

6.2.2 Color histograms

Color histograms represent appearance by frequencies of matching the colors of all pixels inside the region of interest to a vector of bins. Histograms are invariant to affine transformations if they are normalized. Several distance measures for histograms are surveyed and compared by Rubner et al. in [209]. According to their results, the χ^2 distance performed best for histograms with fixed bins; the Manhattan distance (also known as L_1-norm) showed good results as well. For adaptive histograms with variable bin size or different binning, the earth mover's distance (EMD) [210] (which equals the Mallows distance [152]) performed best and dominated the approaches for fixed bins in the case of small number of bins. Similar to the edit distance for strings, EMD computes the minimal amount of "earth" that must be moved such that two histograms converge. It can be applied to arbitrary distributions. Because the distance computation includes solving a minimization problem, it is more costly by far than the other distances. The fast version $\widehat{EMD}$ [187] reduces the computational complexity to empirical $\mathcal{O}(n^{2.3})$ by thresholding distances between the bins and simplifying the minimization problem (The worst time complexity is $\mathcal{O}(n^2 U \log(n))$ assuming integral supply and demands that are bounded by U).

6.2.3 Identification of individual players

The re-identification of individual players is achieved as follows: an individual reference histogram is initialized for each player and constantly updated. The histogram of colors inside the region of an unknown player is compared to these reference histograms according to a distance measure. Individual players are classified using the nearest neighbor algorithm by selecting the best-matching label. In our case, the reciprocal normalized distances of the histogram in question to the ones of the players provide a probability distribution for the tracking process. Team affiliation states a reduced problem. It can be seen as a rough identification because it assigns equal probability to all members of the classified team and zero to the others.

The individual reference histogram of a specific player is filled with colors from the region that maximally overlaps with the expected bounding box above the player's position. This box can be determined according to a rectangle with typical extension of humans which is projected into the image at the current position estimate provided by the tracking system. Contamination of the histograms, which could be introduced by occlusions, is avoided by incorporating regions only if they belong to a freestanding player and exhibit an area inside the expected boundaries (c.f. section 3.5). The restrictions are not applied to unlabeled regions which should be identified only. The histograms are normal-

ized before distance computation to compensate for the difference in the number of pixels used to build them. As Gengembre and Pérez remark for the soccer domain in [90], "the initial reference histogram is common (the colors of the team that can be introduced in a color based detector) but some differences (e.g. hair color) [can be] learnt with [an] adaptation procedure". In our case, adaptation is achieved by incorporating new frequencies into the reference histograms, when an additional region of the corresponding player is detected in the current frame, and subsequent normalization.

Preliminary experiments have revealed that uniform spacing of the color space requires at least $40^3 = 64000$ bins to capture the different colors of the players, because these are – especially for players of the same team – very similar to each other. Because of the high number of dimensions, the computation of distances to all reference histograms becomes too slow for real-time processing. Therefore, we employ color quantization to reduce the number of bins while preserving the accuracy. A single sparse histogram, which contains the frequencies of all colors of the filtered foreground regions, is maintained during an initial phase. After a sufficient number of regions has been extracted for each player, dimension reduction is applied once to this initial histogram. All reference histograms are then binned in the same way as the reduced initial histogram. Because previously unseen players come into sight, the initial histogram is extended by their colors. After a sufficient amount of regions is included for each of them, the initial histogram is quantized again and all reference histograms are rebinned. This re-quantization process can also be started at regular intervals to adapt the selection of bins to changes in illumination. Alternatively, a monitoring module can trigger this process when the distances between the reference histograms and the region in question exceed a given limit. This monitoring comes for free, since the distances are computed during the nearest neighbor search or the probability estimation anyway.

Preliminary quantization allows the use of the χ^2 distance. The χ^2 distance can be applied to histograms with fixed bins as suggested by the comparative survey of [209]:

$$\chi^2\left(H_1, H_2\right) = \sum_{i=1}^{n} \frac{\left(H_1(i) - \bar{H}(k)\right)^2}{\bar{H}(k)} \text{ with } \bar{H}(k) = \frac{H_1(k) + H_2(k)}{2}. \qquad (6.1)$$

This distance is utilized for the nearest neighbor search or the estimation of association likelihoods respectively.

We also evaluated the use of histograms with adaptive bins for each player. These require a quantization step for each player region in question and the use of the earth mover's distance for each identification. Beside its unacceptable computational expense ([209] restricted EMD to 32 bins not for nothing), the

accuracy was even (slightly) worse than that achieved by the proposed method. These results emphasize that the adaptive selection of the bins is more important for the classification rate than the use of the earth mover's metric.

The matching of color to its corresponding bin states the bottleneck of histogram initialization and update. We therefore use a sorted balanced binary tree with bins attached to the nodes for fast selection of the bin corresponding to a specific color. Because colors are three-dimensional, no unique one-dimensional ordering between them exists. Hence, we interleave the bits of the three color channels to provide a Morton ordering of the colors (c.f. [1]), which approximates the spatial ordering in 3D by a space-filling curve.

6.2.4 Evaluation

We compared different color quantization methods, namely Median Cut, Octree and DL3Quant, according to their influence on the resulting correct identification rate. A set of 1500 figure-centric images was automatically extracted from videos of the world championship final 2006 that were captured by two panning, tilting and zooming cameras. The videos constitute the clean feed for broadcast without cuts and overlays and the tactical zoom-out view (see section 7.3 for details). We split the image sets into a training set of 500 images and a test set with the remaining 1000 images. Quantization and histogram creation are based on the training set. Classification of the test images was achieved by nearest neighbor assignment. Due to automated extraction, the data also contain misaligned images; even wrong labels appear.

Table 6.1 shows the classification rates for three different quantization methods applied to two player resolutions. The Octree quantization significantly outperformed the others in terms of speed, also exhibiting higher accuracy. The difference in runtime, however, cannot be expected to be that much during online processing, because quantization is applied rarely; in addition, fewer data samples are typically used. The overall classification rate is around 41% on the average for players zoomed-in and 30% for the view at a distance. This rate increases with the number of bins for the broadcasted view and is fairly constant for the tactical view. Despite their clear supremacy over random assignment with an expected classification rate of $\frac{1}{23} = 0.04$, they are lower than typical rates of image recognition tasks. This emphasizes once again the difficulty of player identification, as this task requires classifying non-rigid and highly dynamic objects of varying and small sizes that contain little structure and are very similar in appearance.

Table 6.2 depicts the confusion matrix for classification of players taken from broadcasted view. The color histograms were quantized by the Octree method.

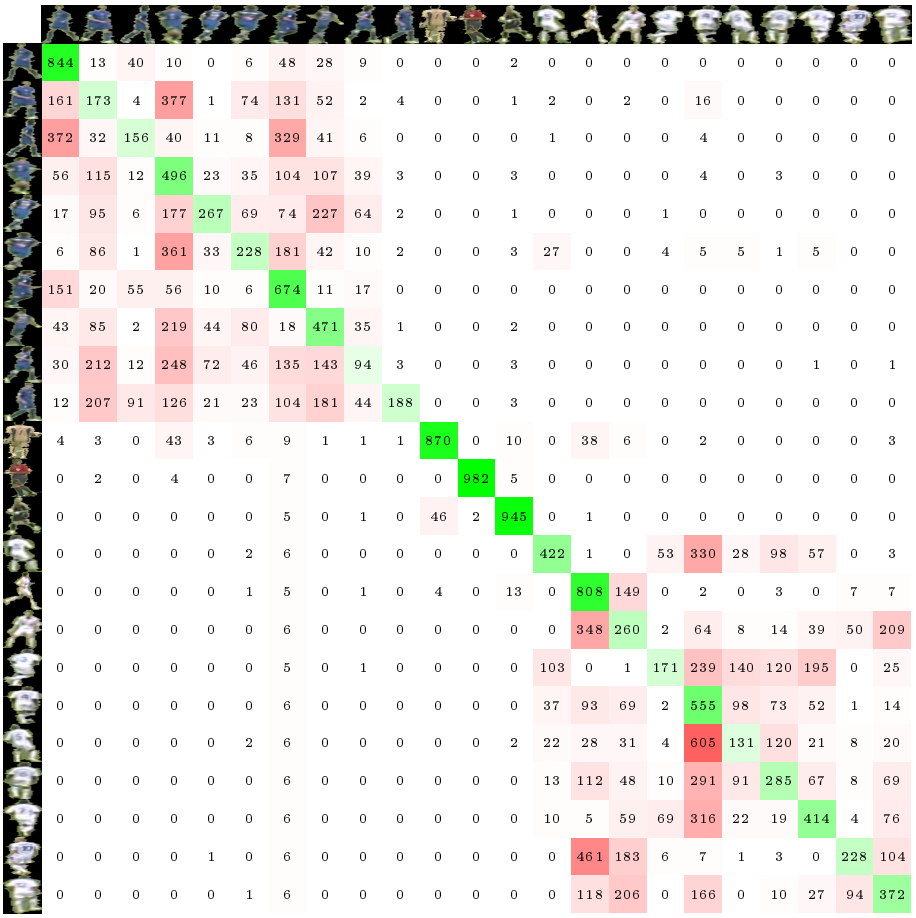
Bins		Tactical zoom-out view			Broadcasted view		
		Octree	DL3	MedianCut	Octree	DL3	MedianCut
64	Class	.309	.303	.297	.381	.311	.372
	Time	7ms	580ms	7884ms	14ms	8103ms	8524ms
128	Class	.305	.313	.273	.404	.359	.382
	Time	7ms	581ms	8643ms	15ms	8123ms	9411ms
256	Class	.301	.312	.234	.424	.369	.363
	Time	7ms	575ms	9812ms	15ms	8046ms	10764ms
512	Class	.312	.312	.307	.433	.359	.344
	Time	7ms	559ms	11016ms	15ms	8079ms	11797ms
1024	Class	.289	.287	.286	.436	.227	.335
	Time	6ms	515ms	12190ms	15ms	7849ms	12802ms

Table 6.1: Comparison of different color quantization methods for identification by color histograms. The table lists classification rates and the computational time needed for quantization.

Referee and goalkeepers are classified almost perfectly, but mix-ups occur for players on the same team. The reason for the misclassification is obvious: the players' appearance is simply too similar. The rare cases when players of different teams are confused may be due to motion blur, which smudges the colors, or defective segmentation of the test data during the gathering process.

To illustrate the performance of estimating the probability distribution of correct association, the difference between the trained (and normalized) histograms are depicted in table 6.3 for the broadcasted view and in table 6.4 for the tactical zoom-out view. Closer proximity of the histograms results in a more similar probability for the association with any player of the same team and degrade to team affiliation, if there is no difference between histograms of players of the same team. This behavior emphasizes the advantage of soft association based on an identity distribution over the hard classification, which would simply converge in the uninformed classification rate of $\frac{1}{23}$ or which may depend on the ordering of players in nearest neighbor classification. The distance matrices exhibit similar structure for both views. Members of the same team form a subgroup with lower distances. The distances reflect the similarity of the jerseys shown in the first row and column of the tables quite well.

Incorporating a single figure-centric image of size 160×120 into the histogram needed about 0.45 ms on the average, while the classification of one region took about 0.7 ms. During online processing, the player regions are not downsized, but all information is incorporated into the histogram, which



844	13	40	10	0	6	48	28	9	0	0	0	2	0	0	0	0	0	0	0	0	0	0
161	173	4	377	1	74	131	52	2	4	0	0	1	2	0	2	0	16	0	0	0	0	0
372	32	156	40	11	8	329	41	6	0	0	0	0	1	0	0	0	4	0	0	0	0	0
56	115	12	496	23	35	104	107	39	3	0	0	3	0	0	0	0	4	0	3	0	0	0
17	95	6	177	267	69	74	227	64	2	0	0	1	0	0	0	1	0	0	0	0	0	0
6	86	1	361	33	228	181	42	10	2	0	0	3	27	0	0	4	5	5	1	5	0	0
151	20	55	56	10	6	674	11	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0
43	85	2	219	44	80	18	471	35	1	0	0	2	0	0	0	0	0	0	0	0	0	0
30	212	12	248	72	46	135	143	94	3	0	0	3	0	0	0	0	0	0	0	1	0	1
12	207	91	126	21	23	104	181	44	188	0	0	3	0	0	0	0	0	0	0	0	0	0
4	3	0	43	3	6	9	1	1	1	870	0	10	0	38	6	0	2	0	0	0	0	3
0	2	0	4	0	0	7	0	0	0	0	982	5	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	5	0	1	0	46	2	945	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	2	6	0	0	0	0	0	0	422	1	0	53	330	28	98	57	0	3
0	0	0	0	0	1	5	0	1	0	4	0	13	0	808	149	0	2	0	3	0	7	7
0	0	0	0	0	0	6	0	0	0	0	0	0	0	348	260	2	64	8	14	39	50	209
0	0	0	0	0	0	5	0	1	0	0	0	0	103	0	1	171	239	140	120	195	0	25
0	0	0	0	0	0	6	0	0	0	0	0	0	37	93	69	2	555	98	73	52	1	14
0	0	0	0	0	2	6	0	0	0	0	0	2	22	28	31	4	605	131	120	21	8	20
0	0	0	0	0	0	6	0	0	0	0	0	0	13	112	48	10	291	91	285	67	8	69
0	0	0	0	0	0	6	0	0	0	0	0	0	10	5	59	69	316	22	19	414	4	76
0	0	0	0	1	0	6	0	0	0	0	0	0	0	461	183	6	7	1	3	0	228	104
0	0	0	0	0	1	6	0	0	0	0	0	0	0	118	206	0	166	0	10	27	94	372

Table 6.2: Confusion matrix of players based on color histograms from broadcasted view (quantized to 1024 bins by Octree). Stronger colors relate to higher confusion for convenience. Concrete numbers of classifications out of 1000 are depicted, too. Mix-ups predominantly occur between players of the same team who are similar in appearance.

is normalized in turn before each comparison. The computational demand is self-regulating because a more zoomed in camera leads to bigger regions but also fewer histograms that must be created and compared, since the number of visible and unoccluded players decreases.

Although the classification rate is obviously better for the broadcasted view with higher resolution than for the zoom-out view, confusion and distance matrices turn out to have similar structures. The distance matrix for the zoom-out view is shown in table 6.4. The decrease in performance is due to the lower en-

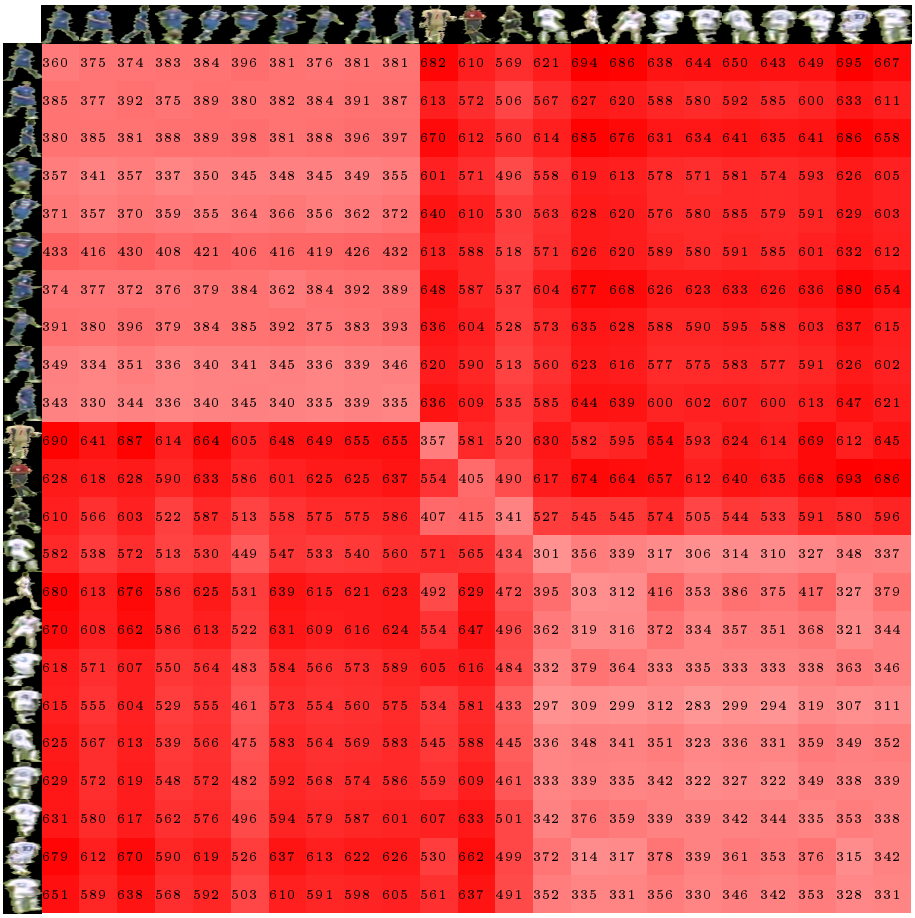
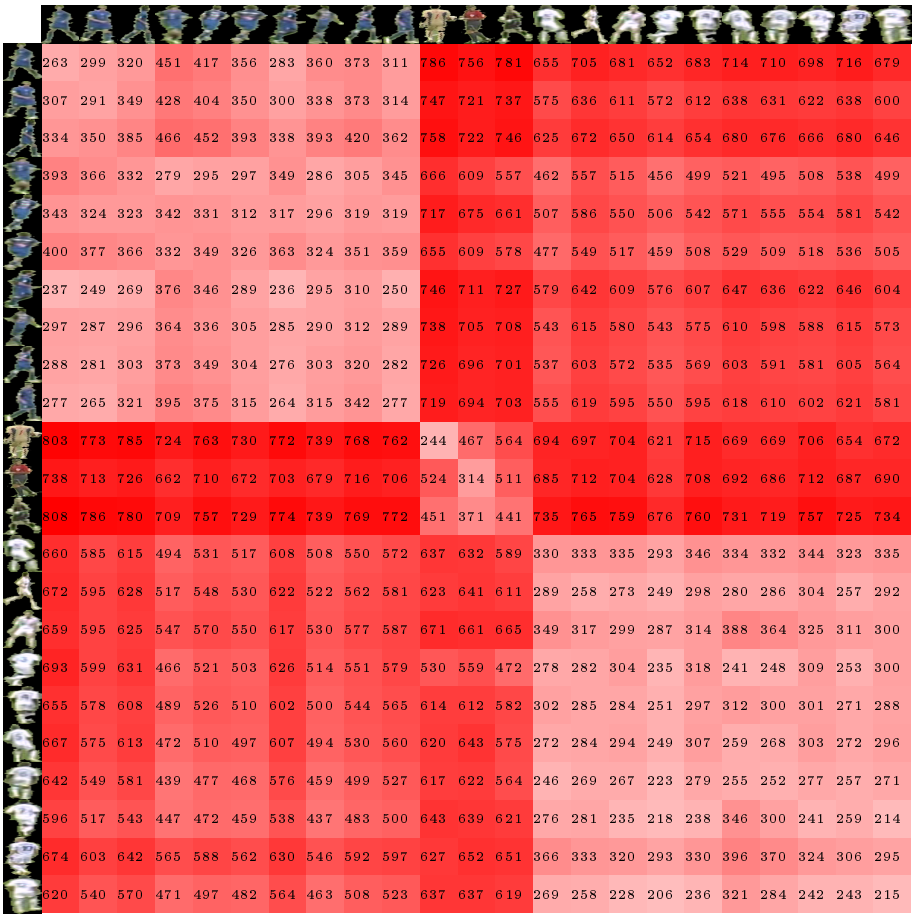


Table 6.3: Distance matrix between color histograms of each player captured from the broadcasted view and quantized to 1024 bins by Octree. Stronger colors relate to higher distance for convenience. Concrete distances are depicted as well. Distances between players of the same team are very low, which makes the identification task difficult.

tropy of the player regions captured in coarse resolution. The size of the players captured by the tactical camera are circa $10px \times 30px$ and players from the main center view vary in size from typically $30px \times 100px$ to zoomed $65px \times 290px$. The proposed method based on color histograms, which provides identity distributions, is fairly robust against the different scales.

We also investigated the use of very fast decision trees [85] for classifying the histograms. Preliminary results with VFDT show much lower performance than the nearest neighbor classification, however. This may be due to the high dimensionality compared to the small amount of training data, which makes it



263	299	320	451	417	356	283	360	373	311	786	756	781	655	705	681	652	683	714	710	698	716	679
307	291	349	428	404	350	300	338	373	314	747	721	737	575	636	611	572	612	638	631	622	638	600
334	350	385	466	452	393	338	393	420	362	758	722	746	625	672	650	614	654	680	676	666	680	646
393	366	332	279	295	297	349	286	305	345	666	609	557	462	557	515	456	499	521	495	508	538	499
343	324	323	342	331	312	317	296	319	319	717	675	661	507	586	550	506	542	571	555	554	581	542
400	377	366	332	349	326	363	324	351	359	655	609	578	477	549	517	459	508	529	509	518	536	505
237	249	269	376	346	289	236	295	310	250	746	711	727	579	642	609	576	607	647	636	622	646	604
297	287	296	364	336	305	285	290	312	289	738	705	708	543	615	580	543	575	610	598	588	615	573
288	281	303	373	349	304	276	303	320	282	726	696	701	537	603	572	535	569	603	591	581	605	564
277	265	321	395	375	315	264	315	342	277	719	694	703	555	619	595	550	595	618	610	602	621	581
803	773	785	724	763	730	772	739	768	762	244	467	564	694	697	704	621	715	669	669	706	654	672
738	713	726	662	710	672	703	679	716	706	524	314	511	685	712	704	628	708	692	686	712	687	690
808	786	780	709	757	729	774	739	769	772	451	371	441	735	765	759	676	760	731	719	757	725	734
660	585	615	494	531	517	608	508	550	572	637	632	589	330	333	335	293	346	334	332	344	323	335
672	595	628	517	548	530	622	522	562	581	623	641	611	289	258	273	249	298	280	286	304	257	292
659	595	625	547	570	550	617	530	577	587	671	661	665	349	317	299	287	314	388	364	325	311	300
693	599	631	466	521	503	626	514	551	579	530	559	472	278	282	304	235	318	241	248	309	253	300
655	578	608	489	526	510	602	500	544	565	614	612	582	302	285	284	251	297	312	300	301	271	288
667	575	613	472	510	497	607	494	530	560	620	643	575	272	284	294	249	307	259	268	303	272	296
642	549	581	439	477	468	576	459	499	527	617	622	564	246	269	267	223	279	255	252	277	257	271
596	517	543	447	472	459	538	437	483	500	643	639	621	276	281	235	218	238	346	300	241	259	214
674	603	642	565	588	562	630	546	592	597	627	652	651	366	333	320	293	330	396	370	324	306	295
620	540	570	471	497	482	564	463	508	523	637	637	619	269	258	228	206	236	321	284	242	243	215

Table 6.4: Distance matrix between color histograms of each player captured from tactical zoom-out view and quantized to 1024 bins by Octree. Stronger colors relate to higher distance for convenience. Concrete distances are depicted as well. Distances between players of the same team are very low, which makes the identification task difficult.

hard for the tree learning to find a good cut in the continuous color frequencies. In addition to the higher computational demand caused by additional training of negative examples, the problem of extracting a probability distribution remains unsolved when decision trees are used.

