

Interactive and Quantitative Knowledge-Discovery in Large-Scale 3D Tracking Data

Zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
der KIT-Fakultät für Maschinenbau
Karlsruher Institut für Technologie (KIT)

genehmigte

Dissertation

von

Benjamin Schott, M.Sc.

geboren am 09. Januar 1989 in Aalen

Hauptreferent:	apl. Prof. Dr.-Ing. Ralf Mikut
Korreferenten:	Prof. Dr.-Ing. Veit Hagenmeyer, Jun.-Prof. Dr.-Ing. Johannes Stegmaier

Tag der mündlichen Prüfung: 17.12.2018



This document is licensed under the Creative Commons Attribution – Share Alike 3.0 DE License (CC BY-SA 3.0 DE): <http://creativecommons.org/licenses/by-sa/3.0/de/>

Zusammenfassung

Neueste Entwicklungen im Bereich der Tracking-Technologien bieten in Kombination mit verbesserten Speicherinfrastrukturen die Möglichkeit, große Mengen an hochauflösenden Objektlokalisierungsdaten in Form von Trajektorien zu generieren und zu speichern. Aufgrund der enormen Größe und Komplexität der erzeugten Trajektorien-Datensätze ist die systematische Analyse der Daten und die systematische Extraktion des relevanten Wissens ein herausforderndes Problem. Darüber hinaus wird vorhandenes *a priori* Wissen oftmals nur unzureichend in der Analyse von Trajektorien-Datensätzen eingebettet. Gründe hierfür sind Limitationen in bestehenden Algorithmen und Software-Frameworks. Durch die Nichtbeachtung des *a priori* Wissens geht jedoch ein bedeutender Teil an Zusatzinformation für die Analyse verloren. Des Weiteren sind bestehende Tracking-Algorithmen bei Objektdaten mit hoher Dichte und enthaltenem Rauschen nicht in der Lage, fehlerfreie Trajektorien zu erzeugen. Dies führt dazu, dass automatisierte Algorithmen für eine detaillierte Analyse mit fragmentierten Trajektorien umgehen müssen.

Die wesentlichen Beiträge der vorliegenden Arbeit sind ein neues Konzept zur systematischen Integration von *a priori* Wissen in den Analyse-Prozess von großen 3D+t-Trajektorien Datensätzen mit automatisierten Algorithmen zur Trajektorien-Analyse und interaktiven Datenhandhabungsstrategien. Des Weiteren wurde ein neues Konzept entwickelt, um fragmentierte Tracking-Daten in den Analyse-Prozess zu integrieren. Basierend auf dem neu entwickelten Visualisierungsframework zur interaktiven Handhabung von 3D+t-Tracking-Daten wird ein neuer Prozess zum Transfer von kompletten Trajektorien-Analysepipelines auf neue Datensätze vorgestellt. Die Funktionalität der entwickelten Methoden wurde auf neu entwickelten künstlichen Benchmark-Datensätzen validiert, um eine Vielzahl von auftretenden Trajektorien-Datensatzcharakteristiken systematisch abzudecken. Alle entwickelten Methoden wurden als plattformunabhängige Open-Source-

Software zur freien Verfügung bereitgestellt. Eine Vielzahl von Tracking-Anwendungsproblemen im Bereich der 3D-Lichtscheibenmikroskopie wurde erfolgreich mit Hilfe der in dieser Arbeit entwickelten Methoden gelöst. So wurde die entwickelte Methodik beispielsweise erfolgreich genutzt, um Zellgruppen in Zebrafärbungen während frühen Entwicklungsstadien zu separieren, quantifizieren und zwischen verschiedenen Embryos zu vergleichen, um unterlagerte biologische Effekte quantitativ beschreiben zu können.

Abstract

New developments in tracking technologies combined with enhanced data storage techniques provide powerful ways to collect a tremendous amount of highly resolved object localization data represented as trajectories. Due to the enormous size and complexity of the routinely produced datasets, the systematic analysis and the extraction of relevant knowledge out of the data is a challenging problem. Furthermore, available prior knowledge could be utilized for sophisticated analysis can in many cases not be sufficiently considered for such complex datasets, due to the technical limitations of existing analysis approaches and software tools. Moreover, existing state-of-the-art tracking algorithms are not able to create error-free tracks in the presence of highly dense and noisy data measurements leading to crucial problems coping with the fragmented tracking data.

The major contributions of the present thesis are a new concept to systematically incorporate prior knowledge in the knowledge discovery process of large-scale tracking data combining interactive visual exploration and automated trajectory analysis methods. In addition, a new approach to incorporate fragmented tracking data in the analysis of large-scale 3D+t tracking data was developed. Based on the new visualization framework for the interactive handling of 3D+t tracking data, a new approach is presented to transfer complete analysis pipelines to similar datasets. The overall variety of developed methods was consistently validated on newly developed simulated benchmarks allowing to quantitatively investigate the applicability to different tracking databases. In addition, all developed methods were implemented as a platform independent open-source software framework to make the algorithms accessible to the community. Moreover, the successful application of the newly developed methods to large-scale 3D+t tracking datasets of fluorescent

light-sheet microscopy images was shown. In particular, the proposed framework was used to separate, quantify and compare cell groups in whole embryo data sets during gastrulation events in early zebrafish development.

Acknowledgments

First and foremost, I would like to express my gratitude to my direct supervisor apl. Prof. Dr.-Ing. Ralf Mikut for his guidance, countless constructive discussions, and his encouragement on both my research and career path. I would also like to thank Prof. Dr.-Ing. Veit Hagenmeyer and Prof. Dr.-Ing. Georg Bretthauer a lot for the great opportunity to spend my time as a PhD student at the Institute for Automation and Applied Computer Science (IAI) at the Karlsruhe Institute of Technology (KIT) and for the enormous support far beyond the doctoral thesis. I deeply appreciate having had the opportunity to work freely and responsibly on highly fascinating projects. Special thanks to Jun.-Prof. Dr.-Ing. Johannes Stegmaier for the indescribably valuable discussions, suggestions and the reviewing of the thesis. It was an honor and a great pleasure to work with him and to be co-advised by him.

I would also like to express my grateful thanks to all colleagues at our working group, especially to Andreas Bartschat, Simon Waczowicz, Wolfgang Doneit, Jorge Ángel González Ordiano, Nicole Ludwig, Marian Turowski, Moritz Böhland, Tim Scherr and Michele René Tuga for the pleasant, cooperative and familiar working atmosphere. I am very grateful to each of them for the time spent together.

Many thanks to all collaboration partners from the Institute of Toxicology and Genetics (ITG) and the Institute of Applied Physics (APH) for constructive and successful collaborative work during my time at the KIT. Namely, I would like to express my thanks to Dr. Masanari Takamiya, Prof. Dr. Uwe Strähle, Dr. Andrei Kobitski and Prof. Dr. G. Ulrich Nienhaus for all the joint projects I was allowed to contribute.

I am especially grateful for all the valuable contributions of my past Bachelor and Master students that I had the pleasure to working with. I would espe-

cially like to mention Manual Traub, Denis Blessing, Katharina Löffler, Cornelia Schlagenhauf and Thomas Antritter.

For the constructive discussions, ongoing support and the guidance during my thesis I want to thank my thesis advisory committee, namely apl. Prof. Dr.-Ing. Ralf Mikut, PD Dr.-Ing. Markus Reischl, Dr. Thomas Dickmeis and Prof. Dr. Nicholas Foulkes. For the opportunity to attend the exiting and advanced training courses I want to thank the BioInterfaces International Graduate School (BIF-IGS).

I would like to express my heartfelt thanks to all of the colleagues at the IAI and the KIT far beyond our working group for the cohesiveness in the last years and resulting deep friendships. On behalf of everyone I would like to mention Claudia, Achim, Robert and Kersten.

Writing my dissertation was hardly possible without my family support. Words cannot express how grateful I am to my parents Hildegard and Reinhold and my sisters Franziska and Sarah for all their emotional support! Last but certainly not least, I would like to express my heartfelt appreciation and deepest gratitude to Christina for her endless support and understanding throughout the last years.

Contents

Zusammenfassung	i
Abstract	iii
Acknowledgments	v
1 Introduction	1
1.1 Motivation	1
1.2 Theoretical Background and Related Work	3
1.2.1 Generation of Trajectory Databases	3
Trajectory Representation	3
Direct Measure Techniques	3
Image-Based Techniques	5
Reasons for Fragmented Tracks	6
1.2.2 Application-Specific Solutions - Light-Sheet Microscopy	6
Image Acquisition	7
Image Preprocessing	8
Registration and Fusion	9
Segmentation	9
Tracking	9
1.2.3 Knowledge Discovery in Trajectory Databases	10
Importance of Knowledge Discovery	10
Knowledge Discovery in General	11
Extract Information in Trajectory Databases	11
Integrating Prior Knowledge	17
Handling Fragmented Tracks in the Analysis	18
Visual Explorative Data Analysis	19

1.2.4	Available Software Solutions	20
1.3	Open Questions	22
1.4	Objectives and Thesis Outline	23
2	Synthetic Benchmark Generation for Quality-Dependent Trajectory Databases	27
2.1	Systematization of Problem Classes	28
2.2	Reasons for Fragmented Tracks	29
2.3	Generate Quality-Dependent Trajectory Benchmark Databases	31
2.3.1	Fulltrack Generation Module	33
Existing Databases	33	
Purely Synthetic	34	
Simulated on Real Data	34	
2.3.2	Quality Reduction Module	35
Markov-Based Quality Reduction Module	35	
Bunch-Based Quality Reduction Module	37	
Object Division Module	40	
2.4	Systematic Generation of Validation Benchmarks	42
2.4.1	General Scheme	42
2.4.2	Benchmarks Used in this thesis	45
3	Integrating Prior Knowledge in the Knowledge Discovery of Trajectory Databases	47
3.1	General Design of the Knowledge Discovery Process	48
3.1.1	Information Allocation Process	51
3.1.2	Interactive Modifications	52
3.2	Systematization of Prior Knowledge	52
3.2.1	Systematization of Different Sources of Prior Knowledge	53
3.2.2	Efficient Incorporation of New Prior Knowledge	53

3.3	Conceptual Approaches to Analyze Large-Scale 3D+t Trajectory Databases	56
3.3.1	Overview of Conceptual Approaches	56
3.3.2	Handling Fragmented Object Tracking Data	57
	Possibilities to Handle Fragmented Trajectory Data	57
	Calculate Trajectory Neighborhood	62
	Select Representative Trajectories	65
	Allocation of Fragmented Trajectories	67
3.3.3	Quantitative Description of Object Movement Behavior	79
3.3.4	Focusing on Spatio-Temporal Regions	82
3.3.5	Feature-Based Extraction of Groups of Trajectories	86
3.3.6	Transfer of Analysis Pipelines to New Trajectory Databases	89
3.3.7	Comparison of Large-Scale Trajectory Databases	92
3.3.8	Quantitative Lineage Analysis	93
3.3.9	Quantitative Neighborhood Assumptions	95
3.4	Discussion	97
4	Enhancing Interactivity Through Prior Knowledge Integration	99
4.1	Visual Analysis of Object Migration Patterns	100
4.1.1	3D Rendering of Large-Scale Object Trajectory Data	100
4.1.2	Maximum Intensity Projection Overlay	102
4.1.3	3D+t Visualization of Migrating Objects	104
4.1.4	Scatter Plot for Track-Based Features	104
4.1.5	Lineage Visualization for Dividing Objects	107
4.1.6	Modular Incorporation of New Visualizations	108
4.2	Virtual Dissection of Large-Scale Object Tracking Data	108
4.2.1	Process of Interactive Selection Strategies	109