6.3 Identification based on Texture

This section investigates texture as the identification feature by exploiting colors in combination with spatial information.

6.3.1 Texture

We model texture by complete, figure-centric regions containing free-standing players. These regions, which are delivered by the foreground segmentation, are normalized to images of the same size that preserve their aspect ratio, and centered. Some examples are shown in figure 6.2. The idea is to keep a selection of textures for each of the players and perform nearest neighbor classification or estimate the probability distribution based on normalized distances respectively. We take the total texture into account because most texture-related features seem to concentrate on the pose or shape of the player instead of his exclusive details. Because this appearance model is much richer than the color histograms, one may expect better classification, but its use also requires more computational time.

(a) Zoom-in of Player (b) Player at a distance

(c) Motion blur (d) Imperfect segmentation

Figure 6.2: Figure-centric images constitute the texture used for identification.

The figure-centric idea is similar to the method in [61], where motion vectors are extracted from figure-centric soccer player image sequences. The motion has been used to recognize actions like walking left or running right at a distance. However, the authors Efros et al. remark:

> The main requirement is that the tracking be consistent – a person in a particular body configuration should always map to approximately

the same stabilized image.

Good stabilization is especially crucial for motion vectors but remains important for texture learning as well. Their paper presents very clear and well stabilized figure-centric sequences. Unfortunately, they do not detail how the preceding image extraction is achieved. The training sequences have been selected in a supervised way, discarding images which include overlapping or court marks. The same accuracy as given in their paper can hardly be achieved for arbitrary videos inside a bootstrapping process with unavoidable, imperfect segmentation due to the variety inside a game.

6.3.2 Vector quantization

The straight-forward approach for identification is to cache all textures for every player seen so far and compare these to the player region in question. However, this method reveals itself as impractical, even for short videos because it is increasingly expensive in terms of memory and computational complexity. Growing neural gas and our delayed variant LateGNG can be applied to reduce the incoming images to a constant number of prototypes online. The figure-centric normalized player images are vectorized by appending all rows and then fed to the self-organizing network of the corresponding player. In contrast to GNG, LateGNG requires additional conversion of the images from interleaved byte representation into a vector of floating points to avoid numerical problems like overflow during the summarization for the deferred update. Figure 6.3 shows the nodes of an exemplary LateGNG network that forms the texture model for a single player. The classification of player regions is achieved by nearest neighbor search in the models. During the tracking process, the normalized distances are inverted, thus forming an approximation of the identity likelihood, which is incorporated in the sampling probability of associations during the tracking process.

6.3.3 Evaluation

We compare three different approaches, namely the caching approach, GNG and LateGNG for identifying players by texture. The caching approach was evaluated for comparison purpose because it should exhibit the upper boundary of identification performance that can be achieved. We trained the models with 500 figure-centric images and tested them on another 1000 images. These images were automatically gathered during the tracking process. They also contain badly segmented players and even wrong identity assignments. Better results would be achieved with cleared training and test sets, but the evaluation

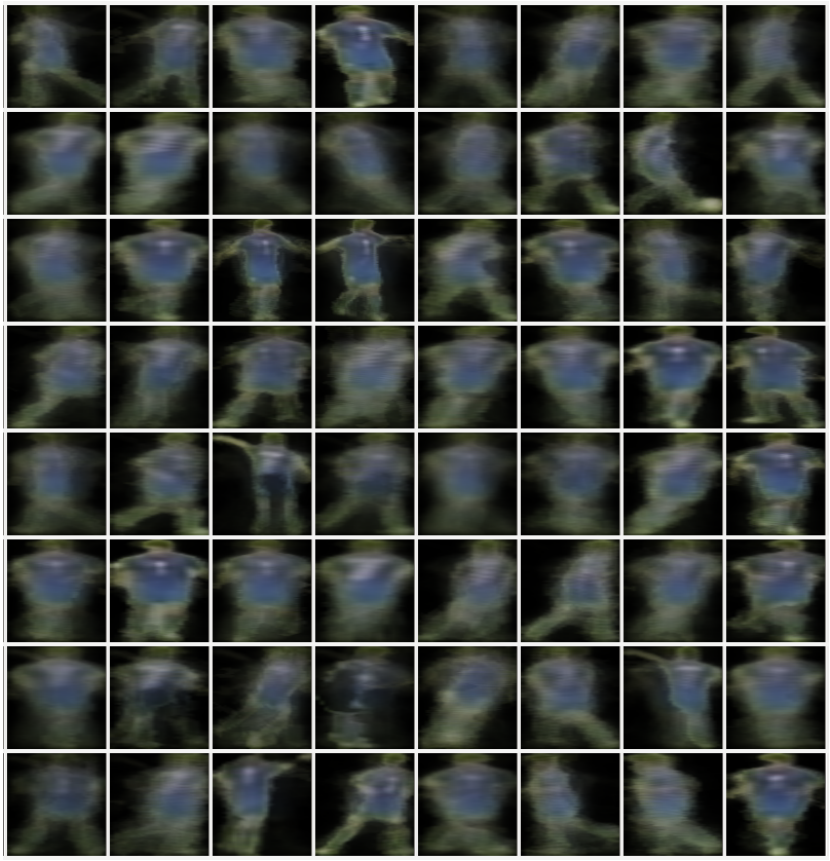


Figure 6.3: Texture model of a single player learned by LateGNG.

as presented here allows better predictions about identification performance during the real process of tracking in our cognitive system. We applied GNG and LateGNG for vector quantization of the data to a maximum of 64 prototypes. Runtimes for incorporating and classifying a single image were measured using a 2.5 GHz mobile CPU. The results for distant and medium zoom are depicted in table 6.5.

One can see that the identification based on caching all images exhibits – as expected – the best classification rate, while demanding the most computational time. However, the difference in identification is only marginal compared to GNG and LateGNG. Unfortunately, the classification rate is lower than that achieved by identification based on color histograms, albeit the prototypes in figure 6.3 demonstrate that the images are learned correctly. The lower identification rate can be explained by the high variability of the different poses that the color histograms abstract from. It is noteworthy that the identification rate

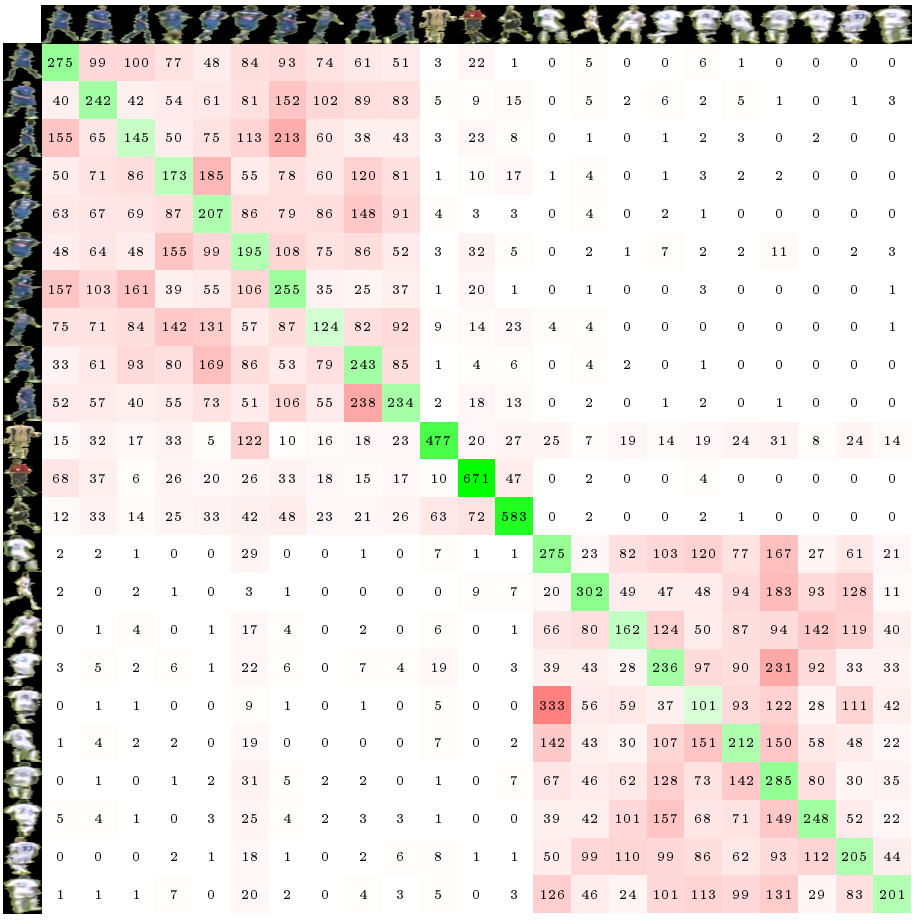
Image size		Tactical zoom-out view			Broadcasted view		
		Cache	GNG	LateGNG	Cache	GNG	LateGNG
160	Class	.281	.289	.288	.282	.275	.265
×	Time	783.1ms	64.7ms	28.1ms	618.3ms	64.3ms	27.6ms
120	Learn	0.85ms	3.44ms	1.57ms	1.07ms	3.39ms	1.61ms
80	Class	.275	.276	.270	.284	.275	.266
×	Time	205.1ms	18.8ms	8.2ms	235.6ms	18.8ms	7.9ms
60	Learn	1.00ms	1.98ms	1.18ms	1.34ms	1.95ms	1.21ms
53	Class	.279	.275	.269	.270	.272	.262
×	Time	94.2ms	10.0ms	4.4ms	94.6ms	9.9ms	4.4ms
40	Learn	1.03ms	1.68ms	1.11ms	1.06ms	1.69ms	1.14ms
40	Class	.278	.261	.264	.285	.274	.263
×	Time	56.6ms	6.8ms	3.0ms	56.7ms	6.8ms	2.9ms
30	Learn	1.03ms	1.57ms	1.07ms	1.05ms	1.64ms	1.09ms

Table 6.5: Comparison of different vector quantization methods for identification by total texture. The table lists classification rate and computational time for classification (Time) and learning (Learn) of a single figure-centric image of given size.

is fairly constant regardless of the resolution. This might be due to scaling from an originally lower resolution; it might also be caused by the abstraction of unnecessary details that compensates for information loss. The results for different views do not differ significantly, which might be due to the same reason.

The learned networks represent the images well, as can be deduced from the small reduction below 0.01 in the classification rate compared to the caching version. LateGNG performs slightly worse than GNG, which might be due to the higher averaging, while GNG more accurately fits the recent images. Differences in terms of identifiability between players are depicted in table 6.6. The confusion matrix was created using the LateGNG model with images of size 40×30 captured from a broadcasted view. Its structure is similar to table 6.2, which depicts a confusion matrix for identification by color histograms. Erroneous training data have a higher impact on the identification rate if vector quantization is applied to the figures in contrast to their representation by color histograms. Vector quantization creates a separate prototype for the misassigned player image since it differs from the other prototypes and that information needs a long time to be removed. Many misclassifications are due to this – actually correct – behavior of self-organizing networks.

Despite the marginal deficiency in classification rate, LateGNG is more than



275	99	100	77	48	84	93	74	61	51	3	22	1	0	5	0	0	6	1	0	0	0	0
40	242	42	54	61	81	152	102	89	83	5	9	15	0	5	2	6	2	5	1	0	1	3
155	65	145	50	75	113	213	60	38	43	3	23	8	0	1	0	1	2	3	0	2	0	0
50	71	86	173	185	55	78	60	120	81	1	10	17	1	4	0	1	3	2	2	0	0	0
63	67	69	87	207	86	79	86	148	91	4	3	3	0	4	0	2	1	0	0	0	0	0
48	64	48	155	99	195	108	75	86	52	3	32	5	0	2	1	7	2	2	11	0	2	3
157	103	161	39	55	106	255	35	25	37	1	20	1	0	1	0	0	3	0	0	0	0	1
75	71	84	142	131	57	87	124	82	92	9	14	23	4	4	0	0	0	0	0	0	0	1
33	61	93	80	169	86	53	79	243	85	1	4	6	0	4	2	0	1	0	0	0	0	0
52	57	40	55	73	51	106	55	238	234	2	18	13	0	2	0	1	2	0	1	0	0	0
15	32	17	33	5	122	10	16	18	23	477	20	27	25	7	19	14	19	24	31	8	24	14
68	37	6	26	20	26	33	18	15	17	10	671	47	0	2	0	0	4	0	0	0	0	0
12	33	14	25	33	42	48	23	21	26	63	72	583	0	2	0	0	2	1	0	0	0	0
2	2	1	0	0	29	0	0	1	0	7	1	1	275	23	82	103	120	77	167	27	61	21
2	0	2	1	0	3	1	0	0	0	0	9	7	20	302	49	47	48	94	183	93	128	11
0	1	4	0	1	17	4	0	2	0	6	0	1	66	80	162	124	50	87	94	142	119	40
3	5	2	6	1	22	6	0	7	4	19	0	3	39	43	28	236	97	90	231	92	33	33
0	1	1	0	0	9	1	0	1	0	5	0	0	333	56	59	37	101	93	122	28	111	42
1	4	2	2	0	19	0	0	0	0	7	0	2	142	43	30	107	151	212	150	58	48	22
0	1	0	1	2	31	5	2	2	0	1	0	7	67	46	62	128	73	142	285	80	30	35
5	4	1	0	3	25	4	2	3	3	1	0	0	39	42	101	157	68	71	149	248	52	22
0	0	0	2	1	18	1	0	2	6	8	1	1	50	99	110	99	86	62	93	112	205	44
1	1	1	7	0	20	2	0	4	3	5	0	3	126	46	24	101	113	99	131	29	83	201

Table 6.6: Confusion matrix between texture models of each player captured from a broadcasted view quantized by LateGNG. Stronger colors relate to higher classifications for convenience. Concrete numbers of mix-ups are depicted as well. Mix-ups of players on the same team are highly demonstrative of the difficulty of identifying these similar objects.

twice as fast in classification and also needs less time for learning a player region than the Growing Neural Gas. While the caching approach increases its computational demand with every image incorporated, the requirement of self-organizing networks converges to a time which will be slightly above the values shown in table 6.5. The classification is more costly in terms of runtime because the distances to all player models must be evaluated, while the learning step requires only the search for the best matching unit inside a single network. LateGNG is suited for processing both teams on a multicore CPU in real-time.

6.4 Gait as Appearance over Time

Experts for video-based tactical analysis (Dirk Sobottka, Executive Manager of Sports Analytics GmbH, personal discussion) reported that they also identify players according to their running style or gait. We therefore review the related work in this field and discuss an approach for identifying athletes at a distance by gait.

6.4.1 Gait recognition

Much work has been done in gesture and motion analysis of humans for surveillance and Human Machine Interface (HMI). Gesture analysis requires preliminary sequence alignment and matching of sequences of postures (shapes) that have been recognized in advance. Hu et al. survey visual surveillance of object motion and behaviors [108]. Hidden Markov Models (HMM) [197] are frequently used for probabilistic sequence alignment and action recognition. Weinland et al. [259] have proposed motion history volumes (MHV) as a free-viewpoint representation for human actions, where time is encoded by color in aggregated shape images.

Cassel et al. [40] recognize acrobatic gestures in gymnastics and trampoline competitions by matching position and orientation of the bounding box of the athlete region which was detected by block filtering. Sullivan and Carlsson [233] classify tennis postures in video by matching shapes against stored templates. Similar actions are recognized in broadcasted tennis by Roh et al. [206, 207]. The shapes are represented by curvature scale spaces (CSS); posture matching is achieved by RANSAC and the sequences are matched to templates by continuous dynamic programming (CDP). Classification of actions, transfer of 2D/3D skeletons as well as synthetization of novel action sequences in broadcasted low resolution videos has been proposed for ballet, tennis and soccer by Efros et al. [61]. Actions are recognized by nearest neighbor matching of local motion vectors gathered by optical flow on figure-centric image sequences to a database of learned actions. Shechtman and Irani [222] detect postures in images and recognize actions in videos based on local self-similarities, given only single examples of each action. They applied their method successfully to gymnastics, ballet and ice-skating.

In biometrics, human identification at a distance is done by analyzing gait patterns extracted from video; approaches are either based on a model (explicitly tracking parts of the body) or silhouettes. Ekinci [64, 63] represents shapes as distance vectors between the bounding box and silhouette; these are compressed by PCA and finally matched by nearest neighbor classification.

Hu et al. [108] stated in 2004 that "[a]lthough many researchers have been

working on gait recognition, current research of gait recognition is still in its infancy". Most of these approaches rely on very good foreground segmentation and sufficient resolution of the videos (often captured by several cameras in parallel). No gait recognition has been applied to the sports domain until now.

6.4.2 Identification by gait

Gait constitutes the characteristic motion during walking and running. These activities are cyclic by nature and can be modeled as a discrete Markov process, where shapes form the discrete states. This approach is similar to Schödl et al. [215], who introduce the term "video textures" as Markov processes of images. The foreground region masks can be taken as shapes; sequences of these shapes build the Markov process.

We tested several dimension-reduction techniques specific to shape representation: scaled images containing the complete raw mask, polar coordinates as distances of the contour to the centroid at several degrees as well as projections to the bounding box forming one-dimensional functions as in [63]. Figure 6.4 depicts a LateGNG network that has learned the player's shape. As expected, preliminary experiments showed that shape on its own is not an appropriate feature for distinguishing between players.

Merge Growing Neural Gas (MGNG) (see section 8.2.2 for detail) can be used to learn a probabilistic automaton for each player representing his gait by a Markov process. The idea is to track the best matching unit inside the network by recording the sequence of visited nodes and the frequencies of transitions. These statistics allow computing the likelihood of a sequence having been generated from the model. Unfortunately, we lack sufficiently long training sequences for each player to learn a proper model. The players frequently overlap or leave the field of view and therefore the sequence is interrupted after a short period. Long sequences were captured for standing players, but obviously no gait can be extracted from these data. Gait recognition therefore seems to be impractical for automated re-identification inside the cognitive tracking system. However, the model could be used for visualization or action recognition by matching against preselected sequences.

6.5 Conclusions

This chapter has investigated appearance as a feature for re-identification of soccer players. Prototypes of appearance are learned online for each player given sequences of figure-centric regions; the reciprocal, normalized distances to these models provide a probability distribution for the identity of an unassigned

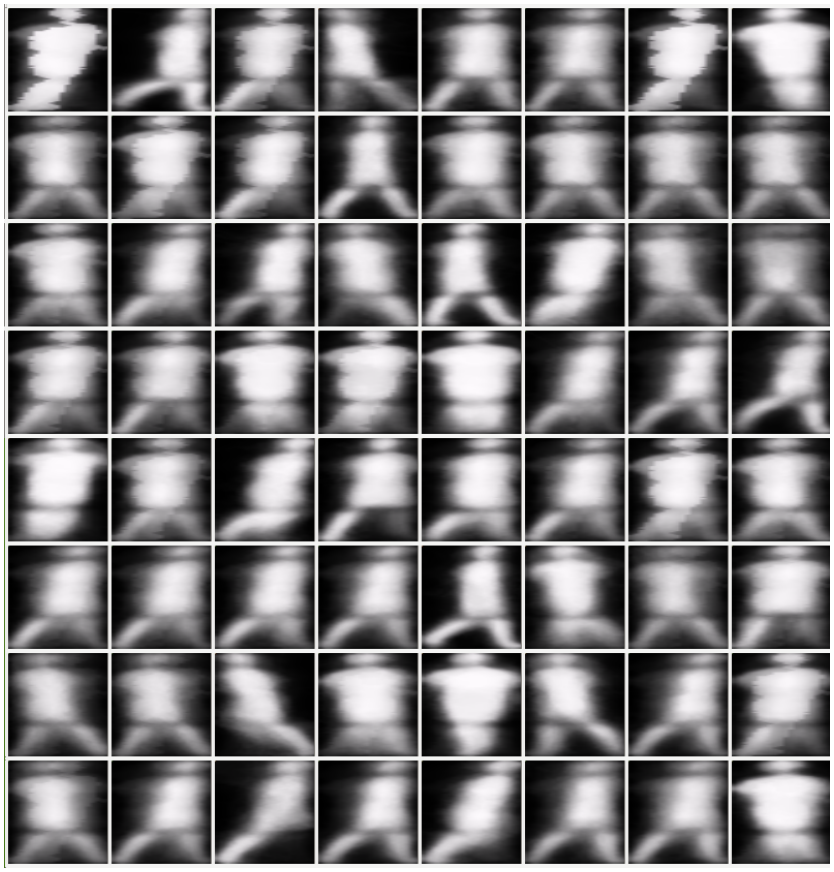


Figure 6.4: Shape model of a soccer player learned by LateGNG.

region. Three different models of rising complexity have been examined: color, texture and gait. Color is represented by adaptive, quantized histograms; it shows the best identification performance. Texture is captured by figure-centric image templates that are bounded to a fixed number by applying LateGNG online. Gait is modeled as a Markov process of shapes, but revealed itself as inappropriate for identification purposes due to a shortage of sufficient training material. Identification by color and texture has been evaluated on players captured from distant and broadcasted views. Training and also test images were automatically gathered during the cognitive tracking process, and thus provide an unsupervised labeled data set. Training as well as identification can be achieved by both approaches in real-time. Although their identification performance is far better than a random approach, distinguishing between players of the same team cannot be accomplished with high accuracy. Identification based on appearance will, however, support tracking based on motion.

Chapter 7

Experimental Results

The total system is evaluated using three common scenarios for computer aided sports video analysis according to the kind of cameras used for capturing a soccer game. First, we describe the evaluation metric that is used throughout the chapter and detail each experiment quantitatively.

7.1 Evaluation Metric

There are many metrics available to evaluate multi-target tracking systems. Bernardin and Stiefelhagen propose the general CLEAR MOT Metrics [28] and give a good overview of recent tracking performance metrics. Needham and Boyle introduced metrics and statistics to compare two trajectories spatially as well as temporally [180]. Li, Dore and Orwell suggest methods to evaluate identity and category tracking of players and the ball in soccer [156].

All proposed metrics have in common that they are designed to evaluate fully automated tracking without identity management (during short sequences) only. At the current stage of research, however, a semi-automated process is much more likely. Identities are mostly not taken into account or they are assumed to be not given either in ground-truth or in the output of the tracking method.

We apply the evaluation method for identity tracking that has already been used in [135] and in chapter 4: The output of the tracking system in real-world coordinates is continuously compared to ground-truth data in terms of the Euclidean distance between targets with the same label. The average of the distances correlates reciprocally with the precision of the system under investigation. When a single distance exceeds a predefined threshold θ, tracking of the corresponding player is marked as a failure, the failed player is reset to the ground-truth position and tracking continues. The number of failures corresponds with the accuracy of the system under investigation, since mix-ups are

implicitly taken into account (although one mix-up is counted twice: once for each target).

The proposed performance metric is easy to implement, scales well with the length of the tracking sequence and provides an intuitive measure. The failures also indicate how often a human operator would have to take corrective action during a semi-automated tracking process to ensure the given upper bound θ of deviation from the correct position. In addition, immediate correction by resetting the target is necessary for the adaptive tracking process. Otherwise, models would have been trained based on wrong data.

There is no real ground-truth available for most of the recorded or broadcasted soccer games, because players can be tracked only visually and camera parameters are only known vaguely. To provide a definite evaluation of accuracy, a second reference system based on laser range finders or other active sensors would be needed. We provided ground-truth by manually marking the position of the players in the videos with a camera calibration that was continuously estimated as described in section 3.2.

The camera calibration adds a systematic error to any vision-based tracking. If multiple cameras are used, the coherence of their calibration directly influences tracking performance because evidence is fused in real world coordinates. In addition, the assumption that the playing field can be described by a plane is not valid in reality due to the construction of sports fields: "Most fields are crowned down the longitudinal axis of the field to promote positive surface drainage, although flat fields sloping towards one touchline or sideline are not uncommon. [...] Probably the best overall design is a side-to-side sloped field with 1% slope and an installed drain system." (see [194], p. 279). The resulting height difference of about 30 cm at the sidelines can cause even higher positional error due to inverse projection computations. The same argumentation holds for position errors induced by inaccurate segmentation and low resolution of the videos. For example, a single pixel in the video corresponds to more than 10 cm height difference at the opposing sidelines of a typical soccer field of size 105 m $\times$ 68 m if captured side by side by two cameras in high definition $1440px \times 1080px$. The best achievable accuracy, assuming perfect calibration and segmentation, would be about 1 m in the direction of the center line due to the projection of the height difference.

7.2 Multiple Static Cameras

Static cameras have the advantage that an initial calibration is sufficient for the complete game without the need for further camera estimation. This setup should be preferred over dynamic cameras if there is a choice, because it saves

much computational time and also makes monitoring easier. We evaluate our tracking system on a publicly available data set to ensure comparability of the performance. Li and Orwell [156] offer a Service to Evaluate the Performance of Tracking and Recognition of Events (SCEPTRE) at `http://sceptre.king.ac.uk/sceptre/default.html`. The English premier league soccer match Fulham vs. Manchester United was recorded on December 30, 2001 by eight static cameras in parallel, covering the whole pitch. The different views are depicted in figure 7.1. The authors provided videos and calibration data for each camera. A XML format to upload tracking results for evaluation is also provided. Unfortunately, the site is not maintained anymore, so the tracking results could not be validated against their ground-truth (which is not downloadable).

Nevertheless, the evaluations of some tracking approaches have been published for this benchmark, so far. Their results are discussed in the following paragraphs.

Li et al. [156] applied the tracking system of [270] and provided a spatio-temporal evaluation of identity tracking as a curve showing the proportion $PI(\Delta d, \Delta t)$ of correctly tracked players over 2000 frames. This metric provides information about the time during which targets could be tracked continuously with a maximal deviation of $\Delta d = 5$ meters from ground-truth.

Song et al. [226] provide qualitative and quantitative results for tracking in a single view (camera 2) through a part of the video. They describe the dataset as "very challenging, where complex interactions frequently occurred. [... Their] ground truth was obtained by software ViPERGT, and the failed tracking were including target missed, false location and identity switch." The conditions for these different types of failures have not been detailed, which makes a direct comparison impossible. In their paper, two graphs and two tables depict the number of failed tracking and a success rate for various tracking methods. The first compares variable multiple target tracking methods on 2000 frames by tracking rate (BPF [182] 68.66%, MCMC-PF[135] 75.63% and theirs [226] 83.75%). The second validates the tracking of seven players during 1200 frames (NNSF 62.17%, JPDAF[14] 73.13%, MC-JPDAF[217] 82.75% and their approach [226] 86.20%). Despite the fact that these success rates look nice, more detailed information that could reveal their true meaning is missing. The authors published another paper [225] at about the same time, where "the detected results of sports video were obtained by Adaboost detection". Four methods were evaluated on a continuous part of the video (again recorded by camera 2), consisting of 300 frames and showing 12 targets at maximum. A graph depicts the number of correct tracking over time. No true identity tracking was evaluated in either publication, because the number of failures in the graphs always drop to zero again after a while, which is highly unlikely if real

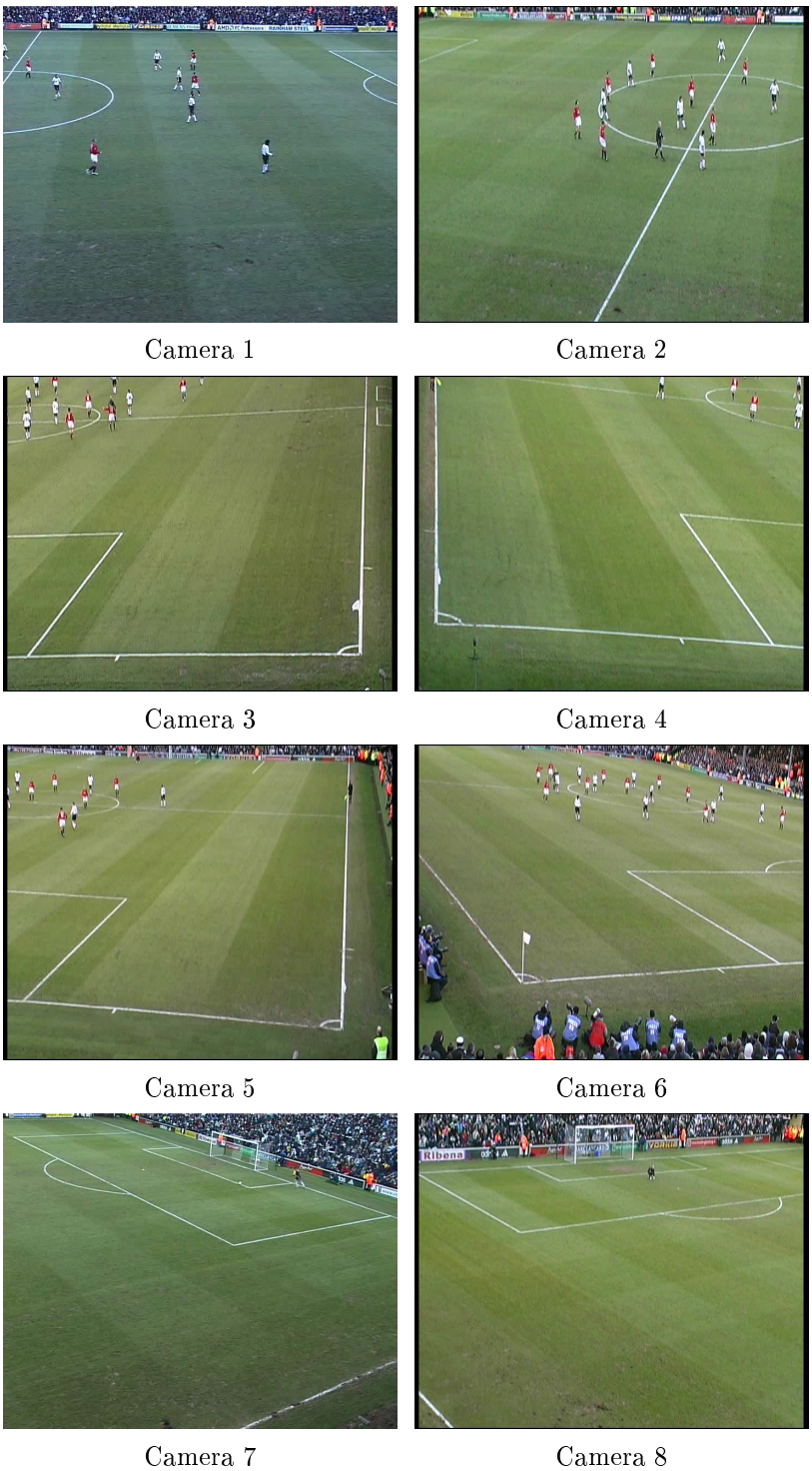


Figure 7.1: The publicly available SCEPTRE dataset provides 3 minutes of a soccer match captured by eight static cameras in parallel.

identities had been tracked.

Gengembre and Pérez [90] provide a qualitative analysis of their tracking results by showing still frames of a selected scene captured by camera 5 to illustrate correct tracking of two players of the same team through an occlusion.

Khan and Shah [131] tracked the players by fusing all frame-synchronized views. They remark that "occlusions are quite abundant due to the large number of players. There is also a lot of clutter due to jitter of the cameras. We believe this is a result of wind or the shaking of the platform on which the cameras were mounted. Another challenge is the lack of pixel resolution on players. Depending on the view, player patches could be as small as 5 x 25 pixels." Three frames were shown in the paper to support qualitative evaluation of the tracking performance.

We tracked all players and the referee through the full length of the given videos (4024 frames or 161 seconds of the match respectively). A maximum of $N_{max} = 50$ particles was used to track the 23 targets with real world coordinates in meters. The constant velocity model was parameterized as follows: $\Delta t = 0.04$ (due to 25fps) and $\tilde{q} = 0.0008$ (due to the maximum acceleration of humans). The multiplicity of measurements is constrained by the model of equation 4.38 with $p_d = 0.5$ and $p_{sd} = 0.2$. Covariances were initialized with the identity matrix $V_0 = I_{4N}$. The probability for clutter was set to $p\left(J_k(l) = 0|\hat{x}_k, z_k\right) = 0.001$.

A total of 538 failures was detected during tracking with identification by color in 4024 frames, resulting in a tracking rate of 86.63%. Figure 7.2 depicts the cumulative failures for each player in detail. Results for tracking with identification by texture are 535 failures (correct tracking rate 86.71%). Details per player differ only marginally from the ones shown in figure 7.2, so we omit depicting them here. It is noteworthy that one goalkeeper was tracked without failures while the other one caused the most failures in tracking. The failing goalie wears a black jersey and is mainly captured by camera 8. Failures are due to mixing with the perimeter advertising boards and occlusions. The rest of the players are tracked with a significantly lower failure rate.

Preprocessing of all eight cameras and the tracking were executed on a single Quadcore computer. A single measurement sweep of one camera needed 11.80 ms on an average (with a standard deviation of 11.25) for identification by color and 11.73 ms per frame (with a standard deviation of 11.02), when texture was used as feature. This means that up to three cameras could be processed on a single machine in real-time. When preprocessing was run distributed, the tracking achieved real-time (25fps).

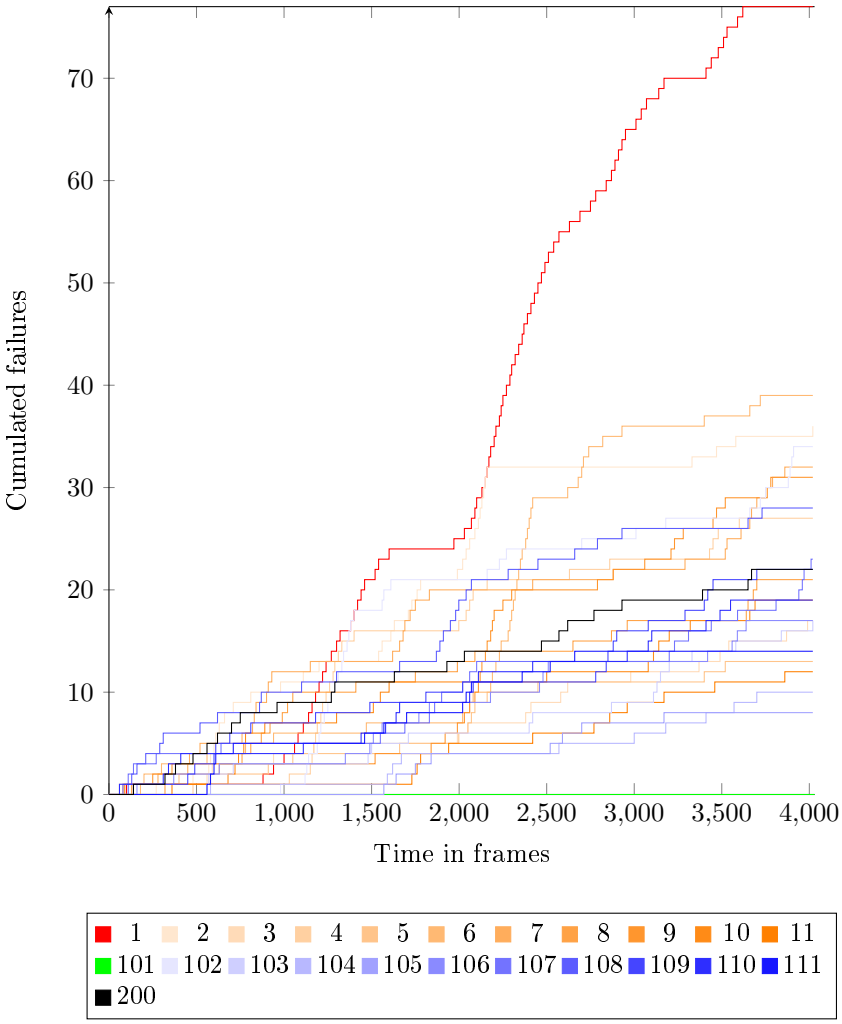


Figure 7.2: The graphs depict cumulative failures for each player during the 4024 frames of the SCEPTRE dataset. Failures for the goalkeeper shown in red are mainly due to similarity with the advertising boards.

7.3 Single Dynamic Camera

Digital videos captured by a single panning, tilting and zooming camera, which is located at one corner of the stadium roof about 13 m above the playing field, provide the basic raw material for the second experiment. The videos are encoded in Digital Video (DV) format with a frame rate of 25Hz in interlaced resolution of 720×576 pixels.

We tracked all players and the referee through a complete halftime of 48 minutes. To the best of our knowledge, no quantitative evaluation of sports video

analysis of comparable length has been published so far. The videos show the final game of the FIFA world championship 2006 with France vs. Italy. Examples of typical views and some challenging scenes are depicted in figure 7.3. Although the changes in zoom are usually moderate, some extreme close-ups are also included in the material. Players are 10×30 in size but may be less when they are closer to the French goal.

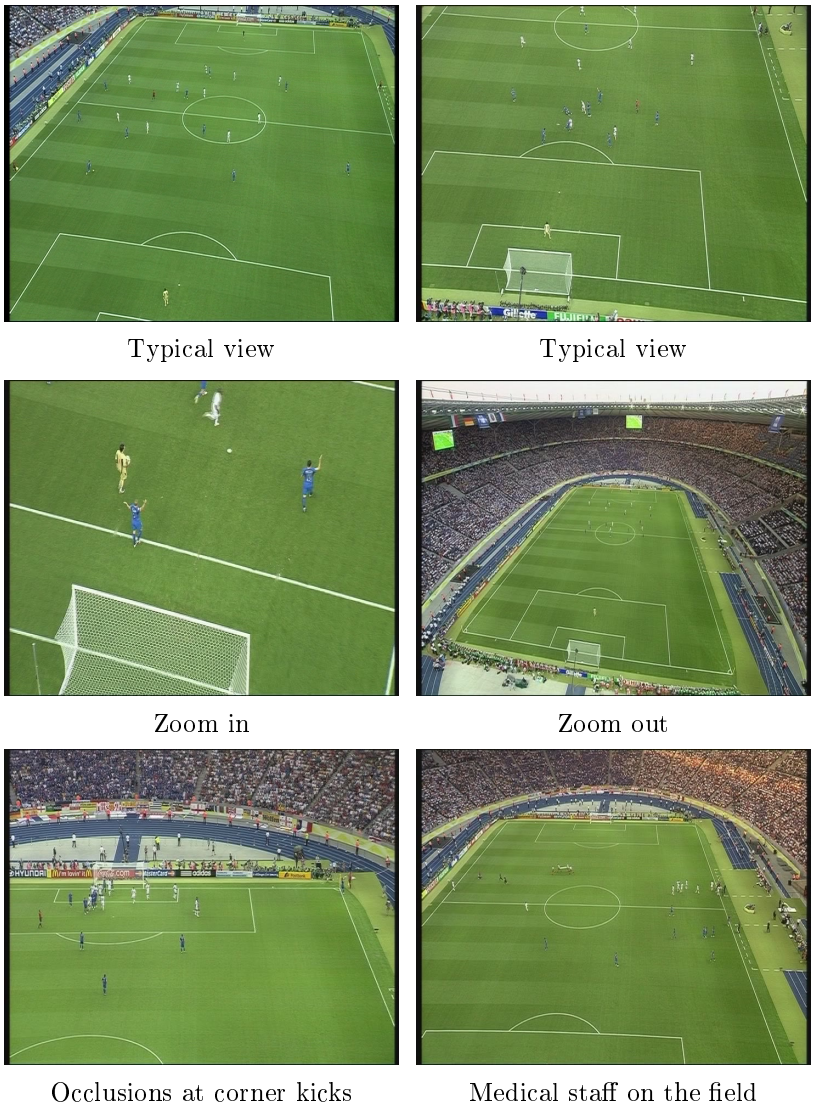


Figure 7.3: The first halftime of the 2006 world championship final provides 71778 frames recorded with 25Hz by a single panning, tilting and zooming camera.

The number of particles was set to $N_{max} = 50$ to track the 23 targets. We used the following parameters for the constant velocity model: $\Delta t = 0.04$ (due to 25fps) and $\tilde{q} = 0.0008$ (due to max acceleration of humans). The multiplicity of associations was constrained by the model of equation 4.38 with $p_d = 0.9$ and $p_{sd} = 1.0$ allowing only feasible associations. Covariances were initialized with $V_0 = 0.001 I_{4N}$. The probability for clutter was set to $p\left(J_k(l) = 0|\hat{x}_k, z_k\right) = \frac{1}{720 \cdot 576} \approx 0.000002$. Failures were encountered when a target deviated from the ground-truth position by more than $\theta = 5.0$ meters. After a failure, the failed player was reinitialized to the ground-truth position and tracking was resumed.

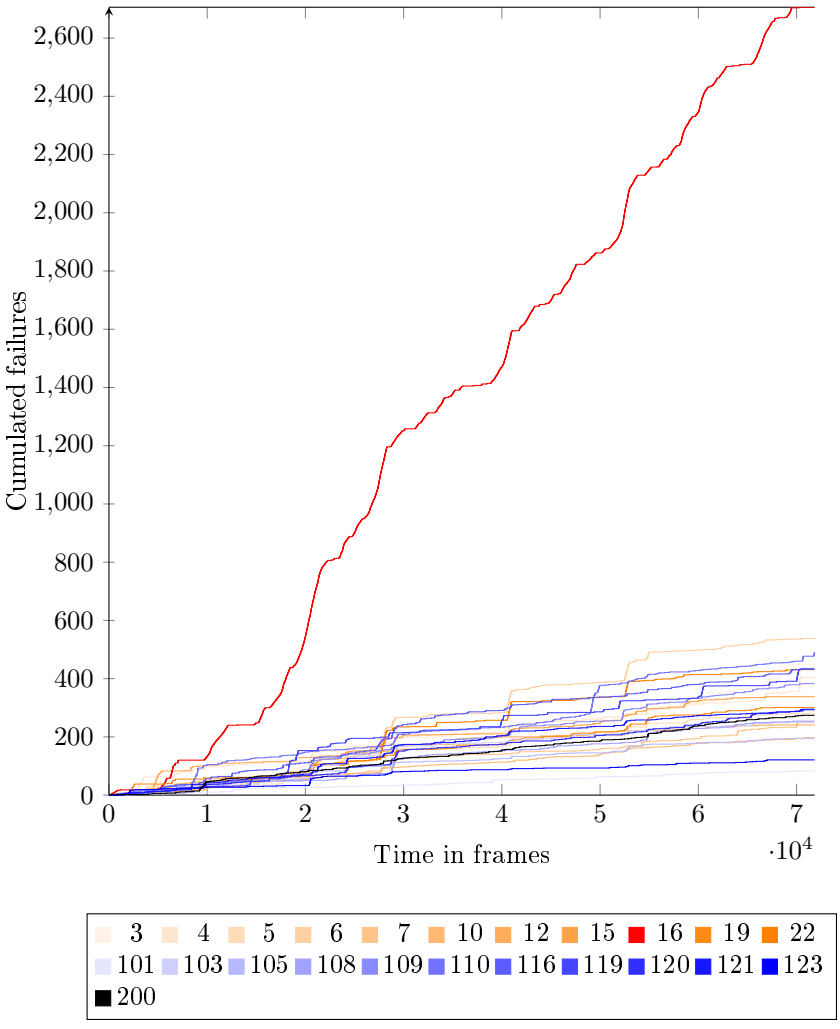


Figure 7.4: The graphs depict cumulative failures for each player during tracking with identification by color of the 71778 frames of the world championship final 2006. The French team is colored orange and the Italian team is blue. The French goalkeeper with number 16 is often too small to be detected.

A total number of 9646 failures was counted during tracking of 71778 frames, if identification by color histogram was applied, resulting in correct tracking of all 22 players and the referee in 86.57% of the time. Figure 7.4 visualizes the failures for each player. Each frame needed 22.64 ms (with a standard deviation of 11.14) to be processed on a single machine with an Intel Quad CPU at 3.0GHz.