4.2.2	Online Propagation in Trajectory Visualizations	110
4.3	Interactive Hierarchical Analysis Process	112
4.3.1	General Scheme for the Hierarchical Analysis Process	112
4.3.2	Visual Realization of the Hierarchical Analysis Process	113
4.3.3	Interactive Hierarchical Analysis Transfer	113
4.4	Interactive Visually-Guided Tracklet Curation	117
4.5	Discussion	118
5	New Implementations and Software Tools	123
5.1	EmbryoMiner: A new Toolbox for SciXMiner	124
5.1.1	Overview	124
5.1.2	Interactive Trajectory Visualization Framework	125
5.1.3	Feature-based Description of Movement Behavior	127
5.1.4	Interactive Manual Curation Framework	128
5.1.5	Interactive Hierarchical Analysis Process	129
5.1.6	Importing Existing Tracking Databases	129
5.2	Comparison to Existing Software Solutions	130
5.3	Discussion	131
6	Study To Evaluate The Knowledge-Discovery Framework	133
6.1	Design of the User Study	134
6.2	Content of the User Study	135
6.3	Results of the User Study	138
6.4	Discussion	142
7	Application	143
7.1	Separation of Hypoblast and Epiblast Cell Groups in Zebrafish Embryos	145

7.1.1	Dataset Description	145
7.1.2	Interactive Trajectory Knowledge Discovery Framework .	146
7.1.3	Results	149
7.1.4	Discussion	152
7.2	Application of the KD-Process to Analyze Neural Crest Cells in Zebrafish Embryos	155
7.2.1	Dataset Description	155
7.2.2	Interactive Trajectory Knowledge Discovery Framework .	156
7.2.3	Results	158
7.2.4	Discussion	160
8	Conclusion	163
A	Nomenclature and Symbols	167
B	Benchmark Databases and Parametrization	173
C	Knowledge Discovery Process	177
D	Interactivity	195
E	Implementation Details	197
F	Evaluation of the Knowledge Discovery Framework	201
	List of Figures	217
	List of Tables	223
	Bibliography	217

1 Introduction

1.1 Motivation

New achievements in tracking technologies combined with enhanced data storing techniques lead to the collection of a tremendous amount of highly resolved object localization data represented as trajectories. The systematic analysis of moving objects trajectories leads to a comprehensive understanding of the movement behavior and dynamics of the underlying objects. Current research in highly actual fields in science, such as airspace monitoring, traffic management, surveillance, and security as well as social behavior studies and developmental biology fundamentally rely on the analysis of such trajectory data [1–3]. There is an inherent problem to efficiently extract relevant knowledge out of the huge amount of trajectory data. Especially, the fact, that state-of-the-art tracking algorithms are not able to create error-free tracks due to tracking abortion in the presence of highly dense objects and noisy data measurement, lead to crucial problems coping with these fragmented tracks. Furthermore, available prior knowledge is in many cases not adequately incorporated in the analysis of highly complex 2D+t and 3D+t trajectory databases are containing objects in the 2D and respectively 3D space over the time. Various approaches to analyze trajectory databases were developed in the past. However, most of the approaches were devised for 2D trajectory databases and they are not able to efficiently handle huge amounts of 3D trajectory databases.

The aim of this thesis was to develop a generic concept for the efficient analysis of huge 2D+t and 3D+t trajectory databases by incorporating prior knowledge using an interactive visual explorative interface in combination with underlying trajectory analysis methods. The introduced concept was successfully applied to trajectory databases to efficiently extract and quantify relevant groups, allowing to gain knowledge about hidden and underlying phenomena within the trajectory data. Furthermore, methods compensating errors within tracking

algorithms, causing fragmented tracks, are developed in order to include such fragmented tracks within the analysis. Therefore, a neighborhood-based assignment is introduced and successfully applied to existing trajectory database containing fragmented tracks. Moreover, an efficient approach to transfer complete trajectory analysis pipelines is developed allowing also the handling of divergence in different trajectory databases. All methods were validated on newly developed artificial benchmarks for the purpose of proofing robustness against a multitude of different trajectory database scenarios. The developed concept provided excellent results in extracting and quantifying information out of a tremendous amount of trajectory databases, by increasing interactivity and improving incorporating prior knowledge. Finally, the introduced concept was successfully used for statistical quantification of trajectory data resulting from multidimensional fluorescent microscopy imaging and tracking. The presented methodology was therefore successfully used to extract different cell groups during gastrulation phase in early developing zebrafish embryos, to quantitatively analyze the extracting groups and successfully transfer the analysis pipelines to new, yet not analyzed, zebrafish embryos.

The next section present related work and point out the context of the thesis, such as the process to integrate prior knowledge in the discovery of large 2D+t and 3D+t object databases and a general knowledge discovery process to analyze highly complex trajectory databases. Furthermore, several interactive visual frameworks are provided to incorporate existing knowledge from experts in an efficient way handling fragmented tracks within tracking databases as well as using multiple representations for an effective refinement of the added prior knowledge. In addition, existing software solutions dealing with the knowledge discovery in 2D+t and 3D+t object databases are introduced. An overview of remaining open questions and general aims, as well as an outline of the major contributions of this thesis to solve the existing problems, is provided at the end of this chapter.

1.2 Theoretical Background and Related Work

1.2.1 Generation of Trajectory Databases

Trajectory Representation

The result of applying image-based or direct tracking techniques to continuous moving objects is a sequence of coordinates indicating the position of each object at each time point [4, 5]. First, the trajectories of the tracked objects have to be reconstructed from the measured coordinates. Therefore practically always linear interpolation is used resulting in piecewise linear trajectories. To yield physically or biologically more precise representation although higher-order interpolation concepts such as cubic splines are used [6].

For a uniform nomenclature in this thesis a trajectory \mathcal{T} is in the following defined as a sequence of discrete points $\mathcal{T} = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{N_P}\}$ with $N_P \in \mathbb{N}$ describing the number of points within each trajectory. Further, each point $\mathbf{p}_i = \{x_i, y_i, t_i\}$ in 2D and respectively $\mathbf{p}_i = \{x_i, y_i, z_i, t_i\}$ in 3D is described through geographic coordinates x_i, y_i, z_i of the moving object at time t_i , assuming a constant frame rate of $\Delta t = \text{constant}$. Moreover, a set of trajectories \mathcal{T}_{set} is described as $\mathcal{T}_{set} = \{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_{N_T}\}$, with $N_T \in \mathbb{N}$ as the number of trajectories within the trajectory database.

Besides the representation of a trajectory through points, also a trajectory can be described using a model [7]. Hereby, a trajectory is modeled as a dynamic process with different outputs at different time points [8, 9]. Commonly used models are Hidden Markov Models (HMM), Probabilistic and Regression Models (PRMs) and Dynamic Bayesian Networks (DBN) [10–15]. Using this model based representation of a trajectory, the similarity between trajectories can be accessed through the similarity between their corresponding models, e.g. using the cross likelihood ratio (CLR) between HMMs [16]. A detailed overview of existing methods for trajectory representation can be found in [17].

Direct Measure Techniques

New data acquisition technology in combination with current developments in mobile sensors and tracking technologies lead to the emergence of a huge

amount of directly captured trajectory databases in a peerless temporal and spatial resolution, which becomes more and more available [18–20].

Such trajectory databases facilitate the description of dynamics in movement behavior of the tracked objects and therefore unveil underlying phenomena and gain information about the underlying objects. The applications range from traffic management [18, 21, 22], air traffic analysis [3, 23, 24], animal behavior studies [25–28], particle dynamics analysis [29–32] and sports play analysis [33–37] up to surveillance and security monitoring and route optimizations tasks [38, 39].

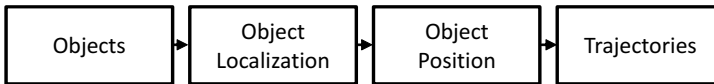


Figure 1.1: General scheme: Direct measurement techniques to create trajectory databases.

In general, the objects are localized using direct measurement techniques, such as Global Positioning System (GPS). The extracted positions for each time point form the basis for the trajectories (see the general scheme of direct measurement trajectory databases in Figure 1.1).

The benefit of capturing trajectory databases to gain knowledge about underlying objects characteristic and understand movement behavior gains more and more importance in current science and economy [40]. In GPS tracking, there is a wide range of application reaching from Pedestrian tracking to understand mobility patterns and social interaction [38, 41, 42], cars tracking for traffic management control and route optimization [18, 22], animal tracking to understand behavioral patterns and quantify phenomena in ecology [25, 27, 28] as well as tracking sports player to quantify movement paths and optimize tactical moves [34, 35, 43]. Also, smartphones are used in urban lifestyle and mobility monitoring to capture the behavioral patterns of pedestrians and modeling flows of people in cities for spatial planning and crisis management which is becoming increasingly important in the last decade [44–46]. Further, Bluetooth communication devices are used for real-time vehicle movement tracking in the context of road management and optimization as well as for tracking of humans

in indoor environment, such as shopping malls, to optimize and understand purchasing behavior and plan evacuation concepts [47, 48]. Signals in the ultra short waves (UKW) frequency are further used for tracking of ships and vessels [49, 50] to avoid collisions and analyze and monitor fishing behavior allowing a subsequent analysis of the gathered trajectory data [20, 51].

Image-Based Techniques

In contrast to the direct measurement trajectory databases it is not always possible to track objects with mobile sensors [52, 53]. In these cases, image-based techniques are often used for the tracking of moving objects. In many different application ranging from traffic surveillance and planning [54], nanoparticle tracking in material science up to space debris tracking in spacecraft [52, 55, 56], image-based techniques for tracking are enormously important in science and economy and new areas of application arise more and more [57–60]. In contrast to the direct measurement techniques to create trajectory databases require a module to capture the objects using cameras, microscopes or other optical measurement and recording devices. Once the objects are recorded a subsequent segmentation lead to the extraction of the positions for each time point forming, in the end, the trajectories (see the scheme in Figure 1.2). If there is a continuously visible ID for each object, e.g. vehicle license plate or a specific color code, the image-based techniques are similar to the direct measurement techniques. However, without such a clear ID, several problems arise to track single objects without losing the objects. The wide range of applications in

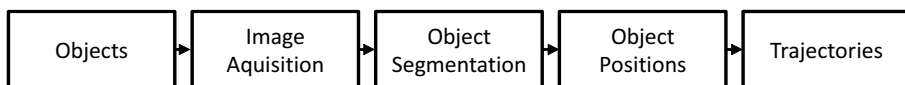


Figure 1.2: General scheme: Indirect measurement techniques to create trajectory databases.

current research reach from tracking in material science [61], tracking in traffic control and surveillance [54, 62, 63], tracking of nanoparticles in biological applications [55, 64], people tracking in video surveillance and customer behavior mining as well as ship tracking using earth observation satellites [52] up to tracking applications in health and medicine [53, 65] as well as particle

tracking in various fields [66–70]. This enormous field of applications makes the image-based creation of trajectory databases indispensable [71].

Reasons for Fragmented Tracks

The generation of trajectory databases (Section 1.2.1) not always lead to error free tracks over the complete acquisition time interval. Sensor failure or instability in data connections for example lead to fragmented tracks in direct measurements in trajectory acquisition techniques [72–74]. If the identity of the objects is recorded, the linking of the fragmented tracks becomes easier. Without the recording of the object identity the linking problem becomes difficult. In image-based tracking techniques, traditional pipelines can hardly perform precise object detection due to segmentation errors and occlusion, as the density of objects increase or the image quality is reduced by a noisy environment [75–77]. Therefore, it is not always possible to keep the object identity over the complete acquisition interval leading to fragmented tracks [78–80].

Without having error free tracks, it is impossible to quantitatively describe trajectories according to global characteristics and make object-specific assumptions on a very detailed single object level leading to an enormous challenge in analyzing and handling such fragmented trajectory datasets [81–83].

1.2.2 Application-Specific Solutions - Light-Sheet Microscopy

The workflow (Figure 1.3) to create a trajectory database out of light-sheet microscopy images is illustrated in Figure 1.4. Here, the microscopy images lead to great challenges through a highly fragmentation of the tracks due to enormous density and high noise ratios within the data. The single steps to extract trajectories out of raw images are described in more detail in the following [84].

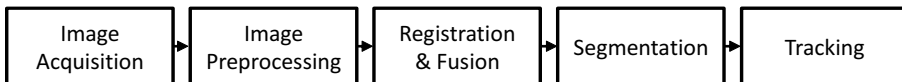


Figure 1.3: Pipeline to process light sheet microscopy images.

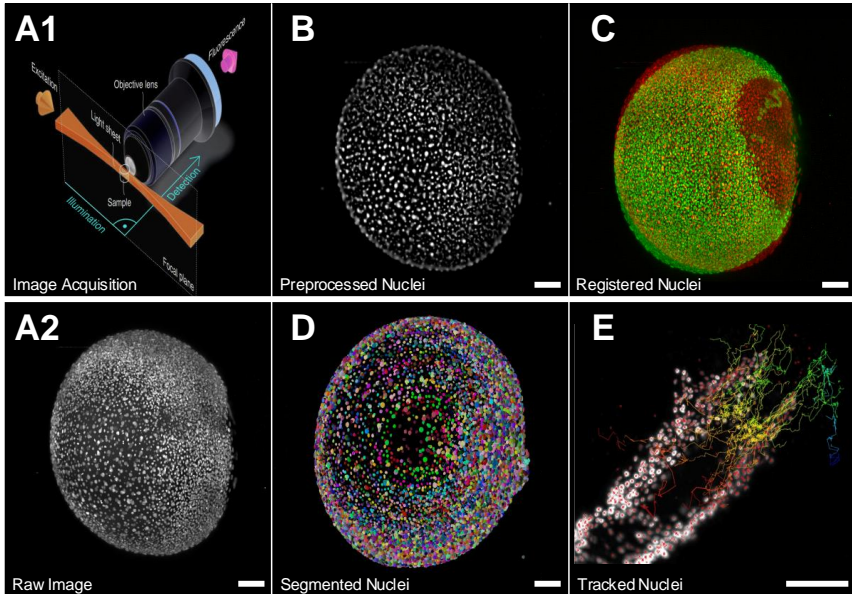


Figure 1.4: Representation of single steps within a pipeline to process light sheet microscopy images. (A1) Fundamental principle of light-sheet fluorescent microscopy (reproduced from [85]), (A2) Snapshot of an early developmental stage in zebrafish embryo, (B) Preprocessed nuclei using noise filtering, (C) Registered views of two zebrafish embryos, (D) Segmentation of the nuclei, (E) Tracked nuclei on the basis of the segmentation results [84]. Scale bar $100 \mu\text{m}$.

Image Acquisition

In microscopy-based setups, the objects have to be recorded using one of the common microscopy approaches, such as confocal, light electron, light-sheet or two photon microscopy [86–90]. In light-sheet microscopy, an optical sectioning is performed on the probe to obtain a 3D image stack in high resolution [91]. A focused laser light-sheet hereby illuminates areas of a few micrometers at a time and the emitted fluorescence is captured by orthogonal detectors [92–94]. Here, the laser light-sheet expand with the depth in the probe and is additionally reflected to a small extend, leading to losses in image quality [95]. A 3D image is hereby generated by sequentially scanning the laser light-sheet through the probe [96–98]. Furthermore, the high imaging speed allows to cap-

ture time-series of 3D images and therefore is able to monitor dynamic objects [99–101]. For imaging biological cell dynamics, the fluorescence has to be inserted mostly in the cell nuclei by marker proteins such as green fluorescent proteins (GFP). In contrast, inserting the fluorescent proteins within the cell plasma leads to a worse distinguishability of the cells due to the less specificity of the plasma in comparison with the cell nuclei.

The fluorescence within the cell nuclei is then emitted based on the illumination at a given excitation wavelength. Using detection optics with a special filter, matching to the spectrum of the fluorophore, the microscope is able to capture cell dynamics within the recorded specimens [102–104]. The signal emitted from the fluorophores is captured by detectors and digitalized for further analysis steps [105–107] yielding to a huge amount of approaches studying gene expression and cell dynamics in whole organisms at a detailed cell level [108, 109].

Image Preprocessing

An appropriate pre-processing of the images can have a notable impact on the quality of subsequent processing steps [110]. Once additional metadata, such as image resolution and acquisition procedure is stored, the image is further modified or information is extracted [111]. Hereby, the images are mostly exonerated from noise, dirt and limited resolution. The noise in the images can be suppressed using different filtering methods, such as mean filtering, Gaussian low-pass filtering or more advanced methods [112]. Further techniques like high-pass filtering, surface fitting or histogram equalization can be used to handle contrast adaption or inhomogeneity due to illumination artifacts [113, 114]. Also, morphological operations can be used to handle imperfect raw image data by applying operations, such as erosion and dilation to separate image regions, smooth structures or correct illumination settings [115–117]. Often, only a specific region of interest is important for further preprocessing. Therefore, the raw image is trimmed by either manual selection, geometrical properties or template matching [118]. The listed image preprocessing steps finally facilitate the application of following processing steps [119].

Registration and Fusion

The multiple views from the 3D light-sheet microscopy have to be spatially aligned to each other to create a fusion 3D image. Due to the acquisition conditions of the microscope, an additionally morphing and scaling of the images could be necessary. In general, the registration of the light-sheet microscopy images can be done based on extracted features (e.g. landmarks) or intensity values. In both registration approaches, the aim is to find an optimal transformation that maximizes a given similarity measure [120, 121]. In this PhD thesis, the registration based on seed points as landmarks is applied. Here the corresponding seed points in the two views are used to calculate the transformation iteratively using least-squares estimations or a random sample consensus approach [122, 123]. Once the transformation is identified, the different images are merged in order to obtain a single 3D image [124].

Segmentation

After the light-sheet images are registered and merged to a 3D image, seed points have to be extracted and the image has to be segmented [125]. The aim of the seed point detection is to obtain precise locations of labeled nuclei of cells to serve as a basis for further analysis steps [126]. Hereby, methods to extract the seed points are image filters that emphasize intensity changes such as Laplacian-of-Gaussian (LoG) and Difference-of-Gaussian (DoG) [122, 127]. The seed points are then extracted using the local extrema of the neighborhood within the filtered image [128]. Based on provided seed points, the segmentation is used to compute a binary image with objects of interest separated from the foreground regions of the image [129–131]. Here, the segmentation algorithms can also be used to extract local information present in the image, such as shape and intensity of the segmented objects [132–134].

Tracking

To analyze the spatio-temporal behavior of single moving cells and investigate the cellular dynamics, tracking algorithms are used [135]. The tracking is directly performed on the results of the preceding segmentation and seed point results. Here, the aim is to determine one-to-one linkings in subsequent video

frames of the captured 3D light-sheet microscopy images [136, 137]. One of the simplest ways to identify linking candidates is to use a nearest neighbor approach by searching for the spatially nearest neighbors [138]. Heuristics and feature matching approaches can thereby improve the quality of tracking [99]. Furthermore, new tracking approaches using global optimal search strategies or non-parametric contour evolution algorithms can be used to track the moving cells [137, 139, 140]. Once the object associations are successfully identified, spatio-temporal object dynamics, as well as lineage trees and fate maps, can be visualized and the cellular dynamics can be analyzed [136, 141].

1.2.3 Knowledge Discovery in Trajectory Databases

Why is Knowledge Discovery Useful

Knowledge discovery in trajectory databases, which can also be found in literature under the term "trajectory data mining", has become enormously important in many areas of science and technology [142]. The traditional way of analyzing trajectory object data is done by human observers by visual inspection. However, new developments in tracking technologies combined with new data acquisition techniques lead to the collection of a tremendous amount of data in a remarkable high resolution [143–145]. Therefore, whole datasets can roughly be analyzed and evaluated manually [146]. Knowledge discovery in trajectory databases is, therefore, the basis for a systematic extraction of a deeper understanding of object behavior and the corresponding underlying processes. The aim is to detect spatio-temporal relationships and systematically characterize movement behavior to gain understanding of the fundamental incident of movement. There are several applications in various domains reaching from surveillance and security tasks, traffic management [1, 39], human movement analysis [147], airspace monitoring [23], social behavior studies [148, 149], urban planning, wildlife protection, location-aware advertising [150, 151], military analysis [152] and sport scene analysis [145] up to developmental biology tasks [153] and the analysis of natural phenomena [5]. The aims hereby are for example, the analysis of climate change and atmospheric phenomena, the optimization of transportation routes, monitoring and simulation of dynamic phe-

nomena [23, 154], as well as the understanding of biological events in model organisms [153, 155] and the improvement of tracking performance [156]. In all these applications, knowledge discovery solutions are required to extract information about the underlying data and gain knowledge about the domain-specific objects characteristics [145].

Knowledge Discovery in General

A systematic scheme to extract the required information is shown in Figure 1.5. Here, first in the data collection step one trajectory database (*TDB*) or a set of *TDBs* with $TDB_{set} = \{TDB_1, TDB_2, \dots, TDB_{N_TDB}\}$, with $N_TDB \in \mathbb{N}$, containing trajectories of moving objects, is generated using domain-specific tracking and storing methods [54, 157]. Afterward, pre-processing steps like trajectory smoothing using wavelet decomposition or other filtering methods, removing corrupted data or adding semantics to the trajectories in form of underlying raw data as images sequences or maps are applied [62, 158].

Once the trajectory databases are pre-processed, the knowledge discovery process is used to extract the required information and therefore answer open questions [159]. Here, the main categories described afterward in more detail, are clustering, trajectory pattern mining as well as anomaly detection [160–162]. As a result of the knowledge discovery, a knowledge databases is generated where the extracted information is summarized and therefore can be used to gain deeper knowledge and insights in the underlying phenomena [5].