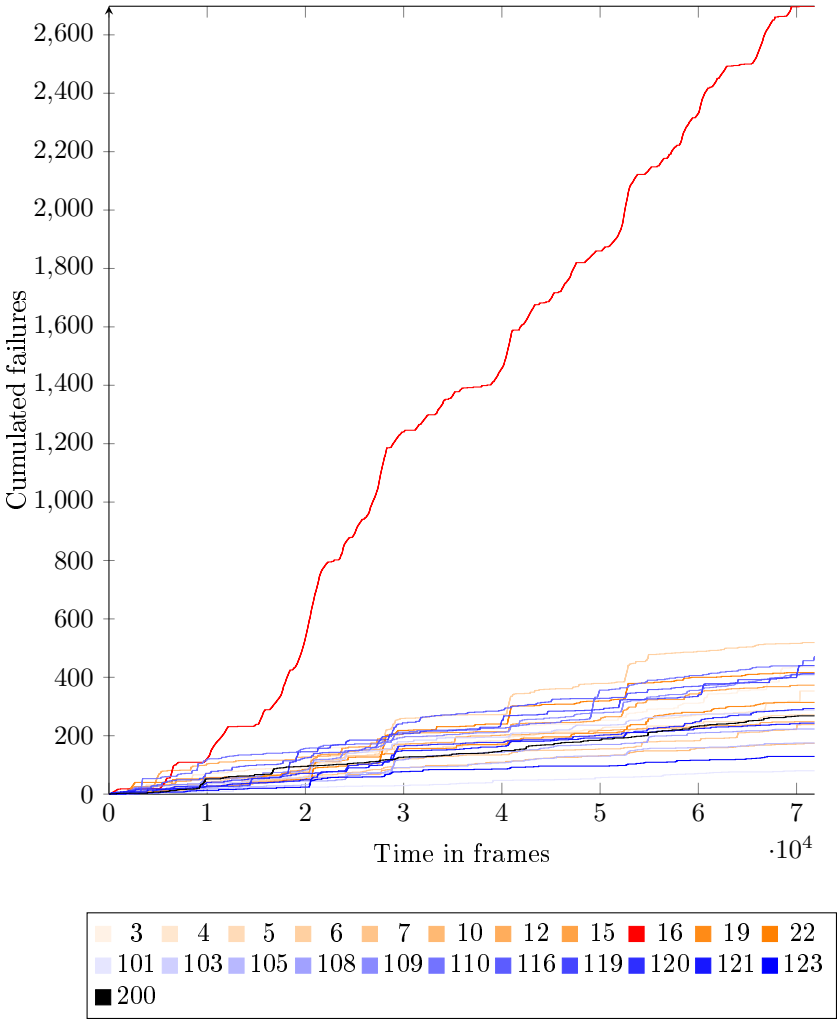


Figure 7.5: The graphs depict cumulative failures for each player during tracking with identification by texture of the first halftime of the world championship final 2006. The French team is colored orange and the Italian team is blue. The French goalkeeper with number 16 is often too small to be detected.

Tracking with identification by texture failed 9474 times during the halftime of 71778 frames, which means a correct tracking rate of 23 targets in 86.80% of the time. Figure 7.5 visualizes the failures for each player. A single frame needed 22.65 ms in average (with a standard deviation of 11.07) to be processed

on the same 3.0 GHz Quadcore computer as used for evaluation of identification by color. The time translates to a frame rate of 44 fps which demonstrates again the real-time capability of the proposed system.

For both tracking experiments, the goalkeeper of the French team (with jersey number 16) was much harder to track than the other players, as can be seen in figures 7.4 and 7.5. This is due to the small size of this athlete in the image, as he constitutes the athlete who is farthest from the camera. Therefore, he is often not detected as a valid player region but discarded as noise, causing tracking to fail. A change in the position or the focal length of the camera, however, would solve this problem. Alternatively, the negative information-handling could be turned off especially for this player, because the goalkeeper does not move that far from the goal anyway. Omitting this goalkeeper, the correct tracking rate would increase to more than 90%.

7.4 Broadcasted Material

The third dataset consists of 1000 continuous frames of broadcasted material without cuts. Some still frames are depicted in figure 7.6. The camera is panning, tilting and zooming during the scene. The video provides a typical view for broadcasts from a camera located at the height of the center line about 19 m above the playing field and 36 m away from the sideline. The video shows about two minutes (minute 24:58 until 25:38) of the UEFA Champions League soccer match Sporting Lissabon versus FC Bayern München on February 22, 2009. We recorded the game from a normal broadcast with a DVB-T converter card.

Similar parameters as in the other tracking experiments were used. The multiplicity of associations was constrained by the model of equation 4.38 with $p_d = 0.6$ and $p_{sd} = 1.0$ allowing only feasible associations. Covariances were initialized with $V_0 = I_{4N}$. The probability for clutter was set to $p\left(J_k(l) = 0|\hat{x}_k, z_k\right) = 0.001$ due to more clutter induced by motion blur and compression artefacts.

We encountered 165 failures with identification by color (corresponds to a tracking rate of 83.5%) demanding 10.91 ms on an average for a single frame (with standard deviation of 13.14). If the players are identified by texture, a correct tracking rate of 87.0% (130 failures) could be achieved. Tracking needed 14.26 ms on an average (with standard deviation of 17.95) including the training and classification by the texture models. Times were measured on a 2.2 GHz DualCore AMD processor. The difference between the performances can be explained by the ability of the texture models for faster learning, which makes the difference for this short sequence, where players are visible during parts of the video only. The failures are depicted individually for each player in figure 7.7

Figure 7.6: A broadcasted UEFA Champions League soccer match provides an uncut scene of 1000 frames for evaluation.

and 7.8. One can see that a mix-up of player 11 and 106 could be resolved by the texture models in contrast to the color models.

The main problem of tracking broadcasted video is the lack of complete visibility of the match. Cuts and zoom-ins occur frequently, stating the need to estimate which camera is currently recording the visible scene. It is almost never the case that all players are shown in the image. Overlays shorten the visible area even further. Data quality is also an issue because the DVB-T stream we recorded is compressed and shows typical MPEG block artifacts. Tracking rates could be improved if a second camera stream from a further-off perspective would be added. This would also support human interaction, since the identities of all players are hard to monitor in the broadcasted footage.

7.5 Towards Normal Operating Conditions

The proposed tracking system is currently in use by sports scientists of Prof. Martin Lames's group. Several soccer matches have already been tracked to provide

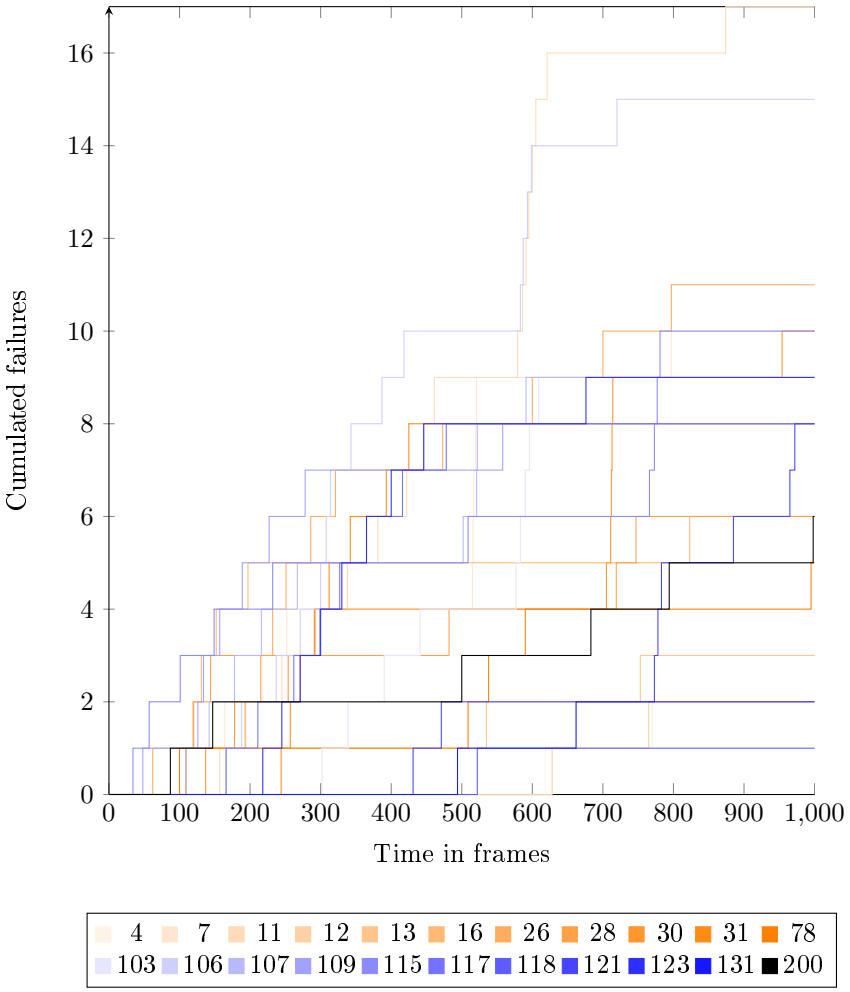


Figure 7.7: The graphs depict cumulative failures for each player during the tracking with identification by color of 1000 frames of broadcasted material.

analyses of the first and second German Bundesliga for the FC Bayern München and the FC Augsburg as well as statistics of national competitions for the female German soccer team. Usually, three stationary but portable static cameras were used to record the games in high definition. One or two operators monitored and corrected the tracking process. We found that the human capacity to monitor a single team with eleven or 23 players limits the processing speed – rather than the computational demand.

In addition to a distributed graphical user interface for monitoring the tracking process, tools for post-processing trajectories and for visualizing analyses by chroma keying were developed. The user can navigate through a virtual view of the game and watch the play from an arbitrary perspective. Augmented videos

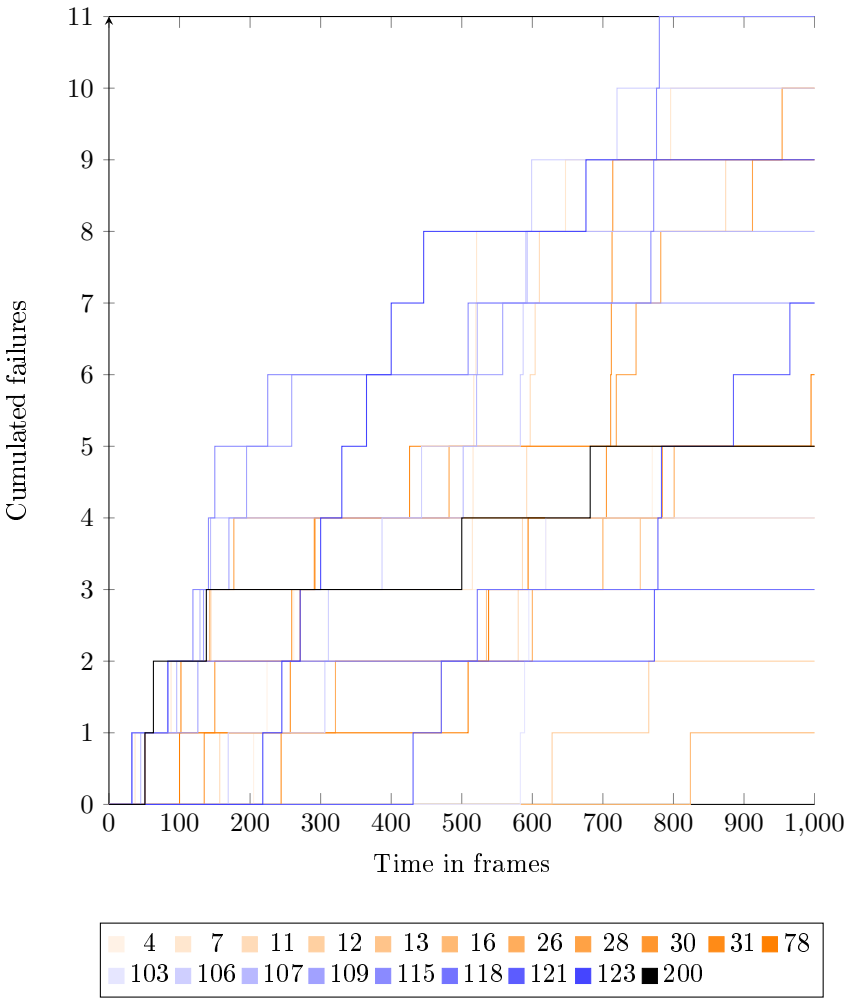
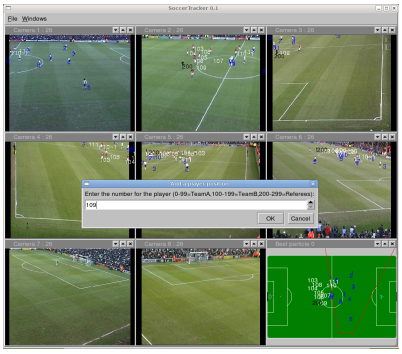


Figure 7.8: The graphs depict cumulative failures for each player during the tracking with identification by texture of 1000 frames of broadcasted material.

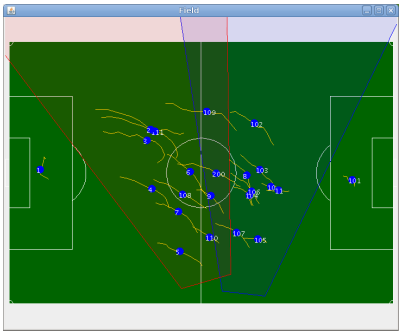
can be generated in batch mode via movie scripts written in XML format, which also enables automated video generation as an answer to user queries. Some of these augmented videos were included in the video installation "Deep Play" by Harun Farocki and were presented at the international art exhibition documenta 12. Figure 7.9 illustrates some of these visualizations.

7.6 Conclusions

In this chapter, the real-time tracking system for soccer videos has been evaluated using three different application scenarios for sports video analysis in

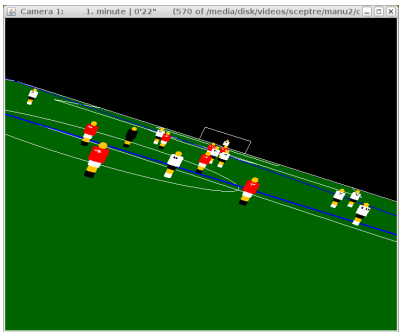


(a) Monitoring

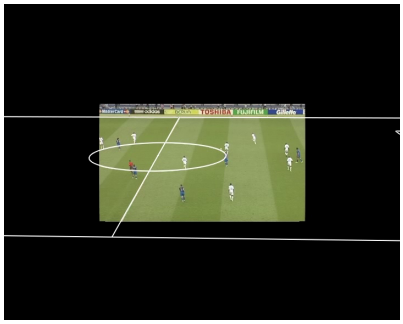


(b) Postprocessing

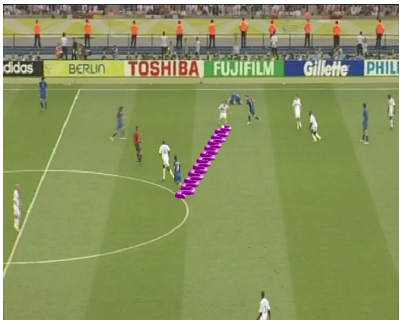
(c) Augmented reality



(d) Virtual view



(e) One of four contributions to Deep Play by Harun Farocki at documenta 12



(f) Another contribution to Deep Play

Figure 7.9: Tools for monitoring and visualization.

terms of the camera set-up used. We introduced an intuitive performance measure for identity tracking which encountered failures by a thresholded deviation from ground-truth. The proposed system exhibited robust tracking of all players and the referee with a failure rate of around 86% on an average in these domains. Tracking with identification by texture showed – contrary to the results of chapter 6 – better performance than tracking with identification based

on color histograms. It is especially suited for short video clips. The performance on a publicly available dataset demonstrated the capability of fusing multiple cameras. Accurate camera calibration was revealed as important because errors may be magnified due to projective geometry. Further, we provided quantitative results on a full-length soccer match, tracking all 22 players and the referee including challenging scenes through 48 minutes of play. To the best of our knowledge, nobody has published quantitative evaluation of tracking systems for sports video analysis on footage of comparable length so far. Broadcasted material is the most challenging due to frequent cuts and necessary re-initialization, because only a subset of the players is shown most of the time. Our tracking system could also handle this scenario well. To reduce manual correction and re-initialization, the use of additional cameras is recommended in practice. The experiments demonstrate that the proposed system enables semi-automated tracking of all players and the main referee during complete soccer games in real-time.

Chapter 8

Tactical Sports Video Analysis

Until now we have described how to gather trajectories of all players from sports videos in real-time. The next question is what can be done with these data. Position data of all players provide a rich base for further analysis. In this chapter, we review the work on automated tactical sports video analysis. We introduce a method for a specific analysis of team behavior and give an outlook on further analysis inside an action model logics framework. The considered computational problem of tactical sports video analysis is illustrated in figure 8.1.

8.1 Related Work

A wide range of literature for soccer tactics from the coaching perspective exists, f.i. [168]. The scientific analysis of soccer and soccer players based on physics and statistical data is described in [261].

Research in automated tactical analysis based on trajectories can be categorized as either situational or team behavior analysis. The former analyzes situations as static snapshots of the game and the latter focuses on the temporal aspect of behavior. We subdivide the research of team behavior further as to its use of either unsupervised or supervised learning methods, which allow the detection of unknown or predefined patterns in the game.

8.1.1 Situational analysis

Kawashima et al. [130] propose a qualitative group analysis. Medium and large player groups are extracted by multi-scale analysis in image sequences. They state in [130]: "The exact information, such as the velocity, direction, or its

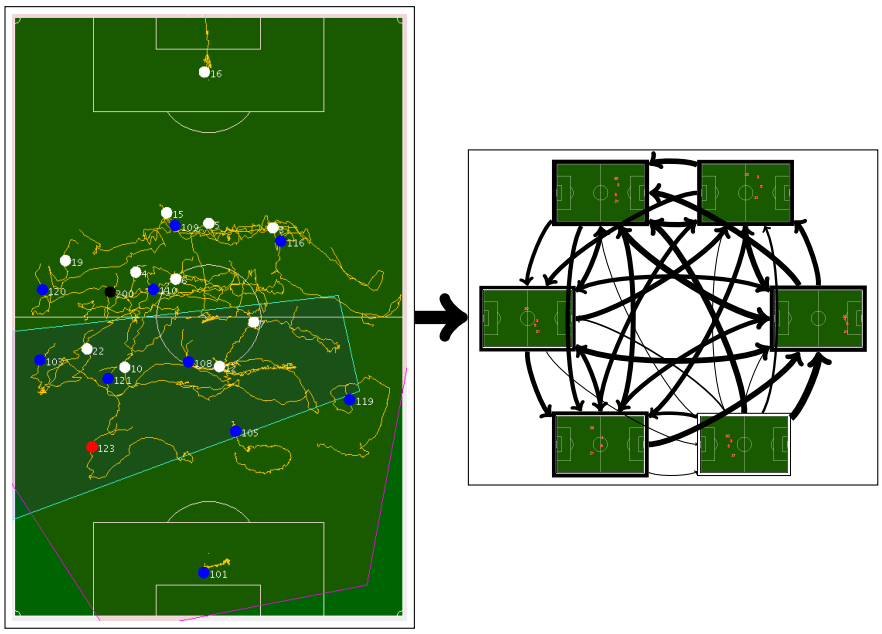


Figure 8.1: Tactical sports video analysis provides insights into the game and supports humans to extract knowledge from the players' trajectories.

distribution is not used in the interpretation. These parameters are, of course, useful in accurate analysis." Taki and Hasegawa [235, 234] suggest the use of minimum moving time patterns and dominant regions of a player for analysis of soccer game scenes. Therefore, the playing field is partitioned into Voronoi regions. These regions are defined by the contour consisting of positions that can be reached in the same minimal time from different players. Kang et al. [125] quantitatively evaluate the performance of soccer players based on the average size of catchable and safe areas and according to a safe pass ratio. Catchable and safe regions are defined similarly to the minimum moving time and the dominant regions. The performance measure was evaluated on data obtained from a soccer simulation game.

A quantitative analysis of trajectories of Brazilian soccer players was done in [19]. Pingali et al. [193] describe coverage maps as a visualization of the positions of a (tennis) player for tactical analysis based on real data. The time a player stays inside a bin of the quantized field is integrated and finally visualized by color mapping similar to infrared images. Little and Gu [160] split motion trajectories into paths and speed curves by extracting their spatial or temporal component only. A query-by-example interface is supplied for fast retrieval of similar motions based on the proposed representations. Wünstel et al. [263] train self-organizing maps (SOM) with trajectories that are represented

in a spatially focused manner (SFR) for clustering. As already mentioned in section 5.1, Visser et al. [250] as well as Ramos and Ayanegui [199] classify formations in simulated RoboCup games.

8.1.2 Unsupervised team behavior analysis

In the RoboCup community, the topic of team behavior analysis attracts high attention because simulated trajectory data are available in huge amounts. Visser and his group investigated online learning of team behavior [249] using Time Series-Based Decision Tree Induction and extending it for real-time analysis and prediction [175] based on qualitative description of motion scenes.

Kubo and Kakazu [144] propose the artificial soccer agent MATEUS to generate goal-directed behavior based on Stochastic Learning Automata. They evaluated the simulation traces of MATEUS and concluded that "[i]n [their] model, the expected coordinated motion and strategy emerge but they are too much sensitive with the pre-fixed or given system parameters so [they] should tackle to this proposed formulae more mathematically."

The Advanced Scout system by Bhandari et al. [30] applies data-mining techniques to NBA basketball games to help coaches find interesting patterns in their players' and opponents' behaviors. Attribute Focusing (AF) compares the overall distribution of an attribute with the distribution of this attribute for various subsets of the data. If the distribution of a certain subset of data significantly differs from the overall one, the constraints on the attributes that define this subset are marked as interesting statistical anomalies.

Nair et al. [177] proposed an automated team analyst called ISAAC for post-hoc, online agent-team analysis. Decision trees are learned from the external behavior traces of the teams. These models can be used for detecting possible agent improvements and allow comparison of different teams based on the human readable form of the learned decision trees.

In [105], Hirano and Tsumotoa studied the discovery of meaningful pass patterns and sequences from ball trajectories of soccer games. Multiscale Matching was used as a distance measure between two planar curves, partly changing observation scales. Given this metric, the trajectories are hierarchically clustered to identify similarities.

8.1.3 Classification of team behavior

For the purpose of video indexing and retrieval, predefined team behavior must be recognized in the trajectories. Intille [114, 115, 112] proposes a framework for the recognition of highly structured, multi-person action from noisy trajectories. Visually grounded goal-based primitives and low-order temporal relationships

are integrated in a probabilistic manner by automated generation of Bayesian networks that classify single-agent and multi-agent actions. A system was evaluated for American football.