Extract Information in Trajectory Databases

Distance and Similarity Measure

The distance and similarity measures for trajectories serve as a basis for comparing trajectories and quantify their differences regarding a given characteristic [163]. These similarity measures form the foundation for trajectory clustering algorithm, anomaly detection, and many further trajectory pattern extraction algorithms [164, 165]. The similarity of trajectories can be calculated based on the similarity of single dimension time series data or incorporating all dimensions (2D, 3D) at the same time [166]. Here, it is necessary to distinguish between spatial, temporal and spatio-temporal similarity depending

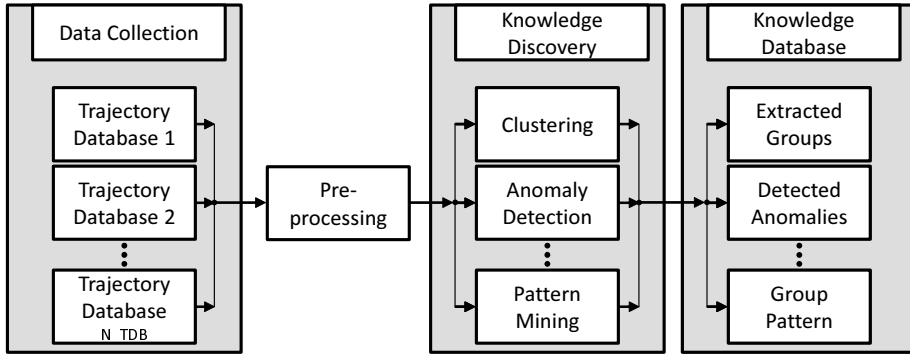


Figure 1.5: General scheme for the knowledge discovery in trajectory databases.

on whether only the pure spatial coordinates, the pure temporal timestamps or both are incorporated [167]. There are in general two main categories of similarity measures. The first category is based on models extracted for each trajectory and the second category is based on the positions of the trajectory. For the model-based approach, each trajectory can be modeled as a dynamic process having different outputs at different times [9, 13]. Here, the distance between trajectories is determined based on the similarity between their corresponding models [12, 14, 15]. Mainly Hidden Markov Models (HMM) and Dynamic Bayesian Network (DBN) models are used [8, 10, 11]. Here, the similarity is measured using a cross likelihood ratio (CLP) for the extracted models [16].

For the position-based approach, there are several distance and similarity measures presented in the literature (see Table 1.1).

The most important are the Euclidean distance [168, 174], the Dynamic Time Warping (DTW) [167, 175, 176] and the Longest Common Subsequence distance (LCSS) [177, 178]. Here, the Euclidean Distance focuses on the similarity of velocities, but having the drawback of trajectories having the same number of points [179–181]. The DTW distance is therefore used to compare unequal length trajectories focusing mainly on the spatial extend by aligning on a common time axis [169, 182, 183]. In contrast, the LCSS distance focus on comparing trajectories with considering a spatial space shifting [170, 184]. Further, com-

Similarity measure	Publication
Euclidean	Yanagisawa 2003[168]
DTW	Antonini 2006 [169]
LCSS	Fashandi 2005 [170]
HD	Morris 2009 [171]
MODHD	Morris 2009 [171]
MSD	Makris 2002 [172]
PCAD	Jiang 2009 [7]
ERP	Chen 2004 [173]

Table 1.1: Similarity measures for trajectory data.

monly used distance measures are the Average Euclidean distance, the PCA-based distance (PCAD) using a transformation to a lower dimensionality [7], the Hausdorff distance (HD), the modified Hausdorff distance (MODHD) [171] as well as Edit distance with Real Penalty (ERP) [173] and the Maximum Separation distance (MSD) [172]. In application tasks, different similarity measures are used regarding the kind of similarity that is investigated.

Trajectory Patterns

Trajectory patterns describe characteristics of a moving object and its corresponding trajectory, either according to the dynamic, the shape or other features and additionally serve as a possibility to address the similarity of trajectories besides the similarity and distance measures described above. Furthermore, these trajectory patterns form the foundation for research to explore and investigate movement dynamics of objects to understand individual behavior and the underlying dynamics. In general, trajectory patterns are used to characterize one single objects and its trajectory isolated or in combination with the neighboring trajectories. Therefore, the literature distinguishes between generic patterns focusing more on single trajectories or behavioral patterns incorporating multiple objects.

The generic patterns (Table 1.2) describe the characteristic of a trajectory either in its pure spatial (e.g. pedestrians that moved the same path no matter at which time), pure temporal (e.g. cars that moved in a given time interval) or in a new combined spatio-temporal way (e.g. together flying airplanes or similar velocity profiles occurring at different time points).

Dimension	Primitive	Primary parameters	Secondary parameters
Spatial	Position (x, y, z)	Distance $f(pos)$	Spatial distribution $f(dist)$
		Spatial extend $f(pos)$	Directional change $f(direc)$
		Direction $f(pos)$	Sinuosity $f(dist)$
		Movement path $f(pos)$	
Temporal	Instance (t)	Time period $f(t)$	Temporal distribution
	Time interval (t)	Duration $f(t)$	Durational change $f(duration)$
Spatio-temporal		Velocity $f(pos, t)$	Approaching rate
		Speed $f(pos, t)$	Acceleration $f(speed)$

Table 1.2: Generic trajectory patterns. Adapted from [185].

In contrast, the group and behavioral patterns describe the interaction of different objects represented by their trajectories [186, 187]. This allows making assumptions about a trajectory in the context of its neighborhood and further reveals phenomena not only based on single objects movement but instead in the relation to a group of objects. The most common group and behavioral patterns are listed in Table 1.3

Clustering

The distance and similarity measures, as well as the trajectory patterns, serve as a basis for trajectory clustering, which is a process of grouping similar objects based on their spatial and temporal characteristics [171, 197]. While the temporal and spatial dimension alone are manageable, the combination of both reveals a number of challenges [150]. Here, the weighting of the single dimension within a distance metric is in most cases application-dependent and has to be adapted according the requirements [198]. Clustering in terms of trajectory data means to discover groups of objects that behave in a similar way in or-

Group patterns	Publication
Flock	Wang 2006 [188], Gudmundsson 2006 [189]
Convoy	Jeung 2008 [190], Kalnis 2005 [191]
Leadership	Andersson 2008 [192], Wu 2014 [193]
Closed Swarm	Li 2010 [194]
Gathering	Zheng 2015 [5]
Meeting	Wachowicz 2011 [195]
Avoidance	Li 2013 [196]
Attraction	Wu 2014 [193]
Convergence	Li 2013 [196]

Table 1.3: Overview of the most common group trajectory patterns.

der to display for example common trends and representative paths or identify anomalies within a dataset [199, 200].

In general, there are three main steps in trajectory clustering: (i) data representation (feature-, raw-data-, model-based), (ii) distance measure and (iii) the grouping rules for the cluster algorithms [201].

Feature-based clustering approaches are in general application-specific [202]. Here, trajectories are reduced to a set of low dimensional features where a given set of features is applicable for a particular application. The similarities are calculated in the new projection [203, 204]. In contrast, for the raw data-based clustering approaches, the spatial and temporal characteristics of objects are directly used for defining similarity. Raw data-based approaches are more frequently used for application independent tasks but therefore often suffer from equal length assumptions [205]. In model-based clustering approaches, the similarity of models learned on the trajectories are used to define groups of objects by means of the EM algorithm. For the grouping rule of clustering, there are also several approaches (see Table 1.4).

Using direct clustering methods like k-means or fuzzy c-means (FCM), the initial guess of clusters is optimized in several iterations by adjusting each cluster

Trajectory clustering technique	Publication
Direct	Pelekis 2007 [206], Chen 2005 [163]
Agglomerative	Kisilevich 2010 [207], Chen 1998 [208]
Divisive	Fu 2005 [209], Morris 2008 [200]
Hybrid	Morris 2009 [171], Lee 2008, [210]
Spectral	Porikli 2004 [10], Zelnik-Manor 2001 [211]
Density-based	Frentzos 2007 [212], Lee 2007 [213]
Micro clustering	Hwang 2005 [214], Li 2004 [215]
Progressive clustering	Rinzivillo 2008 [216], Andrienko 2009 [217]

Table 1.4: Clustering techniques for trajectory data.

component to find the optimal solution [207, 218]. Using hierarchical methods, the data is slit (divisive clustering) or merge (agglomerative clustering) generating a cluster-tree structure [208, 219–221]. Hybrid clustering approaches therefore combine divisive and agglomerative techniques by applying different criteria within the agglomerative and divisive phase. In contrast, the graph and density-based clustering method (OPTICS and DBSCAN) form dense clusters described by reachability plots with a not limiting group size and shape characteristic [150, 222]. To find a defined number of clusters, also the min-cut partitioning algorithm can be used [223]. The spectral clustering methods rely on eigendecomposition of the similarity matrix without making any conjecture about existing distributions of the data points [11, 211, 224, 225].

There are also some special approaches like micro-clustering [214, 226]. Here, trajectories are separated into segments and the method tries to find close time spans where all segments are pair-wise close [213, 215]. Another case is progressive clustering using different distance functions for each cluster step resulting in a further refinement of the clustering result [216, 217].

Anomaly detection

A special case of trajectory clustering is anomaly detection, where objects that are too far away from the clusters are regarded as anomalies [227, 228]. Here,

the words anomaly, abnormal, unusual and outlier trajectory characterize certain movement behaviors that clearly differ from other movement behaviors or have a low probability of occurrence [229, 230].

In most of the literature, a model based on normal trajectory behavior is learned automatically. Anomalous events can be classified by comparing a new trajectory with this learned model [200, 231].

In general, the existing outlier detection algorithms are classified into distance-based [232, 233], distribution-based [234], density-based [235, 236] and deviation-based algorithms [49, 237].

Here, the applications from several fields reach from detecting anomalies in vessel movement deviating from the defined path [49], finding outlier in hurricane paths [194], detecting abnormal behavioral patterns in animal movement [238] up to security and travel surveillance where outliers can be a potential safety threat [239].

Integrating Prior Knowledge

The use of prior knowledge is an important element in trajectory knowledge discovery algorithms [146, 160]. For example, several approaches incorporate specific information about the trajectory movement dynamics (e.g. maximum velocities and acceleration) [240, 241] and their spatio-temporal boundaries (e.g. the preference for common used road and route networks) to adjust and improve the outcome of trajectory clustering algorithms [216, 242]. Characteristics of trajectories are often only representative in a narrow time interval, that has to be set on the basis of prior knowledge. In contrast, calculated on the complete time interval the desired effects are often averaged out and the trajectory characteristics cannot be used for the given analysis task [17]. Further, prior knowledge is often required to add semantics to the trajectories in terms of underlying images or maps to set the trajectories in a given context [243–245]. Also, prior knowledge is an important component in fusing heuristics for fragmented tracks [246–248]. However, the quality and quantity of prior knowledge is varying according the level of detail of the regarded objects. Mostly, on a very high level of detail, the quality and quantity of the prior knowledge sparsely exist and is not always suitable for semi-supervised methods, such as

clustering [249]. There are different types of prior knowledge that can be incorporated within the analysis of trajectory databases, reaching from acquisition conditions, spatial and temporal extend, morphology and geometry of the underlying objects up to object interactions and coherence [250–252]. Also, several methods to incorporate the prior knowledge within the knowledge discovery in trajectory data bases exist. For example, the trajectory patterns and the number of given clusters for trajectory clustering algorithms are often chosen a priori [217, 253–255]. Additionally, a priori defined boundary conditions are used in the context of anomaly detection [227, 256]. Furthermore, the a priori verbal defined characteristics of trajectory movement, shape and dynamics are used to extract the corresponding trajectory patterns for the analysis [145, 178, 256].

Handling Fragmented Tracks in the Analysis

In Section 2.1, problem classes of trajectory databases are presented. The more complex the problem class, the higher the probability of occurring fragmented trajectories, called tracklets, where a single objects could not be tracked perfectly over the complete time interval [5, 78, 257]. This is due to the fact that conventional frameworks of object detection and tracking hardly performed a precise tracking, as the acquisition conditions getting worse and the density of objects increases [247, 258]. Further, influences on an increasing fragmentation of the tracking results are artifacts, double detections, shadow and reflections as well as fused, destroyed and emerging objects, division ambiguities and object crossing [259]. Almost all conventional methods to extract information out of trajectory databases require the trajectories to be over the complete time interval instead of being fragmented [172, 247]. Especially, several clustering algorithm [160, 260, 261], and trajectory pattern extraction algorithms cannot work without the assumption of error free, not fragmented trajectories [146, 242, 262]. Without complete trajectories, the global characteristics of trajectories required for clustering and other trajectory data mining methods, can not be described. Further, no statements on a detailed single object level can be made in a global context without having the trajectory, as global movement path, of the objects. There are several procedures to handle such fragmented trajectories within the existing approaches of analyzing trajectory data [261]. Often, human experts have to manually review the fragmented trajectories and make appropriate manual curations [141]. Without an efficient manual curation

guidance, this is impossible. However, there is a lack of appropriate software solutions to assist the user in the manual correction of large-scale trajectory databases [259]. Another procedure is to simply eliminate the fragmented trajectories and only keep the complete ones for further analysis. If assumptions for each single object are important this approach is not feasible [172]. Further, extrapolation and forced fusion of fragmented trajectories complete the range of procedures [263].

Visual Explorative Data Analysis

Within the visual explorative data analysis and visual analytics process, visual analysis methods are combined with automatic data mining algorithms involving human interaction and integrating prior knowledge for the purpose of gaining knowledge from the data [264–267]. Further, visual data exploration is often used to confirm generated hypothesis produced by automatic methods [207, 268, 269]. These automatic methods discover interesting patterns with regard to a mathematical optimization function, but these patterns do not necessarily correlate with the interestingness of a human analysis [270–272]. Therefore, the human observer needs the possibility to evaluate and refine the results by interacting with the data, modify parameter settings or chose other automatic analysis algorithms [153, 273, 274]. The visualization of huge trajectory datasets additionally leads to growing challenges regarding the presentation of the data in a way that a sufficient analysis and interpretation of the information within the data is assured [216, 275, 276]. Here, the interactive access to the visualizations of the trajectory data is a fundamental prerequisite for human experts to explore underlying phenomena of movement dynamics and for a focused analysis of data of interest [217, 277–279].

There are several existing approaches to visual explorative data analysis leading from traffic management control, vessel surveillance tasks up to airspace monitoring and the observation of natural phenomena [24, 49, 271, 280]. An alternation of automatic and visual methods (see Figure 1.6) is characteristic for these approaches and leading to an ongoing refinement and evaluation of the results [23, 213, 281]. Additionally, the discovery of misguiding results in an interim step at an early stage is leading to more trustworthy results as well

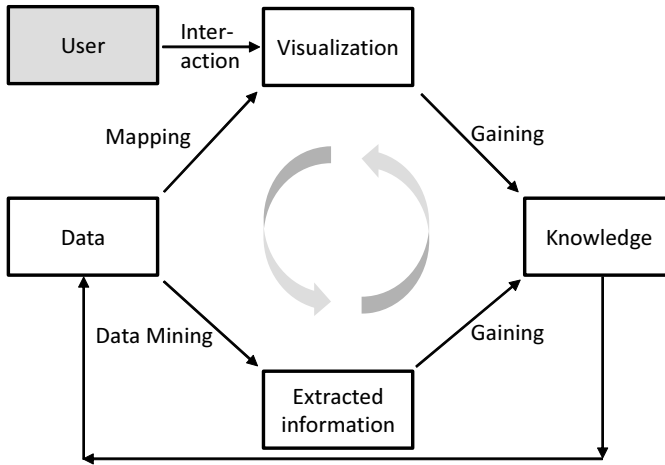


Figure 1.6: Scheme for visual explorative data analysis adapted from [264].

as the use of a combination of multiple views to compare several movement characteristics [282].

1.2.4 Available Software Solutions

There are a huge amount of open-source and commercial software tools available that handle the visualization and analysis of trajectory databases. Starting from Mov-IT [283] using augmented phenomenology for visualization, MoveMine [193, 194] incorporating further relationship mining, MinUS [284] focusing on mobility mining and similarity calculation up to GeT.Move [285] with the possibility to detect swarms, convoys and group patterns. M-Atlas [216, 286] supplying semantic trajectory mining and TraSeer [287] focusing on visual explorative data analysis of trajectory data are further software tools in this field. Most of the software was specifically developed for 2D trajectory databases such as cars, animals, pedestrians and ships pointing out the need for a software combining the mining and visualization of huge amounts of trajectory data in an efficient and interactive way [264, 268, 274]. Additionally, most of the software tools focus either on the interactive visualization tasks for

trajectory data or on the trajectory data mining tasks. A combination of both functionality for 3D trajectory datasets is therefore an optimal solution to handle the tremendous arising amount of new gained 3D datasets.

Moreover, several tools for the data mining task exist like the Matlab toolbox SciXMiner [288], RapidMiner [289], KNIME [290] and WEKA [291]. In the context of biological trajectory data visualization exist many software tools such as TrackMate [292], CellTracker [293], Fiji [294], XPIWIT [295, 296], MaMuT [297], Imaris¹, BigDataViewer [298], Mastodon² (on the basis of MaMuT), (Bitplane AG, Zurich, Switzerland), Paraview [299] and CATMAID [300].

¹ <http://www.bitplane.com>, state: 18.09.2018

² <https://github.com/bigdataviewer/mastodon-collection>, state 07.09.2018

1.3 Open Questions

In consideration of actual research and existing elaborate methodology described in the previous section, there are still several open questions in the domain of trajectory pattern mining and analysis that are still widely unsolved:

1. Most analysis pipelines for 2D+t and 3D+t objects are based on fully automated analysis algorithms. However, in complex problem classes compromising artifacts and errors in tracking, such automated analysis leads to erroneous analysis results using feature driven group extraction.
2. A detailed analysis of 2D+t and 3D+t object data often require the use of prior knowledge combined with automated analysis routines. However, existing prior knowledge is often not sufficiently incorporated in the analysis task. Furthermore, there is no coherent approach to systematically embed and use the convenient prior knowledge within the analysis of complex 2D+t and 3D+t object databases.
3. Existing analysis pipelines for extracting information out of spatio-temporal object databases generally rely on black box approaches and therefore often lack of sufficient interactive visualization to guide the analysis and efficiently integrate the prior knowledge in all levels of detail. Only such a visual interface allows to incorporate prior existing knowledge adequately.
4. State of the art tracking algorithms are not able to extract error-free tracks out of dense objects databases with a huge amount of objects due to noise, double detections and other artifacts. Therefore, methods to handle these incomplete tracks in a visual interactive way are crucial to cope with this problem and compensate the errors within the tracking algorithms.
5. Existing approaches to analyze spatio-temporal trajectory data heavily rely on the assumption of having perfect error-free tracks. If this precondition is missed, the quality of extracting relevant groups out of huge amount of 2D+t or 3D+t data is heavily negatively affected by the errors within the tracking algorithms.

6. Even though analysis tasks can be used to extract relevant groups out of complex 2D+t and 3D+t object datasets, the problem is to efficiently transfer these analysis pipelines to new datasets by handling the divergences of the datasets.
7. Although many usable software tools for the analysis of tracking data have been presented in the path, often there is only limited applicability to efficiently incorporate prior knowledge in the discovery of 2D+t and 3D+t datasets of moving objects. Especially, the interactive handling of huge amounts of 3D+t data by incorporating computational intelligence methods such as clustering is often not feasible. Furthermore, many applications only offer a limited applicability to handle fragmented tracks and tracks of dividing and merging objects.
8. The aforementioned problems often makes the analysis of real-world datasets impossible. One field of application in real-world datasets arise from state-of-the-art light-sheet microscopy techniques producing a huge amount of 2D+t and 3D+t data of moving cells in an uncompromisingly high quality. However, the systematic investigation of these data often lacks adequate analysis frameworks to handle the data efficiently. Thus, there is an increasing need for methods to investigate, analyze and interpret the behavior of individual cells and groups of cells to proof and unveil hypothesis in developmental biology quantitatively and fast.

1.4 Objectives and Thesis Outline

Based on the previous mentioned unsolved problems, the central objectives of the present thesis are:

1. To develop a new systematic concept for the analysis of 2D+t and 3D+t trajectory databases by efficiently incorporating prior knowledge by designing an interactive visual interface combined with new trajectory analysis methods.

2. To devise methods compensating the errors leading to the fragmentation of the tracks within the tracking algorithms to efficiently extract relevant knowledge.
3. The transfer of trajectory analysis pipeline to new datasets by handling the divergences within the datasets.
4. To develop new validation benchmarks incorporating the difficulties of real-world trajectory databases and therefore allow to validate the robustness of entire trajectory analysis pipelines.
5. The implementation of the developed framework for interactive large-scale trajectory analysis in an efficient and user-friendly open-source software toolkit.
6. The proof of the successful performance of the developed framework on large-scale trajectory databases in the context of developmental biology.

To validate the methods for analyzing huge trajectory databases in a systematic way and proof the robustness against noise and fragmentation of tracks within the data, a newly developed benchmark is presented in Chapter 2. For the efficient analysis of trajectory databases, Chapter 3 presents a new generic concept to efficiently incorporate prior knowledge within the analysis pipeline by combining interactive visual explorative data analysis and fully automated trajectory clustering approaches. Furthermore, Chapter 3 also demonstrates how to handle different existing problem classes of trajectory databases in the analysis process, reaching from the compensation of fragmented tracks and the transfer of whole analysis pipelines to new databases by additionally handling the divergence among databases efficiently. The interactive visual explorative framework supporting the analysis task is illustrated in Chapter 4 focusing on the different possibilities to efficiently incorporating existing prior knowledge within the analysis. All existing functionality was implemented as part of a platform-independent open-source software tool described in Chapter 5. Moreover, a user-study is presented in Chapter 6 evaluating the methodological and interactive approaches developed in Chapter 3 and 4. Besides the validation of the new methods on the synthetic benchmark, the presented concepts were applied to huge trajectory databases generated from large-scale microscopy data in the context of developmental biology in Chapter 7 to underpin the practical

use and outcome of the presented concepts. Finally, Chapter 8 summarizes the attained work and gives an outlook on potential future projects.

2 Synthetic Benchmark Generation for Quality-Dependent Trajectory Databases

The analysis of large-scale 2D+t and 3D+t trajectory databases offers valuable information about underlying phenomena and process dynamics in many different research areas. Trajectory data mining methods have become enormously important and are used to access the information hidden within the spatio-temporal trajectories to unveil characteristic movement behavior or to find interesting patterns. However, state-of-the-art tracking technologies do not always lead to perfect tracks over the complete acquisition interval. Depending on the origin of the tracking acquisition methods, there arise many different problem classes for tracking scenarios influenced by the density of objects, the distinguishability and dividing object characteristics within the data, to name only a few. To evaluate trajectory data mining methods for several different real-world problem classes, Chapter 2 presents a new systematic approach to simulate quality-dependent trajectory databases. Therefore, Section 2.1 presents a detailed systematization of problem classes arising in trajectory databases. Further systematic errors leading to fragmented tracks are discussed in Section 2.2. A general scheme to create hybrid trajectory databases consisting of a predefined problem class combined with parameterizable fragmentation artifacts is covered in Section 2.3. To systematically evaluate the newly developed methodology in Chapter 3 and 4, Section 2.4 provides problem-specific benchmark databases that are used to access several aspects of the integration of prior knowledge in the knowledge discovery of large scale 2D+t and 3D+t trajectory databases in a quantitative way.