Li and Woodham [154] developed a proof-of-concept system to represent and reason about hockey play. Finite State Machines (FSM) model domain knowledge of actions, where each transition is associated with an Event Evaluation Function (EEF) that assigns the immediate outcome as a reward. The system emits a description of the game and identifies key plays.

Complex multi-player behavior has been recognized in basketball games by Perse et al. [189]. A probabilistic play model is applied to player-trajectories to segment the game into phases (offense, defense, time out). Trajectories of a single phase are classified by nearest neighbor classification to activity templates that were predefined by experts.

Feldman [75] recognized the behavior of protagonists (ants and bees), analyzing the trajectories with Hidden Markov Models (HMMs). HMMs [197] constitute the status quo for modeling sequences as transitions between discrete states. They are also used to recognize actions in videos or spoken language.

8.2 Merge Growing Neural Gas for Team Behavior Analysis

Team behavior denotes the temporal organization of actions of individual agents towards a common goal. This is reflected in the trajectories of all players by temporal patterns in the formations of each team. In order to extract these patterns automatically, we extend the LateGNG algorithm (c.f. section 5.3.2) to cluster time-series.

8.2.1 Related work

Several vector quantization methods based on Hebbian learning have been extended to handle time series. Common approaches use hierarchies [38], non-Euclidean sequence metrics [101], time-window techniques [172] and mapping to spatial correlations [67] and a wider field of recursive sequence models also exists. These recursive models feed the past input to the learning method, albeit with a delay (see [99] for a survey and [100] for a unifying notation). Temporal Kohonen Map (TKM) [43], Recurrent SOM (RSOM) [141], Recursive SOM (RecSOM) [251], SOM for structured data (SOM-SD) [97], Merge SOM (MSOM) and Merge Neural Gas (MNG) [232] represent popular recursive models which have been applied in several applications [147, 70, 71, 65]. The specific models differ primarily in their internal representation of time series,

which influences the capacity of the model, the flexibility with respect to network topology and the processing speed. MNG has shown superior performance to the other recursive models for acceptable time complexity.

8.2.2 Merge Growing Neural Gas

We propose Merge Growing Neural Gas (MGNG), which advances the vector quantization algorithm LateGNG towards clustering of time-series. Each neuron is extended to hold an additional context vector $\mathbf{c}_n$ representing all past time steps of a sequence in addition to the prototype vector of the static case. This temporal context is also used by MNG [232]. By incorporating the past into the clustered data, the snapshots of the incoming sequence are quantized in conditional dependency to the preceding sequence. Because the input data (snapshots of the sequence) are noisy, the corresponding cluster prototypes are incorporated in the temporal context instead of the real data.

An input sequence $\mathbf{x}_1, \ldots, \mathbf{x}_k$ is assigned to the best matching neuron by finding the neuron n with lowest distance d_n in time step k according to

$$d_n(t) = (1-\alpha) \cdot \|\mathbf{x}_k - \mathbf{w}_n\|^2 + \alpha_t \cdot \|\mathbf{C}_k - \mathbf{c}_n\|^2 . \tag{8.1}$$

The parameter $\alpha_t \in [0,1]$ weights the importance of the recent input signal over the past. $\mathbf{C}_k$ is called the global temporal context and is computed as a linear combination (merge) of the weight and the context vector from the winner r of time step $k-1$

$$\mathbf{C}_k := (1-\beta) \cdot \mathbf{w}_r + \beta_t \cdot \mathbf{c}_r. \tag{8.2}$$

The parameter $\beta_t \in [0,1]$ controls the influence of the more distant over the recent past. The global temporal context is initialized as $\mathbf{C}_1 := 0$.

When the network is trained, $\mathbf{C}_k$ converges to the optimal global temporal context vector $\mathbf{C}_k^{opt}$, which can be written as (c.f. [232]):

$$\mathbf{C}_k^{opt} := \sum_{j=1}^{k-1} (1-\beta_t) \cdot \beta_t^{t-1-j} \cdot \mathbf{x}_j. \tag{8.3}$$

The temporal context therefore represents the entire sequence encoded as an exponentially decreasing series (also called fractal encoding of sequences [232]).

We apply an entropy maximization strategy for node insertion instead of the error minimization approach typically used in GNG and LateGNG. This strategy is justified due to the fact that we are more interested in a representation of frequent sequence patterns than in the detailed reconstruction of the data at discrete time-points. The entropy of a network is highest if the activation of all neurons is balanced. At high entropy, more neurons are used for frequent sequences, reducing the representation capacity for rare ones. This helps to

MGNG-Learn($\mathbf{x}, \epsilon_b, \epsilon_n, \gamma, \eta, \lambda, \theta, \lambda_a, \alpha_t, \beta_t$):

1. IF $|\mathcal{K}| < 2$
2. $\mathcal{K} = \mathcal{K} \cup \{n\}$ with $\mathbf{w}_n = \mathbf{x}$
3. $t = |\mathcal{K}|, t_a = 1, \mathbf{x}_{acc} = \mathbf{x}, s_{last} = r_{last}, r_{last} = n$
4. initialize global temporal context $\mathbf{C}_1 := \mathbf{0}$
5. ELSE
6. $d_n(t) = (1 - \alpha_t) \cdot \|\mathbf{x}_t - \mathbf{w}_n\|^2 + \alpha_t \cdot \|\mathbf{C}_t - \mathbf{c}_n\|^2$
7. find winner $r = \arg\min_{n \in \mathcal{K}} d_n(t)$
8. find second winner $s = \arg\min_{n \in \mathcal{K} \setminus \{r\}} d_n(t)$
9. IF $r = r_{last} \wedge s = s_{last} \wedge t_a < \lambda_a$
10. $\mathbf{x}_{acc} = \mathbf{x}_{acc} + \mathbf{x}$
11. $t_a = t_a + 1$
12. $t = t + 1$
13. $\mathbf{C}_t := (1 - \beta_t) \cdot \mathbf{w}_r + \beta_t \cdot \mathbf{C}_t$
14. ELSE
15. IF $t_a = \lambda_a \wedge |\mathcal{K}| < \theta$
16. Add new node l: $\mathcal{K} = \mathcal{K} \cup \{l\}$ with $\mathbf{w}_l = \mathbf{x}_{acc}, \mathbf{c}_l = \mathbf{C}_t$
17. add connections $\mathcal{E} = \mathcal{E} \cup \{(r_{last}, l), (l, s_{last})\}$
18. ELSE
19. $\mathbf{x}_{acc} = t_a^{-1} \mathbf{x}_{acc}$
20. $\epsilon_b' = 1 - (1 - \epsilon_b)^{t_a}$
21. $\epsilon_n' = 1 - (1 - \epsilon_n)^{t_a}$
22. $\epsilon_{c_b}{}' = 1 - (1 - \epsilon_{c_b})^{t_a}$
23. $\epsilon_{c_n}{}' = 1 - (1 - \epsilon_{c_n})^{t_a}$
24. MGNG-Update($\mathbf{x}_{acc}, \mathbf{C}_t, t, r, s, \epsilon_b', \epsilon_n', \epsilon_{c_b}{}', \epsilon_{c_n}{}', \gamma, \eta, \lambda, \theta$)
25. $t_a = 1, r_{last} = r, s_{last} = s, \mathbf{x}_{acc} = \mathbf{x}$
26. $t = t + 1$
27. $\mathbf{C}_t := (1 - \beta_t) \cdot \mathbf{w}_r + \beta_t \cdot \mathbf{c}_r$

Figure 8.2: The training algorithm of Merge Growing Neural Gas (MGNG) for time series analysis.

MGNG-Update($\mathbf{x}, \mathbf{c}, t, r, s, \epsilon_b, \epsilon_n, \epsilon_{c_b}, \epsilon_{c_n}, \gamma, \eta, \lambda, \theta$):

1. increment counter of r: $\iota_r = \iota_r + 1$
2. connect r with s: $\mathcal{E} = \mathcal{E} \cup \{(r, s)\}$
3. $age_{(r,s)} = 0$
4. increment the age of all edges connected with r
   $age_{(r,n)} = age_{(r,n)} + 1 \quad (\forall n \in \mathcal{N}_r \setminus \{s\})$
5. remove old connections $\mathcal{E} = \mathcal{E} \setminus \{(a, b) | age_{(a,b)} > \gamma\}$
6. delete all nodes with no connections
   $\mathcal{K} = \mathcal{K} \setminus \{n | \forall k \in \mathcal{K}.(n, k) \notin \mathcal{E} \wedge (k, n) \notin \mathcal{E}\}$
7. update r and its direct topological neighbors $\mathcal{N}_r$:
   $\mathbf{w}_r = \mathbf{w}_r + \epsilon_b \cdot (\mathbf{x} - \mathbf{w}_r)$, $\mathbf{w}_n = \mathbf{w}_n + \epsilon_n \cdot (\mathbf{x} - \mathbf{w}_i) \quad (\forall n \in \mathcal{N}_r)$
   $\mathbf{c}_r = \mathbf{c}_r + \epsilon_{c_b} \cdot (\mathbf{c} - \mathbf{c}_r)$, $\mathbf{c}_n = \mathbf{c}_n + \epsilon_{c_n} \cdot (\mathbf{c} - \mathbf{c}_i) \quad (\forall n \in \mathcal{N}_r)$
8. IF $t \mod \lambda \equiv 0 \wedge |\mathcal{K}| < \theta$
9. find neuron q with greatest counter $q = \arg\max_{n \in \mathcal{K}} \iota_n$
10. find neighbor f of q with $f = \arg\max_{n \in \mathcal{N}_q} \iota_n$
11. new node l: $\mathcal{K} = \mathcal{K} \cup \{l\}$,
    $\mathbf{w}_l = \frac{1}{2}(\mathbf{w}_q + \mathbf{w}_f)$,
    $\mathbf{c}_l = \frac{1}{2}(\mathbf{c}_q + \mathbf{c}_f)$,
    $\mathtt{l}\iota_l = \delta \cdot (\iota_f + \iota_q)$
12. adapt connections $\mathcal{E} = (\mathcal{E} \setminus \{(q, f)\}) \cup \{(q, n), (n, f)\}$
13. $\iota_q = (1 - \delta) \cdot \iota_q$
14. $\iota_f = (1 - \delta) \cdot \iota_f$
15. decrease all counters $\iota_n = \eta \cdot \iota_n \quad (\forall n \in \mathcal{K})$

Figure 8.3: Update step of the MGNG network maximizing the entropy.

focus on quantization of important information in addition to the usually combinatorial explosion of time series. Following Fritzke's [84] proposed strategy, we insert a new node in regions with high activation frequency, leading to an increase of the entropy of the network. Frequency is tracked by a counter ι of every neuron and this counter is incremented every time the neuron is selected as the winner. New nodes are inserted between the most active neuron q and its most frequent topological neighbor f, reducing the likelihood of both nodes q and f to be selected as the winner and therefore increasing the overall entropy of the network. The new node l is initialized as the mean of the two selected nodes and inserted between them. The counters of q and f are reduced to reflect the

expected decrease of activation, while the new neuron takes over this activation. The parameter δ controls the amount of the activation shift. All counters are subject to exponential decay by the parameter η in order to give recent changes a higher relevance. To further increase the entropy, nodes with no connections are deleted because the last selection as the first or second best matching unit was too long ago.

In order to represent the temporal structure in the series, all transitions between the nodes are counted as well. The sequence of the best matching unit inside the net forms a discrete first-order Markov process. Probabilities for observing a given sequence can be computed recursively by combining the probability for the current state, for the previous transition and for the remaining sequence as described in [197].

The complete training algorithm for MGNG is depicted in figures 8.2 and 8.3.

8.2.3 Evaluation

The binary automaton experiment was proposed by Voegtlin [251] in order to evaluate the representational capacity of temporal models. This experiment uses a Markov automaton with the discrete states 0 and 1 and the probabilities $P(0) = \frac{4}{7}$, $P(1) = \frac{3}{7}$, $P(0|1) = 0.4$, $P(1|1) = 0.6$, $P(1|0) = 0.3$, $P(0|0) = 0.7$. This automaton is depicted in figure 8.4. A sequence with 10^6 elements was generated and trained in a network with 100 neurons. After training, the winning units for the 100 most probable sequences are determined and the longest sequence that can still be differentiated is associated with multiple winners. An optimal result would be achieved if each of the 100 sequences would have an unique winner. The experiment was carried out comparing MNG and MGNG by using the following parameters in compliance with [232]: $\alpha = 0.5$, $\beta = 0.75$, $\theta = 100$, $\lambda = 600$, $\gamma = 88$, $\epsilon_b = 0.05$, $\epsilon_n = 0.0006$, $\delta = 0.5$, $\eta = 0.9995$.

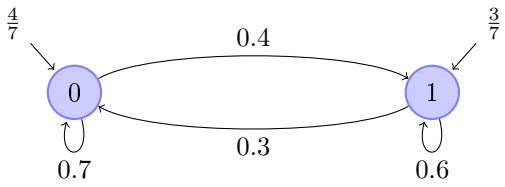


Figure 8.4: The binary automaton experiment proposed by Voegtlin [251] evaluates the representation capacity of temporal models.

Compared to MNG, MGNG shows a slight improvement in the representation capacity as well as a clear advantage in computational time. A total number of 64 longest sequences could be discriminated by MGNG requiring 69 s, while

62 sequences have been discriminated by MNG in 131 s, running non-optimized Java code. Unfortunately, both models cannot distinguish all 100 sequences, because recursive models that represent the temporal context as a weighted sum are not able to discriminate between sequences with repeated '0' signals (such as: 0, 00, 0000...0). Choosing other values like '1' and '2' would improve the results considerably. We have published more experiments comparing MNG and MGNG in [5]. To conclude, one can say that MGNG never exhibited worse accuracy than MNG but reduced the computational time significantly by a factor of two. Contrary to MNG, MGNG is suited for online learning of time series because only constant parameters are used. In addition, the combination of the temporal context and delayed update improves the clustering of continuous data.

8.2.4 Application to team behavior analysis

We model a soccer game as a sequence of coarse formations to provide a summarization. The dynamics can be represented by a probabilistic automaton with states consisting of the formations and edges between them. The edges are labeled with the probability of the transition they denote. MGNG is applied to construct this automaton automatically. The input data comprise positions of all players of interest stacked to a vector in a fixed ordering. The selection of players determines the type of analysis. One can look at a single team or at both teams to summarize the game. If only players with a specific role are selected, e.g. the defense line, the interplay within this group can be analyzed. An interesting investigation is formed by the reduction of data to the positions of a single player with its associated opponent. There are reams of more analyses, all having in common that they are easy to instantiate by simply selecting a subgroup of players and they result in comprehensible graphs. These facts emphasize their potential use by nontechnical persons like coaches, because our approach can be applied as a kind of black box solution. Figure 8.5 depicts an exemplary analysis of the defense line of a single team. The MGNG was trained by passing 30 times (epochs) over the trajectories. Parameters were set as follows: $\epsilon_b = \epsilon_{c_b} = 0.1$, $\epsilon_n = \epsilon_{c_n} = 0.0001$, $\gamma = 50$, $\lambda = 30$, $\eta = 0.95$, $\theta = 50$, $\beta_t = 0.5$ and $\alpha_t = 0.8$.

8.3 Grounded Action Models

As already mentioned, this work is part of the ASPOGAMO research project [23]. The central part of the analysis system of ASPOGAMO is the acquisition of an informative model of the observed games. This model is constructed by applying

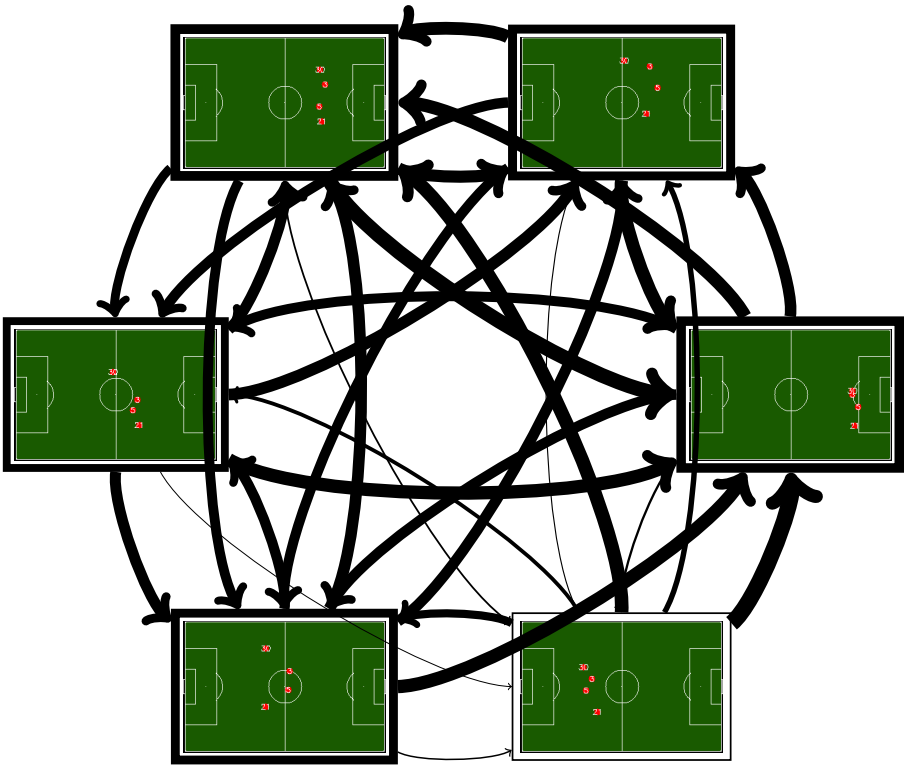


Figure 8.5: The probabilistic automaton shown was extracted by MGNG from a single halftime, selecting the defensive players only. Probabilities of the transitions are visualized by the line width of their corresponding arrow; the probability of being in a specific state is depicted by the line width of its surrounding box.

statistical learning to a number of games, resulting in a model of the type of sports in general. The model provides not only a high abstraction level for investigating the game as a whole, but also allows performing detailed analyses of special aspects of the game. The system adapts the model automatically to data of new games whenever they are available.

The analysis system needs a common interface for user queries. We chose the Web Ontology Language (OWL) as the interfacing language. OWL constitutes an instance of description logics (DL) [7]. Description logics exhibit a balance between expressiveness and complexity because efficient inference methods exist, but the language is weaker than first-order logics. Classes (also known as concepts) and individuals, the subclass relationship and the properties of objects constitute the building blocks of DL. Classes define a group of individuals that belong together because they share common properties. The classes together

with the subclass relationship form a specialization hierarchy on classes: the ontology. Properties represent binary predicates, and can be used to state relationships between individuals or from individuals to data values. The language is designed in a set-theoretic manner and is therefore well suited for interrogator-responder systems (c.f. the OWL Query Language). OWL is specified as XML and is in common use in the semantic web community.

The game model of ASPOGAMO is stated in OWL and can be divided into two parts: the static part, which does not depend on a specific game or class of games, and the dynamic part, which must be adapted for the context it is used in.

The static part contains the domain knowledge about the specific type of sports as concrete definitions that are independent of a specific game or game class. This part can be specified once for the various types of sports and applied to all games in the same way. It contains general rules and classifications, such as the offside rule, what a successful shot is or the number of allowed substitutions, to name only a few examples.

The dynamic part consists of concepts that relate to the same definition but have different specifications. For example, the concept of scoring opportunity has a well defined meaning, but depends on the quality of the involved teams. A given situation would be a scoring opportunity e.g. for a World Cup player, because he would always score in that circumstance, but it could not be classified as such for a junior player, for example. So some (unfortunately most) parts of the model are specific to the game, to the teams or to the players and therefore must be defined in relation to their context. We solve this problem by enabling our system to automatically learn the dynamic part from observed games using machine learning techniques and given the abstract meaning, which is consistent through all contexts.

We propose GrAM (Grounded Action Models), a novel integration of actions and action models into the knowledge representation and inference mechanisms for tactical sports video analysis. Grounded action models correspond to agent behavior and can be specified explicitly and implicitly.

The explicit representation is an action-class-specific set of Markov logic rules that predict action properties (Markov Logics is a kind of probabilistic first-order logics and has been proposed as an interface layer for AI by Domingos [57]). Stated implicitly, an action model defines a data-mining problem that, when executed, computes the model's explicit representation. When inferred from an implicit representation, the prediction rules are learned from a set of training examples, or in other words, grounded in the respective context, enabling the system to forecast typical behavior in turn. Therefore, GrAM allows for the functional and thus adaptive specification of concepts such as the

class of situations, in which a special action is typically executed successfully, or the concept of protagonists that tend to execute certain kinds of actions. The explicit representation of an action class is seamlessly integrated into the description logics inference mechanism by taking the highly probable rules (according to a threshold) as the concept definition.

The implementation of GrAM includes an extension of OWL that allows the definition of classes in a functional manner as well as the equipment of the Java Theorem Prover (JTP), a hybrid reasoner for OWL, with additional mechanisms that allow automated acquisition of action models and the solution of a variety of inference tasks for actions, action models and functional descriptions.

As an example, we examine a model for passes in soccer games. We can easily specify statically that a successful pass denotes a ball contact by one player of a team followed by ball contact of another player of the same team. Unsuccessful passes, however, are much harder to define in a general way. But we can state that the characteristics of unsuccessful passes should be similar to passes in regard to ball velocity, the direction the ball was played in, or the positions of team members and opponents. This definition holds for all games, even if the attributes are highly correlated with the abilities of the players and therefore depend on the league or competition in which the specific game took place. Transforming this static definition into a data-mining task, the system can learn rules for each league that specify which ball actions should be considered as passes and which ones rather as shots. Taking the known set of successful passes, shots and dribbling as training data, a decision tree [196] or respectively a regression tree [262] is automatically learned for each binary classification (pass or no pass, shot or no shot, dribbling or no dribbling). The tree is split into several rules by logical disjunction of the nodes on all possible paths beginning at the root and ending at a leaf. The abstraction comes into play by the pruning of the tree, which is part of the learning algorithm to avoid overfitting. The rules are transformed to descriptions in the ontology. In this manner, they are integrated into the knowledge base. The description (consisting of the rules) as well as the concept itself (referring to the classification of instances), can be transparently inspected.