2.1 Systematization of Problem Classes

Depending on several characteristics, different problem classes are specified in Table 2.1. Here, the problem classes are distinguished by the quality of data, the density of objects, the dividing object characteristics, the appearance and disappearance of objects as well as the distinctiveness of objects quantitatively describing if the same objects are included. To describe the complexity of the single problem classes, the number of problems is subjectively assessed. Here, in each characteristic, the easier method with regard to the generation of valid tracks is rated with the number zero and the more complex method with the number one. For each problem class, the corresponding numbers are summed up.

Problem classes 01-03 are based on direct measures such as positional information provided by a global positioning system (GPS) to detect moving objects like pedestrians, animals, ships, airplanes and cars [47–49, 301–303]. The main difference between the Problem classes 01-03 is the density of objects and the existence of fulltracks of the complete recording time. In contrast to Problem classes 01-03 the Problem classes 04-20 use indirect measure such as image analysis to detect objects and capture the dynamics which often leads to fragmented tracks due to noise and artifacts within the subsequent tracking phase [304]. A defined area with a constant number of distinguishable objects with low to middle probability describe Problem class 04, e.g. cars, animals or pedestrians that are clearly characterized according to pre-defined visual characteristics [305–310]. The Problem class 05 is similar to the Problem class 04, with the exception that the objects are not distinguishable, e.g. moving particles that all look the same [55, 311–313]. In contrast to the Problem classes 04 and 05, the Problem classes 06 and 07 do not contain the same objects, meaning that the moving objects like cars pedestrians and particles that enter and leave the recording scene resulting in a time-variant number of objects [74, 314–316]. An additional dividing of the objects with low to middle density describes the Problem classes 08 and 09, e.g. a car stops, a person is leaving the car and the car drives again [317–320]. The only difference between these two problem classes is the distinctiveness of the objects [321–326]. Further, the appearance and disappearance of objects in a scene describe the increased complexity of Problem class 10 and 11 in contrast to the Problem classes 08 and 09 [327–329]. For in-

stance, cells or particles in a fluid can agglomerate and divide again and due to noise in the cloudy liquids leading to fragmented tracks [330–333]. A high density of non-dividing objects describes the Problem classes 12 and 13. A huge amount of particles in air or liquid, as well as a dense crowd of pedestrians, can be mentioned as examples of these problem classes [31, 31, 61, 334–339]. The low to middle probabilities of dividing objects such as pedestrians that meet and spread again describe Problem classes 14 and 15 [340, 341]. Additionally, Problem classes 16 and 17 contain a high density of the same dividing objects within the scene [292]. Problem class 17 in contrast to Problem class 16 additionally contain appearing and disappearing objects. Occlusion and other tracking artifacts lead to highly fragmented tracks requiring a complex and time-consuming fusion. With an increasing complexity of the problem classes, the fragmentation of the tracks and the complexity of a fusion also increase. The high density of indistinguishable and dividing objects combined with the appearance and disappearance of these objects describes Problem class 18. Particles and pedestrians that agglomerate and divide again without the possibility to be distinguished can be exemplary mentioned for this problem class [342? ? , 343]. In contrast to Problem class 18, a non-constant number of objects due to noise detection and artifacts makes Problem class 19 even more complex [344, 345]. Additionally, indistinguishable objects point out that Problem class 20 has the highest complexity of all problem classes [141, 283]. In general, it can be noted that the higher the complexity of the problem classes, the higher is the difficulty to analyze the tracking data due to an increasing fragmentation of tracks and due to split and merge events leading to an enormous fusion effort [317, 318, 320, 328].

2.2 Reasons for Fragmented Tracks

The different problem classes handling several origins for highly fragmented trajectory databases are listed in Table 2.1. Here, the influence of various types of tracking problems like quality, density, dividing objects or disappearance of objects to the number of problems is shown.

A detailed scheme for the reasons leading to fragmented tracks in trajectory databases is presented in Figure 2.1. Noise artifacts can lead to false detections

Problem Class	Quality of Data		Density of Objects		Same Objects		Dividing Objects		Appear/Disappear		Fulltrack Fusion			Fulltracks		Distinguishable		Number of Problems
	Direct Measurement	Indirect Measurement	Low-middle	High	Yes	No	Yes	No	Yes	No	Easy	Complex	Yes	No	Yes	No		
01	x		x		x										x			0
02	x			x	x										x			1
03	x		x		x											x		2
04		x	x		x											x		2
05		x	x		x											x		4
06		x	x			x										x		3
07		x	x			x										x		5
08		x	x		x			x								x		2
09		x	x		x			x								x		5
10		x	x		x			x								x		4
11		x	x		x			x								x		5
12		x		x	x											x		4
13		x		x	x											x		5
14		x	x		x			x								x		3
15		x	x		x			x								x		5
16		x		x	x			x								x		5
17		x		x	x			x								x		5
18		x		x	x			x								x		7
19		x		x		x		x								x		7
20		x		x		x		x								x		8

Table 2.1: Systematization of various problem classes observed in different applications of automatic tracking approaches. The different levels of complexity arise from dataset properties like quality, density, dividing objects or disappearance of objects.

resulting in short fragments of invalid tracks (Figure 2.1A). The short vanishing of objects causes gaps in a track where the object is lost for a short time period (Figure 2.1B). Further, the appearance of double detections due to segmentation errors can lead to the abortion of the track and at the same time the emergence of a new track in the 2.1C). In image-based tracking technologies, the emergence of shadows and reflections cause invalid segmentation results leading to invalid track fragments (Figure 2.1D). Additionally, artifacts resulting from the destroying (Figure 2.1E) and emerging (Figure 2.1F) of objects lead to the abortion of the tracks and to illegal connections of fragmented tracks (Figure 2.1G). In case of crossing objects, the segmentation often lead to errors in the crossing regions resulting in the abortion of the tracks (Figure 2.1H). Further, division ambiguities resulting from events of dividing objects can also cause the abortion of the tracks (Figure 2.1I) as well as objects leaving the region of image acquisition (Figure 2.1J).

2.3 Generate Quality-Dependent Trajectory Benchmark Databases

To generate hybrid combinations of trajectories with user-defined movement characteristics and different levels of fragmentation artifacts, Figure 2.2 present a systematic approach for the generation of quality-dependent trajectory benchmark databases.

First, the fulltrack generation module (*FGM*) generates a trajectory database (*TDB*) containing a set of trajectories $\mathcal{T}_{set} = \{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_{N_T}\}$, with $N_T \in \mathbb{N}$ as the number of trajectories within the trajectory database. Each trajectory \mathcal{T} within the trajectory database \mathcal{T}_{set} is therefore simulated as a sequence of discrete points $\mathcal{T} = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{N_P}\}$. For each spherical point in 2D ($\mathbf{p}_i = \{x_i, y_i, t_i\}$) or 3D ($\mathbf{p}_i = \{x_i, y_i, z_i, t_i\}$) the spatial coordinates x_i, y_i, z_i are simulated for given time points t_i . The quality reduction module (*QRM*) then deletes linkings between subsequent trajectory points and therefore creates fragmented tracks represented in a new set of shorter trajectories $\mathcal{T}_{set,QR}$. Different parameter settings within the fulltrack generation module *FGM* as well as within the quality reduction module *QRM* allows to efficiently simulate different characteristics of the generated trajectory database *TDB*.

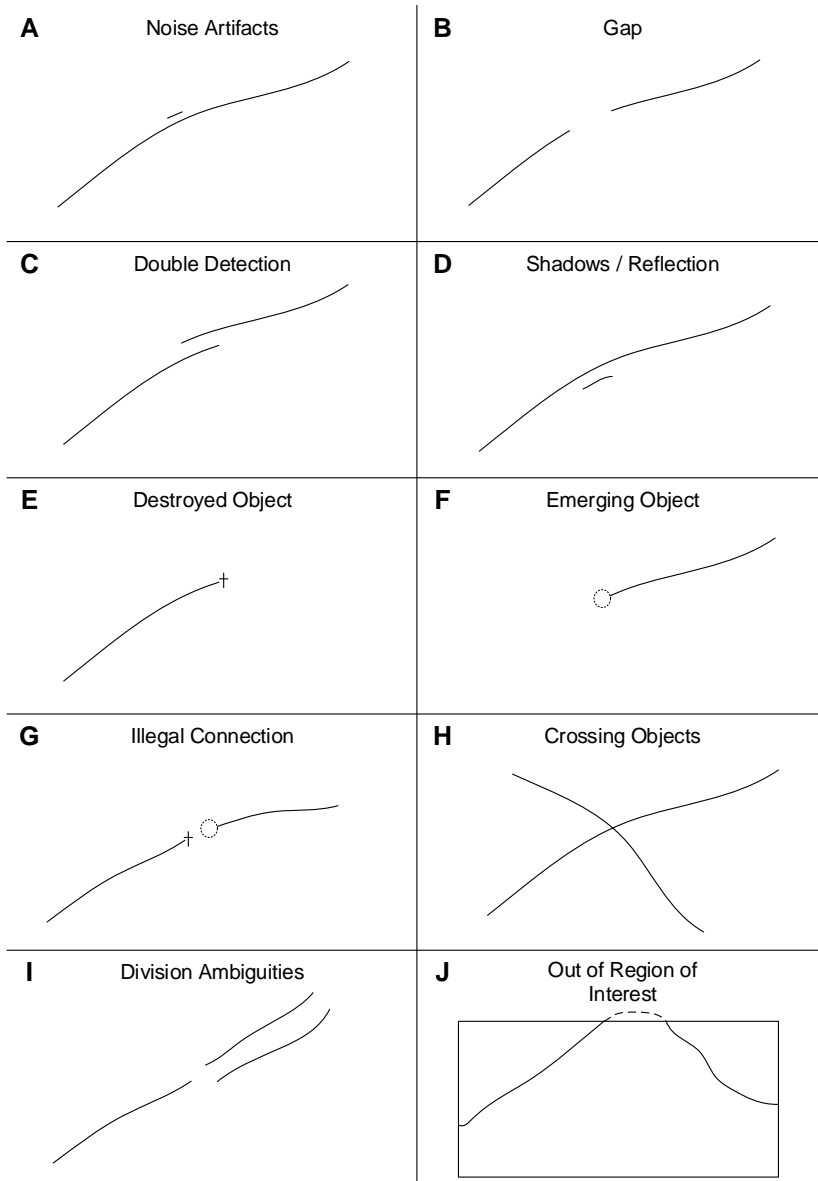


Figure 2.1: Reasons for fragmented tracks. (A) Noise artifacts leads to false detections, (B) Gap caused by a short vanishing of object, (C) Double detection of objects, (D) Shadows and reflections of objects, (E) Object is destroyed, (F) Object is emerging, (F) Illegal connection of fragmented tracks, (H) Crossing objects can lead to aborting tracks, (I) Ambiguities of dividing objects and (J) Objects out of focus.

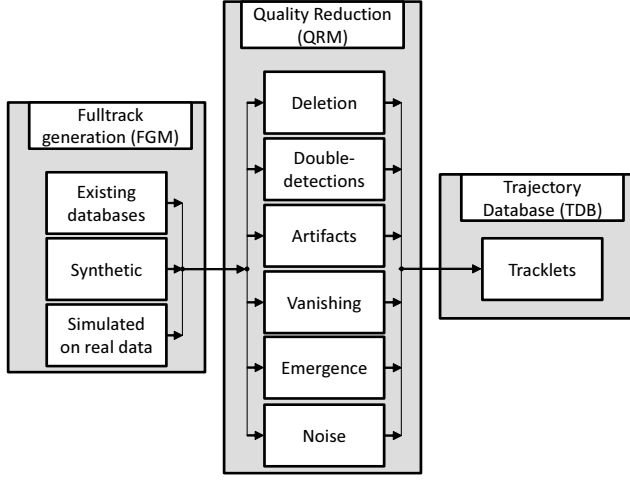


Figure 2.2: Synthetic creation of quality-dependent trajectory benchmark databases. The Fulltrack Generation Module (FGM) generates a trajectory database (TDB) consisting of one or more set of trajectories (\mathcal{T}_{set}). On the basis of the TDB , the Quality Reduction Module (QRM) deletes linkings between subsequent trajectory points. The result is a new trajectory database TDB containing a set of shorter trajectories due to fragmentation.

2.3.1 Fulltrack Generation Module

The output of the fulltrack generation module (FGM) is a trajectory database (TDB) containing a set of trajectories $\mathcal{T}_{set} = \{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_{N_T}\}$. There are various possibilities to generate the trajectory database TDB . Namely, FGM_{ED} on the basis of existing databases (ED), FGM_{SY} on the basis of purely synthetic datasets or on the basis of FGM_{SM} simulated trajectories based on real-world data.

Existing Databases

One option is to use existing trajectory databases with the FGM_{ED} module. Therefore, the set of trajectories \mathcal{T}_{set} is already available from another source and therefore has not to be simulated. All existing trajectory databases can be used, as long as each trajectory within the trajectory dataset \mathcal{T}_{set} consists of subsequent points p_i with a constant frame rate. In cases of non-constant

frame rates, preprocessing steps are required to ensure a unified frame rate for all trajectories. This allows using existing datasets from all different areas of applications (see Section 1.2.1 for more details). On this basis, the quality reduction module QRM can simulate quality-dependent characteristics of these existing databases, allowing in-depth investigation of algorithm behavior depending on the quality of the dataset.

Purely Synthetic

There is also the possibility to generate the set of trajectories \mathcal{T}_{set} purely synthetically with the FGM_{SY} module. This allows to simulate arbitrary spatio-temporal movement characteristics of the trajectories and therefore enables the generation of problem-specific trajectory databases. This type of approach enables the detailed investigation of existing algorithms on specific synthetic problem characteristics. For the synthetic generation of the set trajectories $\mathcal{T}_{set} = \{T_1, T_2, \dots, T_{N_T}\}$, each trajectory is simulated as a discrete sequence of points $\mathcal{T} = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{N_P}\}$. The direction and velocity vectors for each time point t_i are given, resulting in predefined movement characteristics of the trajectories. The purely synthetic approach of generating a set of trajectories therefore allows depicting spatio-temporal patterns in trajectory datasets reaching from simple up to highly complex movement patterns. Further, different noise levels can be simulated by adding uniformly distributed noise with the amplitude $Q_{A,Noise}$ to each point \mathbf{p}_i of the trajectory \mathcal{T} .

Simulated on Real Data

The simulation module FGM_{SM} works on real datasets and simulates the trajectories based on existing movement characteristics of objects [110]. This allows generating realistic benchmarks on the basis of existing trajectory databases with various parametrizable options. Therefore, object location, density information and displacement vectors of the real dataset are used. The simulation leads to realistic movement patterns, object division characteristics and neighborhood related movement dynamics. For further details, and a detailed description of the simulation algorithms, see [110].

2.3.2 Quality Reduction Module

There are several reasons leading to fragmented trajectories described in Section 2.2 in detail. For the synthetic generation of quality-dependent trajectory benchmarks only the resulting fragmentation is simulated. Therefore, we present two quality reduction modules (*QRM*) that are based on Markov chains (*MB*) or Bunch deletion (*BB*), producing fragmented trajectory data are presented in this thesis, namely the QRM_{MB} and QRM_{BB} . The vector $\Theta[P_{t,del}, P_{del}]$ specifies the percentage of trajectories that are fragmented $P_{t,del}$ and the percentage (P_{del}) within a trajectory that is deleted. A notation of $\Theta[1, 0.3]$ would thus indicate that 100 % of the trajectories are fragmented by deleting 30 % of each trajectory.

Markov-Based Quality Reduction Module

The Markov-based quality reduction module QRM_{MB} is based on a two stage Markov model (MM) with transition matrix \mathbf{T} (Figure 2.3):

$$\mathbf{T} = \begin{bmatrix} p_{11} & 1 - p_{11} \\ 1 - p_{22} & p_{22} \end{bmatrix} \quad (2.1)$$

Hereby, state $S = 1$ indicates high tracking quality without missing objects and state $S = 2$ indicates bad tracking quality with missing objects leading to fragmentation. Setting the parameters p_{ij} within the transition matrix \mathbf{T} , the Markov model produces a sequence of states $S_{state} = \{S = 1, S = 2, \dots, S = N_{STS}\}$, with $N_{STS} \in \mathbb{N}$ describing the number of states within S_{state} . In case of simulated fragmentation characteristics, the number of states N_{STS} is equal the number of time points within a trajectory N_P . Each time point within the trajectory is therefore assigned to one state S . A given percentage of deletion $P_{del,MB}$ within a trajectory defines the number $N_{P,del}$ of points that have to be deleted in a trajectory. The probability $P(Deletion|S = 1)$ (short notation $P_{Del,S=1}$) to delete a point, if the state is $S = 1$, is much smaller than the prob-

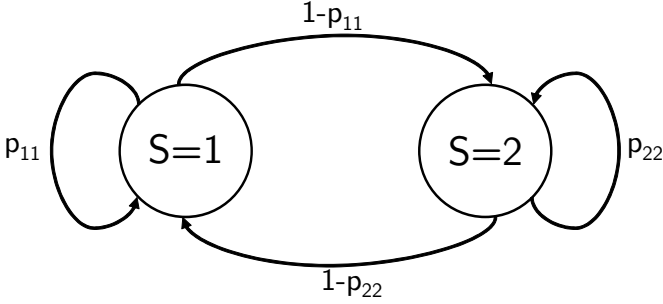


Figure 2.3: Two state Markov model to simulate fragmentation in trajectory databases.

ability $P(\text{Deletion}|S = 2)$ to not delete a point (short notation $P_{Del,S=2}$) in case of state $S = 2$:

$$P(\text{Deletion}|S = 1) \ll P(\text{Deletion}|S = 2) \quad (2.2)$$

Using the probabilities $P(\text{Deletion}|S = 1)$ and $P(\text{Deletion}|S = 2)$, the predefined number $N_{P,del}$ (result from $P_{del,MB}$) of points are sampled within a trajectory to be deleted. Here, the points correlated with $S = 1$ have a much higher probability to be sampled. The result are fragmented trajectories instead of complete trajectories existing in the complete time interval. The parameters p_{11} and p_{22} of the transition matrix \mathbf{T} can be adjusted to simulate different fragmentation behaviors. In the case of choosing p_{11} and p_{22} both small, the result are equally fragmented trajectories with deleted points equally distributed over the complete time span of the trajectory. Whereas, choosing p_{11} and p_{22} both to be high, the result are fragmented trajectories with longer segments of deleted points. Further, different fragmentation characteristics can be achieved by adapting the parameters p_{11} and p_{22} . In the following the notation $QRM_{MB}[p_{11}, p_{22}, P_{Del,S=1}, P_{Del,S=2}]$ is used for the Markov-based quality reduction module QRM_{MB} with parameters $p_{11}, p_{22}, P_{Del,S=1}$ and $P_{Del,S=2}$. The values of the single parameters are listed as subscript indices. Figure 2.4A-C exemplary shows the trajectories in the XY-plane of the fragmentation result for different parameter combinations $QRM_{MB}[p_{11}, p_{22}, P_{Del,S=1}, P_{Del,S=2}]$ of the Markov-based quality reduction module simulating fragmentation characteristics resulting from different artifacts depicted in Figure 2.1. The start points

of the trajectories in Figure 2.4 are randomly initialized in the space resulting in different vertical distances. Hereby, Figure 2.4A depicts short intervals of deleted time points nearly equally distributed over each trajectory. For the deletion in Figure 2.4B and C, the interval of deleted time points gets longer and therefore also the remaining trajectory fragments.

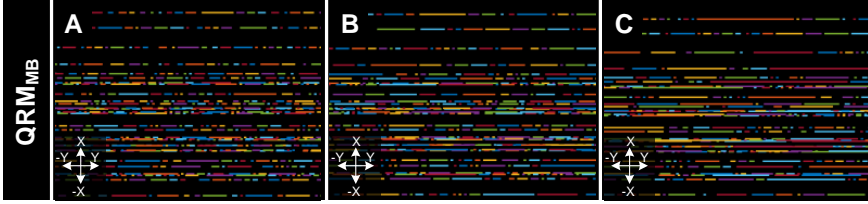


Figure 2.4: Different parameter examples of the Markov-based quality reduction modules. The effect of the parameter choice to the fragmentation of the trajectories with $\Theta[1, 0.25]$. The Markov-based quality reduction module QRM_{BB} with three different parameter settings is shown. (A) $QRM_{MB}[0.5, 0.5, 0.001, 0.99]$, (B) $QRM_{MB}[0.8, 0.2, 0.001, 0.99]$, (C) $QRM_{MB}[0.9, 0.1, 0.001, 0.99]$. The color code indicates the trajectory IDs.

Bunch-Based Quality Reduction Module

The bunch deletion quality reduction module QRM_{BB} is able to simulate different fragmentation characteristics for trajectory data leading from the deletion of short fragments that are equally distributed over the complete time interval up to the deletion of larger connected fragments. The comprehensive scheme for the QRM_{BB} is shown in Figure 2.6. Each trajectory consists of a list of linkings $L_n = \{L_1, L_2, \dots, L_{N_L}\}$. Here, a linking is the connection of two successive point IDs between consecutive time points. The number of linkings $N_L = N_p - 1$ is hereby one less than the number of points N_p within a trajectory. The QRM_{BB} module samples linkings L_n that are deleted from of the trajectory depending on the parametrization of QRM_{BB} . The deletion of linkings lead to fragmented trajectories. Initially, the number of linkings in total $N_{L,del}$ that are deleted is predefined. Furthermore, the parameters of the QRM_{BB} module, namely the mean bunch length μ_{BD} , the standard deviation of the bunch length σ_{BD} and the minimal segment length $\tau_{Seg,min}$ depicting a constraint to achieve segments not shorter then $\tau_{Seg,min}$ are specified. In an

iterative approach, possible start positions of bunch deletions are chosen. Afterwards, using the parameters μ_{BD} and σ_{BD} , the length of the bunch that is deleted is sampled using a normal distribution. The mean bunch length μ_{BD} defines the mean length of the inserted gaps whereas the minimal segment length $\tau_{Seg,min}$ serve as a constraint for the placing of the gap to be inserted.

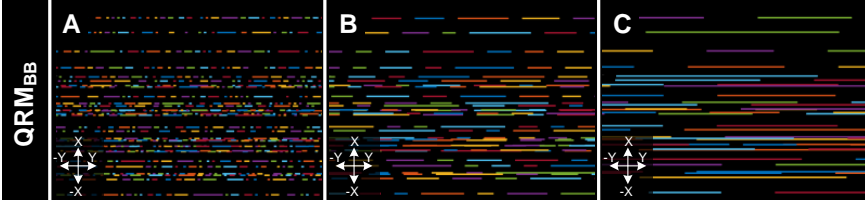


Figure 2.5: Different parameter examples of Bunch-based quality reduction modules. The effect of the parameter choice to the fragmentation of the trajectories with $\Theta[1, 0.25]$. The Bunch-based quality reduction module QRM_{BB} with different parameter settings is displayed. (A) $QRM_{BB}[1, 0, 1]$, (B) $QRM_{BB}[5, 0, 5]$, (C) $QRM_{BB}[20, 0, 20]$. The color code indicates the trajectory IDs.