There is also a second class of definitions that contains a dynamic part. Most of the continuous attributes of actions in team sports are discretized into classes like slow, normal and fast, or short to long. This is usually done by simple thresholding at predefined ranges. If we look closely at these kinds of concepts, the sensitiveness to their context becomes evident. The velocity of a fast sprint in an international competition obviously differs from values for a burst of speed in the minor league. Still, there is a common definition that partitions the usual velocity range in a predefined number of parts, naming the part with the highest

velocities as fast. This partitioning can be achieved by data-mining techniques called clustering [72], which iteratively find a locally optimal subdivision. In addition, smooth fragmentation can be automatically achieved by using probabilistic assignment to clusters obtained by fuzzy clustering [72]. The partition and naming of each part is again transformed into descriptions in the ontology to provide a seamless interface.

All these models, the static and dynamic ones, are accessible to the system due to their integration in the ontology. Their definition and semantics can be inspected and analyzed by the user, resulting in more alternatives for analyzing a game and more objectivity, which is due to the transparency of the models. For example, the user can retrieve the scoring opportunities of the teams and then analyze how they might arise by interpreting the rules generated for the concept. A concept for all situations in which the ball was lost could be stated, and from the resulting rules, some reasons for the failed ball action may be derived. Clustering situations in which the ball was lost can help in gaining insights concerning how the opposing team forced the loss of ball possession, etc.

The proposed framework of GrAM has been successfully transferred to mobile manipulation actions performed by an autonomous household robot in simulation and reality [238].

8.4 Conclusions

In this chapter, existing methods for tactical sports video analysis based on trajectories have been surveyed. The approaches cover the three subfields of situational analysis, unsupervised team behavior analysis and classification of team behavior. We contribute to the state-of-the-art by proposing modeling a game as a Markov process of player formations. The underlying probabilistic automaton is extracted automatically by Merge Growing Neural Gas (MGNG), a novel self-organizing network for time-series clustering. The proposed MGNG constitutes an extension of LateGNG, incorporating temporal information by an additional context vector for every cluster prototype. The context vector represents a fractal encoding of the previously presented sequence elements. MGNG is able to learn temporal automatons with the same accuracy as state-of-the-art MNG [232] but in an online manner and with improved runtime performance.

Further, we introduced grounded action models as a novel means of providing a descriptive logics interface extended by functional descriptions. These new constructs allow definition of context-dependent concepts. Instantiations of these concepts are deduced by solving data-mining tasks transparently in the inference process. Grounded action models prescind natural descriptions

of action-related concepts from specific instantiations that may depend on the skills of the protagonist. This fact is especially beneficial for modeling sports, since most of the concepts used are action-related and skill-dependent, such as scoring chances or room for maneuvers.

Chapter 9

Conclusions

This dissertation has presented a thorough investigation of automated tracking for sports video analysis. We summarize the work of this thesis in the next section, recapitulating the contributions afterwards. Finally, we present an overview of future work.

9.1 Compendium

This dissertation has investigated real-time tracking systems of identities in sports videos for computer-aided analysis. The work was motivated by the need for automated analysis and indexing in the ever growing multimedia databases as well as by various concrete applications within sports. The research was indexed as a cognitive system that touches various sub-fields of artificial intelligence and computer vision.

We proposed a distributed cognitive framework for tracking systems in sports, pointing out the main components and their interconnections. Several information sources provide evidences about locations and identities of players; their fusion is done according to probability theory and the Bayesian approach. The idea of a bootstrapping system was introduced, in which player models are built and adapted online and evidence is distributed over the different information sources.

With this large picture in mind, the complete data processing cycle was investigated, ranging from acquisition of the input signal to the supply of abstract sports video analysis.

After an overview of passive and active sensors, we focused on signals captured by static as well as dynamic standard cameras and broadcasted material for the remainder of the thesis. The detection of players in sports videos is achieved by foreground segmentation. After a review of general methods for

this task, a robust approach for segmentation and localization of athletes in soccer videos was detailed as a specialized preprocessing step, exploiting the homogeneity of the green.

Because temporal and spatial consistency allow for the concept of identity in the first place, this constraint is utilized to track and fuse evidences about identities. We proposed the Rao-Blackwellized Resampling Particle Filter as an innovative and general multi-target tracking algorithm. Assumptions and a theoretical derivation of this variant of an SIR particle filter were explicated in detail. A flexible approach to restrict the multiplicity of measurements assigned to a single target was integrated in the tracking framework. We went into implementational details because the real-time claim demands efficient methods, especially for this time-critical process. Parallel computation and computational adaptivity to the uncertainty in the association is a design feature of RBRPF that helps keep runtime low; the complexity analysis revealed linear runtime in the number of measurements and targets. We demarcated our RBRPF approach of the state-of-the-art in theory and praxis. The performance of the proposed method was demonstrated in simulation, in basketball as well as in ant tracking experiments.

Because team sports athletes are usually assigned specific roles, which manifest themselves in spatial arrangements, we considered the use of positions for identifying players. Several methods based on relative and absolute position models were developed and compared according to their performance in determining the correct label for each team member. Relative sorting based on tactical line-ups gathered from TV or the web was revealed to be the method of choice for initializing the tracker at kick-offs. Learned models representing absolute spatial information can be utilized as soon as sufficient data are available. Statistical aggregation and vector quantization were considered in regard to building such models online. We introduced LateGNG as an extension of growing self-organizing neural networks for the automated abstraction of continuous data. The learning of complete formations by LateGNG performed best in identification, while individual position clusters provided by LateGNG are suited for this task if only partial data are available for learning and classification.

Further we researched the usefulness of appearance for automated association of foreground regions to player labels. Two approaches were considered: color histograms and the complete texture provided the space for nearest neighbor search. Although both methods are effective for representing the individual characteristics of the athletes, players of the same team are often hard to distinguish. Color information was shown to be superior to texture for identification due to the high variability of appearance of each player over time. As expected,

the identification rate decreases together with the resolution of the players.

While the proposed methods of each processing step were evaluated separately, experiments with challenging soccer video sequences demonstrated the effectiveness of the system as a whole. Three possible scenarios were investigated: multiple static cameras capturing the playing field in parallel, a single panning, tilting and zooming camera and broadcasted material. We provided comprehensible performance metrics for tracking soccer matches in full length and covering all kinds of exceptions from typical play.

Finally, the use of the gathered trajectories for tactical sports analysis was examined. We proposed Merge Growing Neural Gas (MGNG) for online clustering of time-series. The self-organizing MGNG network was used to extract probabilistic automatons of formations which summarize team behavior as a Markov process. The subset selection of the players provides a rich although simple source for comprehensive behavior analysis. An information retrieval system based on description logics allows defining context-related concepts of the specific sports domain in a general form, abstracting away from the skill level present in specific games. We explained how these concepts can be integrated into the inference process by transparently solving data-mining tasks.

9.2 Contributions

This thesis contributes to the current research in cognitive systems by implementing a computer-aided video analysis system for soccer. The process run includes the robust extraction of player regions, player localization and tracking as well as adaptive re-identification by appearance and player role. All of these tasks can be computed in real-time. The resulting trajectories can be automatically summarized to analyze team behavior. In addition, a conceptualization framework combining logics and data-mining allows for further analyses. This dissertation is the first to present experimental results on full length soccer games. The system has shown to be applicable for challenging video footage recorded in various setups, also including broadcasted material. The 22 players and the referee can be correctly tracked for about 86% of the time.

Along the way, several general methods were developed or extended in an innovative way:

We proposed the Rao-Blackwellized Resampling Particle Filter as a multiple target tracking method, estimating formations based on the sampling of associations. A thorough deduction from Bayesian theory was given which explained the assumptions made. This approach is suitable for real-time tracking since its runtime complexity is linear in the number of measurements and targets due to smart resampling and memoization. Experimental results of different chal-

lenging domains demonstrated the generality of our approach and revealed its supremacy over current state-of-the-art algorithms.

Novel incremental methods were developed to identify players based on their appearance or spatial relationships between them. Among other things, a self-organizing growing neural network approach was extended to achieve an adaptive vector quantization online. The improved networks can cluster continuous data and time-series incrementally. In addition to their ability to build models of appearance and positions for identification, they were utilized for unsupervised team behavior analysis and to form a general unsupervised learning tool.

9.3 Outlook on Future Work

Xu et al. [265] identified four future topics in sports video analysis: cross-media semantic annotation and retrieval, generic solution and knowledge integration, robust performance on large scale test data as well as user study and personalization. Despite the fact that this thesis contributed to all of these topics, many open issues remain for future research.

The framework and algorithms presented can be transferred to the field of surveillance. The main difference to sports is that the assumption of a fixed number of targets (closed world assumption) must be dropped. The necessary steps for extending RBRPF to track birth and death have been outlined in section 4.1.2, but the approach remains to be evaluated empirically. Additionally, alternative foreground segmentation methods must be applied as described in section 3.3. We expect a better identification rate based on appearance for surveillance, because the targets will not be as similar as soccer players of the same team.

We already mentioned how ball tracking and action recognition could improve player identification. These tasks, however, cannot be solved in isolation to provide evidence, but are intertwined with player tracking and require bidirectional communication. As our framework suggests, many preprocessing results can be shared between these modules. In contrast to player tracking, ball tracking requires handling long periods of continuous occlusion and the modeling of positions and motions in three-dimensions.

Because adaptivity is desirable in many applications, we need to understand bootstrapping systems and self-reference in more detail. Although such systems have been modeled by attractors of differential equations already, implications for the design of artificial systems remain open. Practical conditions for the stability of such systems are required to ensure their reliability.

The tracking and fusion process may still be improved in different ways. Variable-Bandwidth Density-based Fusion (VBDF) [51] states an alternative to

the Kalman update for the merger of multiple measurements, which are associated with a single target. VBDF results in positions and covariances of fused estimates that better fit intuition and therefore may improve tracking results. The runtime for this time-critical step must be evaluated. Incorporating learned motion models for players into tracking can improve prediction of their positions – especially for times when they are outside the field of view – and therefore ease the sampling of associations. Despite their potential for tracking, they will also be of interest for physiotherapists and sports medicine. A prerequisite is a sufficient training set of trajectories covering all common motions. The challenge is to segment the trajectories into smaller pieces without global or future information being available. In addition to these improvements, measurements could be discarded if the localization module is given the expected number of measurements to detect inside a region. In this way, the idea of negative information handling could also be extended to long-lasting occlusions that are due to tight man-marking. The effect of this stronger connection of tracking and localization remains to be evaluated, however.

There are various unconsidered image features that can be used for identifying players at a distance by their appearance as well. Texture can be represented as a bag of features, or the popular SIFT [165] or SURF [21] can be used. Although these features are mainly designed for rigid and highly structured objects, and preliminary experiments with SURF have shown discouraging results, a clever combination of these could reveal their potential to identify non-rigid players in foreground regions. Alternatively, our proposed identification by color histograms could be extended towards spatial information by partitioning the region and creating separate histograms for each sub-region. Polar histograms could be used as well, subdividing the region into segments of a circle and possibly providing a more promising approach. Independent of the models used, various two-dimensional views of the same player could be exploited to extract a textured three-dimensional model of this player for improved identification or later visualization.

Despite their inspiration by nature, self-organizing networks are often said to suffer from high computational requirements. The search for the best matching unit inside the self-organizing networks constitutes the step that demands most runtime for learning as well as for classification. This step could be accelerated by the use of approximate nearest neighbor search. We did some preliminary experiments transforming the images into the (Haar) wavelet domain and therefore enabling a kind of hierarchical search. This is advantageous because the first pixel in the wavelet domain contains the coarse information and further pixels entail more and more details. However, not only the data but also the update step itself must be transferred to the wavelet domain because the computations

of wavelet transforms of the nodes and their inverse cancels the computational surplus gained from accelerated search. Alternatively, the Hilbert encoding [45] of data as described by [1] could be used for faster nearest neighbor search. Thus, transfer of the update step into the new domain forms the crucial point as well.

Spatial data gathered by tracking systems open a rich source for countless kinds of analyses. Future research must develop innovative (semi)-automated methods for higher level abstractions of the gathered trajectories. The analyses should be non-parametric or intuitively adjustable in nature to be accepted in practice. Xu et al. [265] remark, that "[o]ne limitation of current sports video analysis research and sports video services is [that they seldom] consider users' real needs." Therefore, the collaboration of sports science and informatics must be intensified: sport scientists should put more effort into formal and mathematical modeling of sports and attach it to what is technically feasible, while computer scientists should look more for the demands in the field and investigate novel frameworks for solving the real (albeit difficult) problems instead of solving problems determined by currently popular frameworks.

Bibliography

[1] David J. Abel and David M. Mark. A comparative analysis of some two-dimensional orderings. *Int. Journal of Geographical Information Systems*, 4:21–31, 1990. 126, 174

[2] Y.Demiris A.Dearden and O. Grau. Tracking football player movment from a single moving camera using particle filters. In *The 3rd European Conf. on Visual Media Production (CVMP 2006)*, 2006. 38

[3] Impire AG. Vis.track. `http://www.vistrack.de`. 18

[4] E.L. Andrade, E. Khan, J.C. Woods, and M. Ghanbari. Player classification in interactive sport scenes using prior information region space analysis and number recognition. In *Proc. of Int. Conf. on Image Processing (ICIP)*, volume 3, pages 129–132, 2003. 118

[5] Andreas Andreakis, Nicolai von Hoyningen-Huene, and Michael Beetz. Incremental unsupervised time series analysis using merge growing neural gas. In José Carlos Príncipe and Risto Miikkulainen, editors, *Proc. of Int. Workshop on Advances in Self-Organizing Maps (WSOM)*, volume 5629 of *Lecture Notes in Computer Science*, pages 10–18. Springer, 2009. 163

[6] M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Trans. on Signal Processing*, 50(2), Feb 2002. 26, 50

[7] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2 edition, September 2007. 164

[8] Noboru Babaguchi and Ramesh Jain. Event detection from continuous media. In *Proc. of Int. Conf. on Pattern Recognition (ICPR)*, 1998. 44

[9] Noboru Babaguchi, Yoshihiko Kawai, and Tadahiro Kitahashi. Event Based Video Indexing by Intermodal Collaboration. *IEEE Transactions on Multimedia*, 9(2):68–75, 2002. 44

[10] Tucker Balch, Adam Feldman, and Wesley Wilson. Assessment of an rfid system for animal tracking. Technical report, Georgia Institute of Technology, October 2004. 34, 81

[11] Dana H. Ballard and Christopher M. Brown. *Computer Vision*. Prentice Hall, 1982. 119

[12] Yaakov Bar-Shalom. Extension of the probabilistic data association filter to multitarget environments. In *Symp. on Nonlinear Estimation*, San Diego, CA, September 1974. 55

[13] Yaakov Bar-Shalom and Thomas E. Fortmann. *Tracking and Data Association*, volume 179 of *Mathematics in Science and Engineering*. Academic Press, 1988. 8, 26, 50, 55, 65, 70

[14] Yaakov Bar-Shalom, Thomas E. Fortmann, and M. Scheffe. Joint probabilistic data association for multiple targets in clutter. In *Proc. of Conf. on Information Sciences and Systems*, Princeton, NJ, March 1980. 55, 141

[15] Yaakov Bar-Shalom and A. G. Jaffer. Adaptive nonlinear filtering for tracking with measurements of uncertain origin. In *IEEE Conf. on Decision & Control*, pages 243–247, New Orleans, LA, December 1972. 52

[16] Yaakov Bar-Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data association. *Automatica*, 11:451–460, September 1975. 52

[17] Lluis Barceló, Xavier Binefa, and John R. Kender. Robust methods and representations for soccer player tracking and collision resolution. In *Proc. of Int. Conf. on Image and Video Retrieval (CIVR)*, pages 237–246, 2005. 55

[18] Francois Bardet, Thierry Chateau, and Datta Ramadasan. Real-Time Multi-Object Tracking with Few Particles. In AlpeshKumar Ranchordas and Helder Araujo, editors, *Int. Conf. on Computer Vision Theory and Applications (VISAPP)*, volume 1, pages 456–463, Lisboa, Portugal, Feb. 2009. INSTICC, INSTICC press. 57, 78

[19] Ricardo M. L. Barros, Milton S. Misuta, Rafael P. Menezes, Pascual J. Figueroa, Felipe A. Moura, Sergio A Cunha, Ricardo Anido, and Neucimar J. Leite. Analysis of the distances covered by first division brazilian soccer players obtained with an automatic tracking method. *Journal of Sports Science and Medicine*, 6:233–242, 2007. 156

[20] Adrien Bartoli, Navneet Dalal, and Radu Horaud. Motion panoramas. *In Journal of Computer Animation and Virtual Worlds*, 15(5):501–517, nov 2004. 38

[21] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded Up Robust Features. *Computer Vision and Image Understanding (CVIU)*, 110:346–359, 2008. 119, 173

[22] Thomas Bebie and Hanspeter Bieri. Soccerman - reconstructing soccer games from video sequences. In *Proc. of Int. Conf. on Image Processing (ICIP)*, pages 898–902, 1998. 5, 37

[23] M. Beetz, F. Fischer, S. Flossmann, B. Kirchlechner, A. Unseld, and C. Holzer. Watching football with the eyes of experts: Integrated intelligent systems for the automatic analysis of (simulated) football games. In *Conf. dvs-Section Computer Science in Sport*, 2004. 163

[24] Michael Beetz, Jan Bandouch, Suat Gedikli, Nico von Hoyningen-Huene, Bernhard Kirchlechner, and Alexis Maldonado. Camera-based observation of football games for analyzing multi-agent activities. In *Proc. of Int. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, 2006. 38

[25] Michael Beetz, Martin Buss, and Dirk Wollherr. Cognitive Technical Systems – What Is the Role of Artificial Intelligence? In J. Hertzberg, M. Beetz, and R. Englert, editors, *KI 2007*, number 4667 in LNAI, pages 19–42. Springer, 2007. 7

[26] Michael Beetz, Suat Gedikli, Jan Bandouch, Bernhard Kirchlechner, Nico v. Hoyningen-Huene, and Alexander Perzylo. Visually tracking football games based on TV broadcasts. In *Proc. of Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 2007. 38

[27] Michael Beetz, Nicolai v. Hoyningen-Huene, Bernhard Kirchlechner, Suat Gedikli, Francisco Siles, Murat Durus, and Martin Lames. ASPOGAMO: Automated Sports Game Analysis Models. *Int. Journal of Computer Science and Sports (IJCSS)*, 2009. 6, 39

[28] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: The clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008. 139

[29] Marco Bertini, Alberto Del Bimbo, and Walter Nunziati. Automatic detection of player's identity in soccer videos using faces and text cues. In *Proc. of Int. Multimedia Conf. (MM)*, pages 663–666, 2006. 90, 118

[30] I. Bhandari, E. Colet, J. Parker, Z. Pines, R. Pratap, and K. Ramanujam. Advanced scout: Data mining and knowledge discovery in nba data. *Data Mining and Knowledge Discovery*, 1997. 157

[31] Samuel Blackman and Robert Popoli. *Design and Analysis of Modern Tracking Systems*. Artech House Radar Library, 1999. 72

[32] Miklos Bona. *Combinatorics of Permutations*. Chapman & Hall/CRC, 2004. 106

[33] Léon Bottou. Stochastic learning. In O. Bousquet, G. Raetsch, and U. von Luxburg, editors, *Advanced Lectures on Machine Learning*, number 3176 in LNAI, pages 146–168. Springer, 2003. 30

[34] Ronald J. Brachman. Systems that know what they're doing. *IEEE Intelligent Systems*, pages 67–71, 2002. 7

[35] Jack E. Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems Journal*, 4(1):25–30, January 1965. 40

[36] Yizheng Cai, N. De Freitas, and J. J. Little. Robust visual tracking for multiple targets. In *Proc. of Europ. Conf. on Computer Vision (ECCV)*, 2006. 19, 56

[37] David Camacho, Maria D. R-Moreno, David F. Barrero, and Rajendra Akerkar. Semantic wrappers for semi-structured data extraction. *Computing Letters (Cole)*, 4(1):1–14, 2008. 89

[38] Otávio Augusto S. Carpinteiro. A Hierarchical Self-Organizing Map Model for Sequence Recognition. In *Proc. of Int. Conf. on Artificial Neural Networks (ICANN)*, volume 2, pages 815–820. Springer, London, 1998. 158

[39] G. Casella and C.P. Robert. Rao-blackwellisation of sampling schemes. *Biometrika*, 83(1):81–94, March 1996. 54

[40] Ryan Cassel, Christophe Collet, and Rachid Gherbi. Real-time acrobatic gesture analysis. In S. Gibet, N. Courty, and J.-F. Kamp, editors, *Lecture Notes in Artificial Intelligence (LNAI)*, pages 88–99. Springer, 2006. 136

[41] Rick Cavallaro. The foxtrax hockey puck tracking system. *IEEE Computer Graphics and Applications*, 1997. 18, 35

[42] J. Chandaria, G. Thomas, B. Bartczak, K. Koeser, R. Koch, M. Becker, G. Bleser, D. Stricker, C. Wohlleber, M. Felsberg, F. Gustafsson, J. Hol, T. B. Schön, J. Skoglund, P. J. Slycke, and S. Smeitz. Real-Time Camera Tracking in the MATRIS project. In *Proc. of International Broadcasting Convention (IBC)*, September 2006. 37

[43] Geoffrey J. Chappell and John G. Taylor. The Temporal Kohonen map. *Neural Networks*, 6(3):441–445, 1993. 158

[44] Bingqi Chen and Zhiqiang Wang. A statistical method for analysis of technical data of a badminton match based on 2-d seriate images. *Tsinghua Science and Technology*, 12(5):594–601, October 2007. 19

[45] Ningtao Chen, Nengchao Wang, and Baochang Shi. A new algorithm for encoding and decoding the Hilbert order. *Software: Practice and Experience (SP&E)*, 37:897–908, 2006. 174

[46] Rong Chen and Jun S. Liu. Mixture kalman filters. *Journal of Royal Statistics Society*, 2000. 54, 61

[47] Zhe Chen. Bayesian filtering: From kalman filters to particle filters, and beyond. Technical report, McMaster University, Hamilton, Ontario, Canada, 2003. 26, 29, 50

[48] Lior Cohen, Gil Avrahami, Mark Last, and A. Kandel. Info-fuzzy algorithms for mining dynamic data streams. *Applied Soft Computing (ASOC), Special Issue on Soft Computing for Dynamic Data Mining*, 8(4):1283–1294, September 2008. 122

[49] Lior Cohen, Gil Avrahami, Mark Last, Abraham Kandel, and Oscar Kipersztok. Incremental classification of nonstationary data streams. In *Proc. of Int. Workshop on Knowledge Discovery in Data Streams (IWKDDS)*, pages 117–124, Porto, Portugal, October 2005. 122

[50] F. Coldefy and P. Bouthemy. Unsupervised soccer video abstraction based on pitch, dominant color and camera motion analysis. In *Proc. of Int. Multimedia Conf. (MM)*, pages 268–271, 2004. 36, 44

[51] Dorin Comaniciu. Nonparametric information fusion for motion estimation. In *Proc. of Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2003. 172

[52] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 23(6):681–685, June 2001. 120

[53] T. F. Cootes and C. J. Taylor. Active shape models - 'smart snakes'. In *Proc. British Machine Vision Conf. (BMVC)*, pages 266–275, 1992. 119

[54] Ingemar J. Cox and Matt L. Miller. On finding ranked assignments with application to multitarget tracking and motion correspondences. *IEEE Trans. on Aerospace and Electronic Systems*, 31(1):486–489, January 1995. 55, 95

[55] Anthony Dekker. Kohonen neural networks for optimal colour quantization. *Network: Computation in Neural Systems*, 5:351–367, 1994. 123

[56] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39:1–38, 1977. 120

[57] Predro Domingos. What's missing in ai: The interface layer. In P. Cohen, editor, *Artificial Intelligence: The First Hundred Years*. AAAI Press, Menlo Park, CA, 2006. 165

[58] Tiziana D'Orazio, Marco Leo, Paolo Spagnolo, Pier Luigi Mazzeo, Nicola Mosca, and M. Nitti. A visual tracking algorithm for real time people detection. In *Proc. of Int. Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, 2007. 19

[59] A. Doucet. On sequential Monte Carlo methods for Bayesian filtering. Technical report, Dept. End., Univ. Cambridge, UK, 1998. 53

[60] Ling-Yu Duan, Min Xu, Tat-Seng Chua, Qi Tian, and Chang-Sheng Xu. A mid-level representation framework for semantic sports video analysis. In *Proc. of ACM Int. Conf. on Multimedia*, pages 33–44, New York, NY, USA, 2003. ACM. 20

[61] A.A. Efros, A.C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, volume 2, pages 726–733, 2003. 39, 131, 136

[62] Ahmet Ekin, A. Murat Tekalp, and Rajiv Mehrotra. Automatic soccer video analysis and summarization. *IEEE Trans. on Image Processing*, 12(7):796–807, 7 2003. 44

[63] Murat Ekinci. A new approach for human identification using gait recognition. In Marina L. Gavrilova, Osvaldo Gervasi, Vipin Kumar, Chih Jeng Kenneth Tan, David Taniar, Antonio Laganà, Youngsong Mun, and Hyunseung Choo, editors, *Computational Science and Its Applications (ICCSA 2006)*, volume 3982 of *Lecture Notes in Computer Science*, pages 1216–1225. Springer, 2006. 136, 137

[64] Murat Ekinci. A new attempt to silhouette-based gait recognition for human identification. In Luc Lamontagne and Mario Marchand, editors, *Canadian Conf. on AI*, volume 4013 of *Lecture Notes in Computer Science*, pages 443–454. Springer, 2006. 119, 136

[65] Pablo A. Estevez, Ricardo Zilleruelo-Ramos, Rodrigo Hernandez, Leonardo Causa, and Claudio M. Held. Sleep Spindle Detection by Using Merge Neural Gas. In *Proc. of Int. Workshop on Advances in Self-Organizing Maps (WSOM)*, 2007. 158

[66] Leonhard Euler. *Memoires de l'academie des sciences de St.-Petersbourg 3 (1809/1810)*, volume 7 of *1*, chapter Solutio quaestionis curiosae ex doctrina combinationum, pages 57–64. Imperatorskaia Akademiya Nauk, 1811. 106

[67] N. R. Euliano and J. C. Principe. A Spatio-Temporal Memory Based on SOMs with Activity Diffusion. In Oja, editor, *Kohonen Maps*, pages 253–266. Elsevier, 1999. 158

[68] Dirk Farin, Jungong Han, and Peter H. N. de With. Fast camera calibration for the analysis of sport sequences. In *International Conf. on Multimedia and Expo (ICME)*, 2005. 37

[69] Dirk Farin, Susanne Krabbe, Peter H. N. de With, and Wolfgang Effelsberg. Robust camera calibration for sport videos using court models. In *SPIE Storage and Retrieval Methods and Applications for Multimedia*, 2004. 37

[70] Igor Farkas and Matthew Crocker. Recurrent networks and natural language: exploiting self-organization. In *Proc. of Annual Meeting of the Cognitive Science Society (CogSci)*, 2006. 158

[71] Igor Farkas and Matthew W. Crocker. Systematicity in sentence processing with a recursive Self-Organizing Neural Network. In *Proc. of Europ. Symposium on Artificial Neural Networks (ESANN)*, 2007. 158

[72] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. From data mining to knowledge discovery in databases. *AI Magazine*, pages 37–54, 1996. 8, 167

[73] P. Fearnhead and P. Clifford. On-line inference for hidden markov models via particle filters. *Journal of Royal Statistics Society*, 65(4):887–899, 2003. 62

[74] A. Feldman, S. Adams, M. Hybinette, and T. Balch. A tracker for multiple dynamic targets using multiple sensors. In *Proc. of Int. Conf. on Robotics and Automation (ICRA)*, pages 3140–3141, April 2007. 81

[75] Adam M. Feldman. *Using observations to recognize the behavior of interacting multi-agent systems.* PhD thesis, Georgia Institute of Technology, August 2008. 80, 81, 158

[76] FIFA. Laws of the game, 2009. 36, 40, 122

[77] Pascual Figueroa, Neucimar Leite, Ricardo M. L. Barros, Isaac Cohen, and Gerard Medioni. Tracking soccer players using the graph representation. In *Proc. of Int. Conf. on Pattern Recognition (ICPR)*, volume 4, pages 787–790, 2004. 56

[78] P.J. Figueroa, N.J. Leite, and R.M.L. Barros. Background recovering in outdoor image sequences: An example of soccer players segmentation. *Image and Vision Computing*, 24(4):363–374, April 2006. 38

[79] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. X, 37, 39

[80] Myron Flickner, Harpreet Sawhney, Wayne Niblack, Jonathan Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, David Steele, and Peter Yanker. Query by image and video content: The qbic system. *Computer*, 28(9):23–32, 1995. 123

[81] Yoav Freund and Robert E. Schapire. A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence*, 14(5):771–780, 1999. 29

[82] Bernd Fritzke. A Growing Neural Gas network learns topologies. In G. Tesauro, D. S. Touretzky, and T. K. Leen, editors, *Advances in Neural Information Processing Systems (NIPS)*, volume 7, pages 625–632. MIT Press, 1995. X, 11, 98, 120

[83] Bernd Fritzke. A self-organizing network that can follow non-stationary distributions. In *Proc. of Int. Conf. on Artificial Neural Networks (ICANN)*, pages 613–618. Springer, 1997. 98

[84] Bernd Fritzke. *Vektorbasierte Neuronale Netze*. Shaker, 1998. 161

[85] J. Gama, R. Fernandes, and R. Rocha. Decision trees for mining data streams. *Intelligent Data Analysis*, 10:23–45, 2006. 121, 122, 129

[86] J.M. Garbarino and E. Billi. A video computerised tool for analysing the trajectories of players in team game. Videotape at Symposium Technology and Sport, L. Katz (Chair), Calgary, Canada, June 2000. 18

[87] Weina Ge and Robert T. Collins. Multi-target data association by tracklets with unsupervised parameter estimation. In *Proc. of British Machine Vision Conf. (BMVC)*, pages 935–944, Leeds, UK, 2008. 57

[88] Suat Gedikli. *Continual and Robust Estimation of Camera Parameters in Broadcasted Sports Games*. PhD thesis, Technische Universität München, 2008. 37, 40, 44

[89] Suat Gedikli, Jan Bandouch, Nico von Hoyningen-Huene, Bernhard Kirchlechner, and Michael Beetz. An adaptive vision system for tracking soccer players from variable camera settings. In *Proc. of Int. Conf. on Computer Vision Systems (ICVS)*, 2007. 38, 55, 66, 120

[90] Nicolas Gengembre and Patrick Pérez. Probabilistic color-based multi-object tracking with application to team sports. Technical Report 6555, INRIA, May 2008. 37, 39, 56, 125, 143

[91] M. Gervautz and W. Purgathofer. A simple method for color quantization: Octree quantization. *Graphics Gems I*, pages 287–293, 1990. 123

[92] Zoubin Ghahramani. Unsupervised learning. In O. Bousquet, G. Raetsch, and U. von Luxburg, editors, *Advanced Lectures on Machine Learning*, number 3176 in LNAI. Springer, 2003. 30

[93] W. R. Gilks. *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC, December 1995. 8, 56

[94] O. Grau, A. Hilton, J. Kilner, G. Miller, T. Sargeant, and J. Starck. A free-viewpoint video system for visualisation of sport scenes. In *Proc. of International Broadcasting Convention (IBC)*, September 2006. 5

[95] O. Grau, M. Prior-Jones, and G.A. Thomas. 3d modelling and rendering of studio and sport scenes for tv applications. In *Proc. of Int. Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, April 2005. 5

[96] O. Grau, G.A. Thomas, A. Hilton, J. Kilner, and J. Starck. A robust free-viewpoint video system for sport scenes. In *Proceeding of 3DTV conference 2007*, April 2005. 5, 18

[97] Markus Hagenbuchner, Alessandro Sperduti, and Ah Chung Tsoi. A Self-Organizing Map for adaptive processing of structured data. *IEEE Trans. on Neural Networks*, 14(3):491– 505, May 2003. 158

[98] Alfian Abdul Halin, Mandava Rajeswari, and Dhanesh Ramachandram. Automatic overlaid text detection, extraction and recognition for high level event/concept identification in soccer videos. In *Int. Conf. on Computer and Electrical Engineering*, pages 587–592, Los Alamitos, CA, USA, 2008. IEEE. 44, 90

[99] B. Hammer, A. Micheli, N. Neubauer, A. Sperduti, and M. Strickert. Self Organizing Maps for Time Series. In *Proc. of Int. Workshop on Advances in Self-Organizing Maps (WSOM)*, pages 115–122, Paris, France, 2005. 158

[100] B. Hammer, A. Micheli, and A. Sperduti. A general framework for unsupervised processing of structured data. In *Proc. of Europ. Symposium on Artificial Neural Networks (ESANN)*, volume 10, pages 389–394, 2002. 158

[101] B. Hammer and T. Villmann. Classification using non standard metrics. In *Proc. of Europ. Symposium on Artificial Neural Networks (ESANN)*, pages 303–316, 2005. 158

[102] X.L. Han, L.F. Wu, X.S. Liu, Z.H. Cheng, and Y. Gong. Offense-defense semantic analysis of basketball game based on motion vector. In *Proc. of Int. Symp. on Image Analysis and Signal Processing (IASP)*, pages 146–149, 2009. 39, 44

[103] Ville Hautamäki. Efficient color quantization by hierarchical clustering algorithms. Master's thesis, University of Joensuu, February 2005. 123

[104] P. Heckbert. Color image quantization for frame buffer display. *Proc. of the ACM SIGGRAPH*, 16:297–307, July 1982. 123

[105] S. Hirano and S. Tsumoto. Finding interesting pass patterns from soccer game records. In *Proc. of. European Conf. on Principles and Practice of Knowledge Discovery in Databases*, volume 3202, pages 209–218, Pisa, 2004. 157

[106] Shaun Holthouse and Igor van de Griendt. Tracking balls in sports. U.S. Patent Application 2009/0048039, February 2009. 34

[107] P. Horridge and S. Maskell. Real-Time Tracking Of Hundreds Of Targets With Efficient Exact JPDAF Implementation. In *Proc. of Int. Conf. on Information Fusion*, pages 1–8, 2006. 78, 79

[108] Weiming Hu, Tieniu Tan, Liang Wang, and Steve Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE Trans. on Systems, Man, and Cybernetics - Part C: Applications and reviews*, 34(3):334–352, August 2004. 8, 136

[109] Yu Huang, Joan Llach, and Sitaram Bhagavathy. Players and ball detection in soccer videos based on color segmentation and shape analysis. In N. Sebe, Y. Liu, and Y. Zhuang, editors, *Int. Workshop on Multimedia Content Analysis and Mining (MCAM)*, number 4577 in Lecture Notes in Computer Science (LNCS), pages 416–425. Springer, 2007. 38, 119

[110] G. Hulten and P. Domingos. Catching up with the data: Research issues in mining data streams. In *Proc. of Workshop on Research Issues in Data Mining and Knowledge Discovery*, 2001. 121

[111] N. Inamoto and H. Saito. Virtual viewpoint replay for a soccer match by view interpolation from multiple cameras. *IEEE Transactions on Multimedia*, 9(6):1155–1166, 2007. 5, 37

[112] S.S. Intille and A.F. Bobick. Recognizing planned, multiperson action. *Computer Vision and Image Understanding (CVIU)*, 81(3):414–445, March 2001. 8, 157

[113] Stephen S. Intille and Aaron F. Bobick. Closed-world tracking. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, pages 672–678, 6 1995. 19

[114] Stephen S. Intille and Aaron F. Bobick. A framework for recognizing multi-agent action from visual evidence. In *Proc. of National Conf. on Artificial Intelligence (AAAI)*, July 1999. 157

[115] Stephen Sean Intille. *Visual Recognition of Multi-Agent Action*. PhD thesis, Massachusetts Institute of Technology MIT, Sept 1999. 89, 157

[116] Michael Isard and Andrew Blake. Condensation - conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998. 53

[117] Gaël Jaffré and Alain Crouzil. Non-rigid object localization from color model using mean shift. In *Proc. of the International Conf. on Image Processing (ICIP)*, pages 317–320, 2003. 45

[118] Andrew H. Jazwinski. *Stochastic Processes And Filtering Theory*. Academic Press, 1970. X, 52, 66

[119] Yonggang Jin and F. Mokhtarian. Variational particle filter for multi-object tracking. In *Proc. of Int. Conf. on Computer Vision*, pages 1–8, Rio de Janeiro, 2007. 56

[120] Neil Johnson. *Learning Object Behaviour Models*. PhD thesis, University of Leeds, Sept 1998. 89

[121] S. Julier and J. Uhlmann. A new extension of the Kalman filter to non-linear systems. In *Proc. of Int. Symp. on Aerospace/Defense Sensing, Simulation and Controls*, Orlando, FL, 1997. X, 52, 66

[122] Bruno Müller Junior and Ricardo de Oliveira Anido. Distributed real-time soccer tracking. In *Proc. of ACM Int. Workshop on Video Surveillance and Sensor Networks*, 2004. 20, 23, 37, 38

[123] R. E. Kalman. A new approach to linear filtering and prediction problems. *Trans. of ASME, Journal on Basic Engineering*, 82:34–45, March 1960. 51

[124] R. E. Kalman and R. Bucy. New results in linear filtering and prediction theory. *Trans. of ASME, Journal on Basic Engineering*, 83:95–108, March 1961. 51

[125] C. H. Kang, J. R. Hwang, and K. J. Li. Trajectory analysis for soccer players. In *Proc. Int. Conf. Data Mining Workshops*, pages 377–381, HongKong, 2006. 156

[126] Jinman Kang, Isaac Cohen, and Gerard Medioni. Soccer player tracking across uncalibrated camera streams. In *IEEE Int. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (PETS)*, 2003. 37, 120

[127] Rickard Karlsson and Frederik Gustafsson. Monte Carlo data association for multiple target tracking. In *Target Tracking: Algorithms and Applications*, volume 1, pages 131 – 135, 2001. 56

[128] Larry Katz. Sport technology research laboratory. `http://www.strc.ucalgary.ca/`. 4

[129] Larry Katz. The role of interactive video, multimedia, and teaching technology in physical education: Toward the year 2000. In G. Tenenbaum, T. Raz-Liebermann, and Z. Artzi, editors, *Proc. of Int. Conf. on Computer Applications in Sport and Physical Education*, pages 22–31, Netanyi, Israel, 1992. 4, 6

[130] T. Kawashima, K. Yoshino, and Y. Aoki. Qualitative image analysis of group behavior. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 690–693, 1994. 155

[131] Saad M. Khan and Mubarak Shah. Tracking multiple occluding people by localizing on multiple scene planes. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 31:505–519, March 2009. 143

[132] Zia Khan, Tucker Balch, and Frank Dellaert. Efficient Particle Filter-Based Tracking of Multiple Interacting Targets Using an MRF-based Motion Model. In *Proc. of Int. Conf. on Intelligent Robots and Systems (IROS)*, volume 1, pages 254–259, October 2003. 56

[133] Zia Khan, Tucker Balch, and Frank Dellaert. A Rao-Blackwellized Particle Filter for EigenTracking. In *Proc. of Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 980–986, June 2004. 54

[134] Zia Khan, Tucker Balch, and Frank Dellaert. An MCMC-based Particle Filter for Tracking Multiple Interacting Targets. In *Proc. of European Conf. on Computer Vision (ECCV)*, volume 2 of *LNCS 3024*, pages 279–290, October 2004. 56

[135] Zia Khan, Tucker Balch, and Frank Dellaert. MCMC Data Association and Sparse Factorization Updating for Real Time Multitarget Tracking with Merged and Multiple Measurements. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 28(12):1960–1972, December 2006. X, 10, 56, 60, 74, 77, 78, 83, 84, 85, 139, 141

[136] Hyunwoo Kim and Ki Sang Hong. Soccer video mosaicing using self-calibration and line tracking. In *Proc. of Int. Conf. on Pattern Recognition (ICPR)*, volume 1, pages 592–595, 2000. 38

[137] Stephan Kirstein, Heiko Wersing, and Edgar Körner. Online Learning for Object Recognition with a Hierarchical Visual Cortex Model. In *Proc. of Int. Conf. on Artificial Neural Networks (ICANN)*, pages 487–492, 2005. 121

[138] Hiroaki Kitano, Minoru Asada, Yasuo Kuniyoshi, Itsuki Noda, and Eiichi Osawa. RoboCup: The robot world cup initiative. In *Proc. of Int. Conf. on Autonomous Agents (AGENTS)*, pages 340–347, 1997. 8

[139] Teuvo Kohonen. *Self-Organizing Maps*. Springer, Berlin, 3 edition, 2001. 98, 120

[140] Y. Kong, X.Q. Zhang, Q.D. Wei, W.M. Hu, and Y.D. Jia. Group action recognition in soccer videos. In *Proc. of Int. Conf. on Pattern Recognition (ICPR)*, pages 1–4, 2008. 39

[141] Timo Koskela, Markus Varsta, Jukka Heikkonen, and Kimmo Kaski. Temporal sequence processing using Recurrent SOM. In *Proc. of Int. Conf. on Knowledge-Based and Intelligent Information & Engineering Systems (KES)*, pages 290–297. IEEE, 1998. 158

[142] M. Kristan, J. Pers, S. Kovacic, and A. Leonardis. A local-motion-based probabilistic model for visual tracking. *Pattern Recognition*, 42(9):2160–2168, September 2009. 56

[143] M. Kristan, J. Pers, M. Perse, and S. Kovacic. Closed-world tracking of multiple interacting targets for indoor-sports applications. *Computer Vision and Image Understanding (CVIU)*, 113(5):598–611, May 2009. 19, 56

[144] Masao Kubo and Yukinori Kakazu. Acquisition of the various coordinated motions of multi-agent system on soccer game. In *Conf. on Evolutionary Computation*, pages 686–691. IEEE, 1994. 157

[145] H.W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955. 94

[146] Matthew Kyan and Ling Guan. Local variance driven self-organization for unsupervised clustering. In *Proc. of Int. Conf. on Pattern Recognition (ICPR)*, volume 3, pages 421–424, 2006. 98

[147] Dimitrios Lambrinos, Christian Scheier, and Rolf Pfeifer. Unsupervised Classification of Sensory-Motor states in a Real World Artifact using a Temporal Kohonen Map. In *Proc. of Int. Conf. on Artificial Neural Networks (ICANN)*, volume 2, pages 467–472. EC2, 1995. 158

[148] Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999. 120

[149] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*, 13, 2001. 120

[150] Dar-Shyang Lee. Effective gaussian mixture learning for video background subtraction. *Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 27:827–832, May 2005. 40

[151] Marco Leo, Nicola Mosca, Paolo Spagnolo, Pier Luigi Mazzeo, Tiziana D'Orazio, and Arcangelo Distante. Real-time multiview analysis of soccer matches for understanding interactions between ball and players. In *Proc. of Int. Conf. on Content-based Image and Video Retrieval (CIVR)*, volume 1, pages 525–534, Niagara Falls, Canada, 2008. ACM. 19, 37

[152] E. Levina and P. Bickel. The earth mover's distance is the mallows distance: some insights from statistics. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, volume 2, pages 251–256, 2001. 124

[153] Baoxin Li and M. Ibrahim Sezan. Event detection and summarization in sports video. In *Proc. of Workshop on Content-based Access of Image and Video Libraries (CBAIVL)*, page 132, Washington, DC, USA, 2001. IEEE Computer Society. 44

[154] F.H. Li and R.J. Woodham. Video analysis of hockey play in selected game situations. *Image and Vision Computing (IVC)*, 27(1-2):45–58, January 2009. 19, 158

[155] J. Li, W. Ng, S. Godsill, and J. Vermaak. Online Multitarget Detection and Tracking using Sequential Monte Carlo Methods. In *Proc. of Int. Conf. on Information Fusion*, pages 115–121, 2005. 56

[156] Yan Li, A. Dore, and James Orwell. Evaluating the performance of systems for tracking football players and ball. In *IEEE Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, pages 632–637, Los Alamitos, CA, USA, 2005. IEEE Computer Society. 139, 141

[157] LiberoVision. Liberovision. `http://www.liberovision.com`. 18

[158] D. Liebermann, L. Katz, and R. Morey Sorrentino. Preliminary data on attitudes of coaches toward science and technology. Technical report, University of Calgary, Sport Technology Research Lab, Calgary, Canada, 2000. 4

[159] Dario G. Liebermann, Larry Katz, Mike D. Hughes, Roger M. Bartlett, Jim McClements, and Ian M. Franks. Advances in the application of

information technology to sport performance. *Journal of Sports Sciences*, 20(10):755–769, 2002. 2, 4

[160] James J. Little and Z. Gu. Video retrieval by spatial and temporal structure of trajectories. *SPIE Storage and Retrieval for Media Databases*, 2001. 156

[161] GuoJun Liu, XiangLong Tang, Da Sun, and JianHua Huang. Robust registration of long sport video sequence. In *Proc. of Int. Conf. on Computer Vision Systems (ICVS)*, 2007. 38

[162] Jia Liu, Xiaofeng Tong, Wenlong Li, Tao Wang, Yimin Zhang, Hongqi Wang, Bo Yang, Lifeng Sun, and Shiqiang Yang. Automatic player detection, labeling and tracking in broadcast soccer video. In *Proc. of British Machine Vision Conf. (BMVC)*, 2007. 38, 57

[163] Jun S. Liu and Rong Chen. Sequential monte carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93:1032–1044, 1998. 61

[164] S. P. Lloyd. Least squares quantization in PCM. *IEEE Trans. on Information Theory*, 28:128–137, 1982. 120, 123

[165] David G. Lowe. Object recognition from local scale-invariant features. In *Proc. of Int. Conference on Computer Vision (ICCV)*, September 1999. 119, 173

[166] Ascensio System Ltd. Ascensio system. `http://www.footballsoftpro.com/`. 18

[167] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. of Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 674–679, 1981. 39

[168] Massimo Lucchesi. *Soccer Tactics: An Analysis of Attack and Defense.* Reedswain, September 2000. 155

[169] D. T. Magill. Optimal adaptive estimation of sampled stochastic processes. *IEEE Trans. on Automatic Control*, 10:434–439, October 1965. 52

[170] T. Martinetz and K. Schulten. A Neural Gas Network learns Topologies. In *Proc. of Int. Conf. on Artificial Neural Networks (ICANN)*, pages 397–407. Elsevier, 1991. 120

[171] Thomas Martinetz. Competitive Hebbian Learning Rule Forms Perfectly Topology Preserving Maps. In *Proc. of Int. Conf. on Artificial Neural Networks (ICANN)*, pages 427–434. Springer, 1993. 98

[172] T.M. Martinetz, S.G. Berkovich, and K.J. Schulten. 'Neural-gas' network for vector quantization and its application to time-series prediction. *Neural Networks*, 4(4):558–569, 1993. 98, 158

[173] Thomas Mauthner, Christina Koch, Markus Tilp, and Horst Bischof. Visual tracking of athletes in beach volleyball using a single camera. *International Journal of Computer Science in Sport*, 6(2):21–34, 2007. 19

[174] A. Miah. "New Balls Please": Tennis, Technology, and the Changing Game. In S. Haake and A.O. Coe, editors, *Tennis, Science, and Technology*, pages 285–292. Blackwell Science, London, 2000. 5

[175] Andrea Miene. *Räumlich-zeitliche Analyse von dynamischen Szenen*. PhD thesis, Universität Bremen, July 2003. 157

[176] Thomas B. Moeslund, Adrian Hilton, and Volker Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding (CVIU)*, 104:90–126, 2006. 119

[177] Ranjit Nair, Stacy Marsella, and Taylor Raines. Automated assistants for analyzing team behaviors. *Journal of Autonomous Agents and Multiagent Systems (JAMAAS)*, 2004. 157

[178] Chris J. Needham and Roger D. Boyle. Tracking multiple sports players through occlusion, congestion and scale. In *Proc. of British Machine Vision Conf. (BMVC)*, 2001. 38

[179] Christopher James Needham. *Tracking and Modelling of Team Game Interactions*. PhD thesis, University of Leeds, October 2003. 18, 45, 88, 89

[180] Christopher James Needham and Roger D. Boyle. Performance evaluation metrics and statistics for positional tracker evaluation. In J.L. Crowley, editor, *Proc. of Int. Conv. Vision Systems (ICVS)*, number 2626 in LNCS, pages 278–289. Springer, 2003. 139

[181] S. Oh, S. Russell, and S. Sastry. Markov Chain Monte Carlo Data Association for General Multiple Target Tracking Problems. In *Proc. of IEEE Conf. on Decision and Control*, 2004. 56

[182] Kenji Okuma, Ali Taleghani, Nando De Freitas, James J. Little, and David G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *European Conf. on Computer Vision (ECCV)*, pages 28–39, 2004. 56, 120, 141

[183] Orad. Trackvision. `http://www.orad.tv/en/page.asp?id=85`. 18

[184] Henry Orenstein and James J. Maune. Apparatus for detecting moving ball. U.S. Patent 5976038, November 1999. 34

[185] Donovan H. Parks and Sidney S. Fels. Evaluation of background subtraction algorithms with post-processing. In *IEEE Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, pages 192–199, 2008. 37

[186] Donald J. Patterson, Lin Liao, Dieter Fox, and Henry Kautz. Inferring high-level behavior from low-level sensors. In *Proc. of Conf. on Ubiquitous Computing*, 2003. 72

[187] Ofir Pele and Michael Werman. Fast and robust earth mover's distances. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, 2009. X, 124

[188] J. Pers and S. Kovacic. Tracking people in sport: Making use of partially controlled environment. In *Computer Analysis of Images and Patterns*, LNCS 2124, pages 374–382, 2001. 19

[189] M. Perse, M. Kristan, S. Kovacic, G. Vuckovic, and J. Pers. A trajectory-based analysis of coordinated team activity in a basketball game. *Computer Vision and Image Understanding (CVIU)*, 113(5):612–621, May 2009. 158

[190] Massimo Piccardi. Background subtraction techniques: a review. In *Proc. of Int. Conf. on Systems, Man and Cybernetics*, pages 3099–3104. IEEE, 2004. 37

[191] Gopal Pingali, Yves Jean, and Ingrid Carlbom. Real-Time Tracking for Enhanced Tennis Broadcasts. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 260–265, 1998. 18

[192] Gopal Pingali, Agata Opalach, and Yves Jean. Ball tracking and virtual replays for innovative tennis broadcasts. In *Proc. of Int. Conf. on Pattern Recognition (ICPR)*, page 4152, Washington, DC, USA, 2000. IEEE Computer Society. 18

[193] Gopal Pingali, Agata Opalach, Yves Jean, and Ingrid Carlbom. Visualization of sports using motion trajectories: Providing insights into performance, style, and strategy. In *Proc. of 12th Annual IEEE Visualization Conf. (Vis 2001)*, pages 75–82, 2001. 18, 19, 20, 156

[194] Jim Puhalla, Jeff Krans, and Mike Goatley. *Sports fields: a manual for design, construction, and maintenance.* John Wiley & Sons, 1999. 140

[195] E. Punskaya, A. Doucet, and W. J. Fitzgerald. On the use and misuse of particle filtering in digital communications. In *Proc. of European Signal Processing Conf. (EUSIPCO)*, 2002. 62

[196] J. R. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., 1993. 166

[197] Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–286, February 1989. X, 52, 136, 158, 162

[198] Deva Ramanan, David A. Forsyth, and Andrew Zisserman. Tracking people by learning their appearance. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 29(1):65–81, 2007. 120

[199] Fernando Ramos and Huberto Ayanegui. Discovering tactical behavior patterns supported by topological structures in soccer agent domains. In Lin Padgham, David C. Parkes, Jörg Müller, and Simon Parsons, editors, *Proc. of Int. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, volume 3, pages 1421–1424. IFAAMAS, 2008. 88, 157

[200] Christopher Rasmussen and Gregory D. Hager. Probabilistic data association methods for tracking complex visual objects. *Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 23(6):560–576, June 2001. 55

[201] Gary Ravet. System and method for tracking the movement and location of an object in a predefined area. U.S. Patent 7091863, August 2006. 34

[202] Carolyn Ray. *Identity And Universals: A Conceptualist Approach to Logical, Metaphysical, and Epistemological Problems of Contemporary Identity Theory*. PhD thesis, Indiana University, November 1998. 2

[203] D. B. Reid. A multiple hypothesis filter for tracking multiple targets in a cluttered environment. Technical report, Lockheed Palo Alto Research Lab, Palo Alto, CA, September 1977. 55

[204] D. B. Reid. An algorithm for tracking multiple targets. *IEEE Trans. on Automatic Control*, 24:843–854, December 1979. 55

[205] J.R. Renno, J. Orwell, D. Thirde, and G.A. Jones. Shadow classification and evaluation for soccer player detection. In *Proc. of British Machine Vision Conf. (BMVC)*, 2004. 38

[206] Myung-Cheol Roh, Bill Christmas, Joseph Kittler, and Seong-Whan Lee. Robust player gesture spotting and recognition in low-resolution sports

video. In *Proc. European Conf. on Computer Vision (ECCV 2006)*, 2006. 38, 136

[207] Myung-Cheol Roh, Bill Christmas, Joseph Kittler, and Seong-Whan Lee. Gesture spotting for low-resolution sports video annotation. *Pattern Recognition*, 41(3):1124–1137, 2008. 136

[208] Sam Roweis and Lawrence Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000. 120

[209] Y. Rubner, J. Puzicha, C. Tomasi, and J. M. Buhmann. Empirical evaluation of dissimilarity measures for color and texture. *Computer Vision and Image Understanding (CVIU)*, 84:25–43, October 2001. 124, 125

[210] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. A metric for distributions with applications to image databases. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, pages 59–66, January 1998. X, 124

[211] Stuart Russell and Peter Norvig. *Artificial Intelligence – A Modern Approach*. Prentice Hall, Upper Saddle River, New Jersey, 2003. 7, 30, 92, 121

[212] Simo Särkkä, Aki Vehtari, and Jouko Lampinen. Rao-Blackwellized Monte Carlo data association for multiple target tracking. In *Proc. of Int. Conf. on Information Fusion*, volume 7, pages 583–590, Stockholm, June 2004. 74

[213] Simo Särkkä, Aki Vehtari, and Jouko Lampinen. Rao-blackwellized particle filter for multiple target tracking. *Information Fusion Journal*, 8(1):2–15, 2007. X, 11, 56, 60, 61, 74

[214] P. Scheunders. A comparison of clustering algorithms applied to color image quantization. *Pattern Recognition Letters*, 18:1379–1384, 1997. 123

[215] Arno Schödl, Richard Szeliski, David H. Salesin, and Irfan Essa. Video textures. In *Proc. of Int. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 489–498, 2000. 137

[216] Francesca Schuler, Krishna Jonnalagadda, Xun Luo, Irfan Nasir, and Kaidi Zhao. Method and apparatus for tracking sports play. U.S. Patent Application 2009/0111582, April 2009. 34

[217] Dirk Schulz, Wolfram Burgard, Dieter Fox, and Armin B. Cremers. Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. In *RoboCup*, 2001. 56, 141

[218] Kyle Schurman. Technology Super Bowl: Football Concussions vs. Simbex's HIT System. *Hard Hat Area*, 8(1):52–55, January 2008. 34

[219] Daniel Setterwall. Computerised video analysis of football - technical and commercial possibilities for football coaching. Master's thesis, KTH Stockholm, 2003. 18

[220] Mehmet Sezgin and Bülent Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13(1):146–165, January 2004. 37

[221] G. Sharma and H.J. Trussell. Digital color imaging. *IEEE Trans. on Image Processing*, 6(7):901–932, 1997. 119

[222] Eli Shechtman and Michal Irani. Matching local self-similarities across images and videos. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2007. 136

[223] R.A. Singer, R.G. Sea, and K. Housewright. Derivation and evaluation of improved tracking filters for use in dense multitarget environments. *IEEE Trans. on Information Theory*, 20:423–432, July 1974. 52

[224] R. W. Sittler. An optimal data association problem in surveillance theory. *IEEE Trans. on Military Electronics*, 8:125–139, April 1964. 54

[225] Xuan Song, Jinshi Cui, Hongbin Zha, and Huijing Zhao. Probabilistic detection-based particle filter for multi-target tracking. In R. Fraile M. Everingham, C.J. Needham, editor, *British Machine Vision Conf. (BMVC)*, volume 1, pages 223–232, Leeds, Sept. 2008. 141

[226] Xuan Song, Jinshi Cui, Hongbin Zha, and Huijing Zhao. Vision-Based Multiple Interacting Targets Tracking Via On-Line Supervised Learning. In D. Forsyth, P. Torr, and A. Zissermann, editors, *Proc. of European Conf. on Computer Vision (ECCV)*, LNCS 5304, pages 642–655. Springer, October 2008. 30, 56, 119, 122, 141

[227] P. Spagnolo, N. Mosca, M. Nitti, and A. Distante. An unsupervised approach for segmentation and clustering of soccer players. In *Proc. of Int. Machine Vision and Image Processing Conf. (IMVIP)*, 2007. 38

[228] BBC Sports. Hawk- eye. `http://news.bbc.co.uk/sport1/hi/tennis/2977068.stm`, 2003. 18

[229] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *Proc. of Conf. on Computer Vision and Pattern Recognition (CVPR)*, page 252, 1999. 37

[230] R. L. Streit. The PMHT and related applications of mixture densities. In *Proc. of Int. Conf. on Information Fusion (FUSION)*, Florence, Italy, July 2006. 55

[231] R. L. Streit and T. E. Luginbuhl. Probabilistic multi-hypothesis tracking. Technical report, Naval Undersea Warefare Center, Newport, RI, USA, February 1995. 55

[232] Marc Strickert and Barbara Hammer. Merge SOM for temporal data. *Neurocomputing*, 64:39–71, 2005. X, 11, 158, 159, 162, 167

[233] Josephine Sullivan and Stefan Carlsson. Recognizing and tracking human action. In *Proc. of European Conf. on Computer Vision (ECCV)*, pages 629–644, May 2002. 136

[234] Tsuyoshi Taki and Jun ichi Hasegawa. Visualization of dominant region in team games and its application to teamwork analysis. In *IEEE Computer Graphics International (CGI)*, page 227, 2000. 37, 156

[235] Tsuyoshi Taki, Jun ichi Hasegawa, and Teruo Fukumura. Development of motion analysis system for quantitative evaluation of teamwork in soccer games. In *Proc. of the International Conf. on Image Processing (ICIP)*, pages 815–818, 1996. 156

[236] Qing Tang, Irena Koprinska, and Jesse S. Jin. Content-adaptive transmission of reconstructed soccer goal events over low bandwidth networks. In *ACM Multimedia*, pages 271–274, 2005. 6

[237] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, 2000. 120

[238] Moritz Tenorth and Michael Beetz. Towards practical and grounded knowledge representation systems for autonomous household robots. In *Proc. of Int. Workshop on Cognition for Technical Systems*, Munich, Germany, October 2008. 167

[239] Daniel Theweleit. Drei streifen und ein guter freund. *RUND - Das Fussballmagazin*, pages 38–39, September 2005. 34

[240] G.A. Thomas. Real-time camera pose estimation for augmenting sports scenes. In *Proc. of European Conf. on Visual Media Production (CVMP)*, London, nov 2006. 37

[241] TRACAB. Tracab image tracking system. `http://www.tracab.com/`. 18

[242] Roger Y. Tsai. A versatile camera calibration techniaue for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *Journal of Robotics and Automation*, 3(4):323–344, August 1987. 36

[243] Metthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991. 120

[244] Unknown. Verzögerung beim Chipball. Süddeutsche Zeitung AG, June 2005. 34

[245] P. E. Utgoff, N. C. Berkman, and J. A Clouse. Decision tree induction based on efficient tree restructuring. *Machine Learning*, 29:5–44, 1997. 121

[246] Nicolai v. Hoyningen-Huene, Bernhard Kirchlechner, and Michael Beetz. GrAM: Reasoning with Grounded Action Models by Combining Knowledge Representation and Data Mining. In *Towards Affordance-based Robot Control*, 2007. 7

[247] Nicolas Vandenbroucke, Ludovic Macaire, and Jack-Gerard Postaire. Color image segmentation by pixel classification in an adapted hybrid color space. application to soccer image analysis. *Computer Vision and Image Understanding (CVIU)*, 90:190–216, may 2003. 38, 119

[248] J. Vermaak, A. Doucet, and P. Pérez. Maintaining multi-modality through mixture tracking. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, pages 1110–1116, 2003. 56

[249] U. Visser and H.-G. Weland. Using online learning to analyze the opponent's behavior. In *RoboCup 2002*, Lecture Notes in Artificial Intelligence. Springer Verlag, 2002. 157

[250] Ubbo Visser, Christian Drücker, Sebastian Hübner, Esko Schmidt, and Hans-Georg Weland. Recognizing formations in opponent teams. In P. Stone, T. Balch, and G. Kraetzschmar, editors, *RoboCup 2000: Robot Soccer World Cup IV*, volume 2019 of *Lecture Notes in Computer Science*, pages 391–396. Springer Berlin / Heidelberg, 2001. 88, 157

[251] Thomas Voegtlin. Recursive Self-Organizing Maps. *Neural Networks*, 15(8-9), 2002. 158, 162

[252] Nicolai von Hoyningen-Huene and Michael Beetz. Rao-Blackwellized Resampling Particle Filter for Real-Time Player Tracking in Sports. In AlpeshKumar Ranchordas and Helder Araujo, editors, *Int. Conf. on Computer Vision Theory and Applications (VISAPP)*, volume 1, pages 464–470, Lisboa, Portugal, Feb. 2009. INSTICC, INSTICC press. 19

[253] Kongwah Wan, Xin Yan, Xinguo Yu, and Changsheng Xu. Robust goal-mouth detection for virtual content insertion. In *Proc. of ACM Int. Conf. on Multimedia (MULTIMEDIA)*, pages 468–469, New York, NY, USA, 2003. ACM. 37

[254] J. R. Wang and N. Parameswaran. Survey of sports video analysis: research issues and applications. In *Proc. of Pan-Sydney area workshop on Visual information processing (VIP)*, pages 87–90, Darlinghurst, Australia, Australia, 2004. Australian Computer Society, Inc. 8

[255] Liang Wang, Weiming Hu, and Tieniu Tan. Recent developments in human motion analysis. *Pattern Recognition*, 36(3):585–601, 2003. 119

[256] Liang Wang and David Suter. Informative shape representations for human action recognition. In *Proc. of Int. Conf. on Pattern Recognition (ICPR)*, 2006. 119

[257] Liang Wang and David Suter. Visual learning and recognition of sequential data manifolds with applications to human movement analysis. *Computer Vision and Image Understanding (CVIU)*, 2007. 118, 119

[258] Tomoki Watanabe, Miki Haseyama, and Hideo Kitajima. A soccer field tracking method with wire frame model from tv images. In *Int. Conf. on Image Processing (ICIP)*, pages 1633–1636, 2004. 37

[259] Daniel Weinland, Remi Ronfard, and Edmond Boyer. Free viewpoint action recognition using motion history volumes. *Computer Vision and Image Understanding (CVIU)*, 104:249–257, 2006. 136

[260] Greg Welch and Gary Bishop. An Introduction to the Kalman Filter. In *Proc. of SIGGRAPH*, Los Angeles, CA, August 2001. ACM. 29, 51

[261] J. Wesson. *The science of soccer.* Institute of Physics, 2002. 155

[262] Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques.* Morgan Kaufmann, San Francisco, 2 edition, 2005. 166

[263] M. Wünstel, D. Polani, T. Uthmann, and J. Perl. Behavior classification with self-organizing maps. In P. Stone, T. Balch, and G. Kraetzschmar, editors, *RoboCup 2000: Robot Soccer. World Cup IV*, volume 2019 of *Lecture Notes in Artificial Intelligence*, pages 108–118. Springer Berlin / Heidelberg, 2002. 156

[264] Stephan Würmlin, Edouard Lamboray, Michael Waschbüsch, Peter Kaufmann, Aljoscha Smolic, and Markus Gross. Image-space free-viewpoint video. In *Proc. of Vision, Modeling, Visualization*, pages 453–460, 2005. 5, 18

[265] Changsheng Xu, Jian Cheng, Yi Zhang, Yifan Zhang, and Hanqing Lu. Sports video analysis: Semantics extraction, editorial content creation and adaptation. *Journal of Multimedia*, 4(2):69–78, April 2009. 6, 44, 172, 174

[266] Huaxin Xu and Tat-Seng Chua. The fusion of audio-visual features and external knowledge for event detection in team sports video. In *Multimedia Information Retrieval*, pages 127–134, 2004. 20, 44

[267] Huaxin Xu and Tat-Seng Chua. Fusion of av features and external information sources for event detection in team sports video. *ACM Transactions on Multimedia Computing, Communications and Applications (TOMCCAP)*, 2(1):44–67, 2006. 44

[268] Ming Xu, Liam Lowey, and James Orwell. Architecture and algorithms for tracking football players with multiple cameras. In *IEEE Intelligent Distributed Surveillance Systems (IDSS)*, pages 2909–2912, 2004. 19, 36, 55

[269] Ming Xu, James Orwell, and Graeme A. Jones. Tracking football players with multiple cameras. In *Proc. of the International Conf. on Image Processing (ICIP)*, pages 2909–2912, 2004. 19

[270] Ming Xu, James Orwell, Liam Lowey, and David Thirde. Architecture and algorithms for tracking football players with multiple cameras. In *IEEE Proc. of Visual Image Signal Processing*, volume 152, pages 232–241, 2005. 19, 141

[271] Fei Yan, William Christmas, and Josef Kittler. A maximum a posteriori probability viterbi data association algorithm for ball tracking in sports video. In *Proc. of Int. Conf. on Pattern Recognition (ICPR)*, 2006. 55

[272] Jun Yang, Ming yu Chen, and Alex Hauptmann. Finding Person X: Correlating Names with Visual Appearances. In *Proc. of Int. Conf. on Image and Video Retrieval (CIVR)*, July 2004. 118

[273] Qixiang Ye, Qingming Huang, Shuqiang Jiang, Yang Liu, and Wen Gao. Jersey number detection in sports video for athlete identification. In *Proc. of Visual Communications and Image Processing (SPIE)*, volume 5960, 2005. 90, 118

[274] Ho-Sub Yoon, Young lae J. Bae, and Young kyu Yang. A soccer image sequence mosaicking and analysis method using line and advertisement board detection. *ETRI Journal*, 24(6):443–454, December 2002. 37, 38

[275] Xinguo Yu and Dirk Farin. Current and emerging topics in sports video processing. In *IEEE International Conf. on Multimedia and Expo (ICME)*, 2005. 8

[276] ZGDV, Fraunhofer IGD, and GIStec. servingo – Immer auf Ballhöhe. press release, February 2005. 5

[277] Guangyu Zhu, Changsheng Xu, and Qingming Huang. Sports video analysis: From semantics to tactics. In A. Divakaran, editor, *Multimedia Content Analysis, Signals and Communication Technology*, page 295. Springer, 2009. 8

[278] Guangyu Zhu, Changsheng Xu, Qingming Huang, Yong Rui, Shuqiang Jiang, Wen Gao, and Hongxun Yao. Event tactic analysis based on broadcast sports video. *IEEE Trans. on Multimedia*, 11(1):49–67, January 2009. 44