The linkings that are deleted are added to the deletion list $L_{sel,n}$ containing after each iteration $N_{L,ch}$ linkings. As long as the number of linkings that should be deleted $N_{L,del}$ is smaller than the total number of linkings in the deletion list $N_{L,ch}$, further linkings are deleted within the trajectory (see Figure 2.6). This iterative process is applied to all N_T trajectories of the set of trajectories \mathcal{T}_{set} . The result is a new set of trajectories $\mathcal{T}_{set,QR}$ containing fragmented tracks out of the original set \mathcal{T}_{set} . The notation that is used in this thesis to list the parametrization of the QRM_{BB} module is $QRM_{BB}[\mu_{BD}, \sigma_{BD}, \tau_{Seg,min}]$ with the parameter values as subscript indices. A notation of $QRM_{BB}[1, 0, 1]$ would exemplary point out a bunch deletion-based quality reduction module with an mean bunch length μ_{BD} equal 1, a standard deviation of the bunch length σ_{BD} equal 0 and a minimal segmented length $\tau_{Seg,min}$ equal to 1. Examples of several fragmentation results depending on the parameter choice of $QRM_{BB}[\mu_{BD}, \sigma_{BD}, \tau_{Seg,min}]$ are shown in Figure 2.5A-C. Here, the trajectories in the XY-plane are visualized with random initialized start points resulting in different vertical distances. Figure 2.5A shows the result of the bunch deletion module with short bunches equally distributed over the trajectories. In Figure

2.5B and C, the deleted bundles get longer resulting in longer gaps within the trajectories and therefore also longer fragments at a stretch.

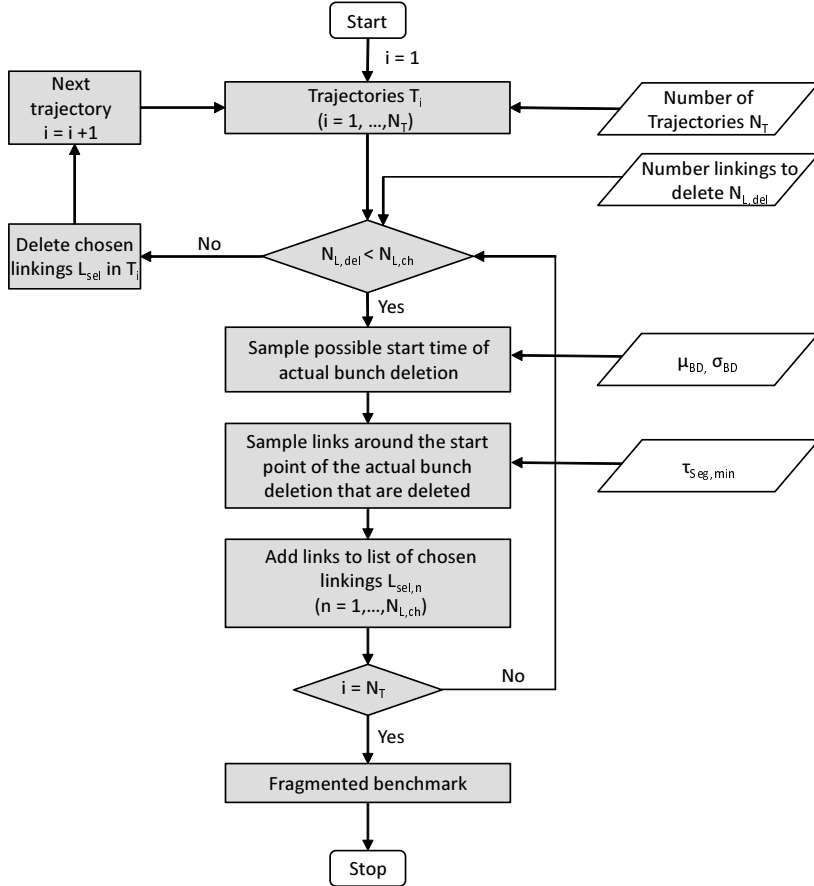


Figure 2.6: General process of the bunch deletion quality reduction module QRM_{BB} . For each trajectory a given number $N_{L,del}$ of linkings between subsequent points are deleted.

Object Division Module

In Section 2.1 the occurrence of dividing object characteristics in tracking databases is depicted. To simulate such object division characteristics, an object division module *ODM* was developed. Two parameters are used to simulate different division characteristics, namely the mean object division rate ψ_{MODR} and the standard deviation of the object division rate ψ_{SODR} . A mean division rate ψ_{MODR} of 20, for example, leads to an object division every 20 time points. The notation $ODM[\psi_{MODR}, \psi_{SODR}]$ is used to characterize the parametrization of the object division module *ODM*. The object division characteristic can be modularly added to any synthetically generated trajectory database *FGM_{SY}* of Section 2.2, leading to an enormous flexibility in the generation of benchmark trajectory databases that combine spatio-temporal characteristics and object division characteristics. Here, Figure 2.7 exemplarily shows the result of different parameter settings of the object division module.

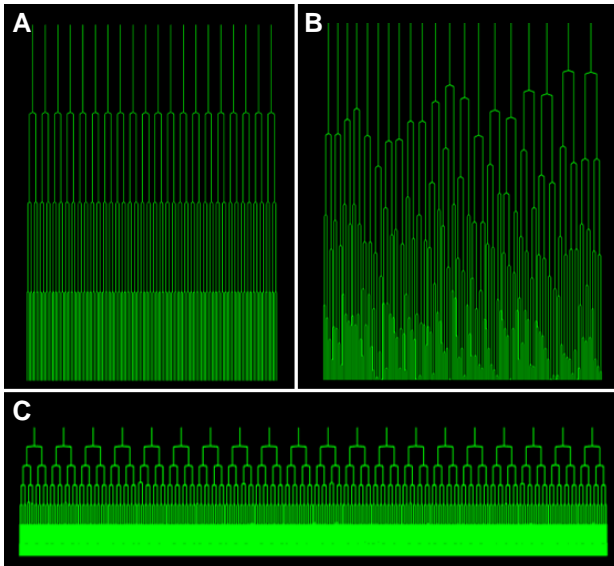


Figure 2.7: Object division simulation module. Here the lineages for different parameterizations of the object division module *ODM* are exemplarily shown for a benchmark with 200 time points of simulation. (A) $ODM[50, 0]$, (B) $ODM[50, 10]$ and (C) $ODM[20, 0]$.

In real-world trajectory databases, object divisions events often lead to the loss of the objects and therefore result in fragmented trajectories. This division tracking error is simulated using the parameter κ_{DA} describing the probability that a trajectory gets fragmented at an object division event. The parameter choice of κ_{DA} equal zero result in no fragmentation of the trajectories leading to two trajectories containing the same trajectory origin before the division event. Whereas, when κ_{DA} is equal one all trajectories get fragmented at object division events. In Figure 2.8A no fragmentation exist due to division events, whereas in Figure 2.8B 50% and in Figure 2.8C 100% of all division events lead to fragmentation.



Figure 2.8: Fragmentation through object division events. Here, exemplary three different parameter choices of κ_{DA} and the resulting fragmentation of the trajectory data in a zoomed region are shown. (A) $\kappa_{DA} = 0$, (B) $\kappa_{DA} = 0.5$ and (C) $\kappa_{DA} = 1$.

Further, any combination of the fragmentation through object division and fragmentation through the quality reduction modules QRM can be simulated. This allows to simulate any given problem class from Section 2.1 with any fragmentation characteristics in a hybrid parametrizable approach (Figure 2.9) by choosing the fragmentation characteristic according to the given problem class (Section 2.1) and assessing the emerging object division characteristic depicted within the problem class.



Figure 2.9: Fragmentation as a result of combined object division module ODM and the quality reduction module QRM_{BB} . Three different parameter combinations of object division events and the parameters of the QRM_{BB} is displayed within a zoomed region. (A) $\kappa_{DA}[0]$, $\Theta[1, 0.2]$, $QRM_{BB}[1, 0, 1]$. (B) $\kappa_{DA}[0.5]$, $\Theta[1, 0.3]$, $QRM_{BB}[1, 0, 1]$ and (C) $\kappa_{DA}[1]$, $\Theta[1, 0.4]$, $QRM_{BB}[1, 0, 1]$.

2.4 Systematic Generation of Validation Benchmarks

2.4.1 General Scheme

The combination of fulltrack generation modules FGM (Section 2.3.1) and quality reduction modules QRM (Section 2.3.2) allows a flexible and parameter-controlled simulation of trajectory benchmark databases. A general overview is displayed in Figure 2.10. For the evaluation of trajectory-based knowledge discovery approaches within large-scale trajectory databases, several benchmark databases are generated using the scheme in Figure 2.10 to evaluate the methods presented in Chapter 3 and 4.

For a given fulltrack generation module FGM different parametrizations of the quality reduction module QRM lead to a bunch of trajectory benchmark datasets with the same spatio-temporal characteristic of the trajectories defined in the FGM and different fragmentation characteristics resulting from the parametrization of the QRM . This allows in different validation scenarios to efficiently simulate quality-dependent characteristics to evaluate specific aspects of the different methods using the given parameter set listed in Table 2.2. To demonstrate the notation exemplary, three generated benchmarks trajectory databases TDB_1 , TDB_2 , TDB_3 are listed in the following. More details can be found in Table B.3.

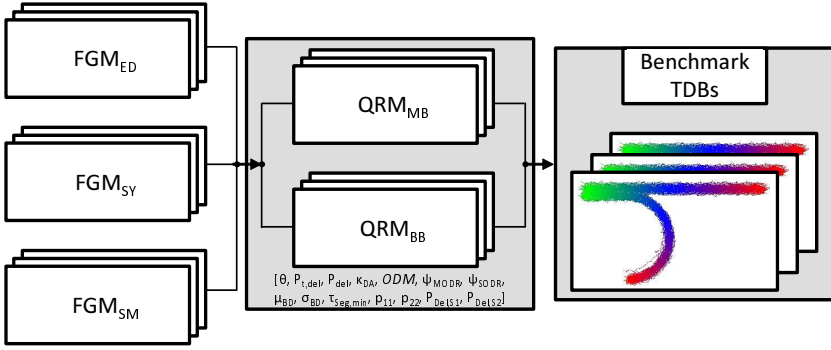


Figure 2.10: Systematic generation of validation benchmarks consisting of several full-track generation modules (FGM) generating trajectory databases and quality reduction modules (QRM) simulating different fragmentation characteristics. The result are benchmark trajectory databases.

The trajectory database TDB_1 is generated using purely synthetic trajectories with predefined characteristics ($FGM_{SY,DB1}$).

$$TDB_1 = \langle FGM_{SY,DB1}, \Theta[1, 0.2], \kappa_{DA}[0], ODM[\infty, 0], QRM_{BB}[5, 0, 5] \rangle \quad (2.3)$$

Regarding the fragmentation characteristics all trajectories are fragmented, whereas in each trajectory 20% of all points are deleted using the bunch-based deletion module QRM_{BB} with

$$\mu_{BD} = 5, \sigma_{BD} = 0, \tau_{Seg,min} = 5.$$

Moreover, there are no object division events within the database.

For trajectory database TDB_2 , the trajectories are also generated purely synthetic ($FGM_{SY,DB1}$).

$$TDB_2 = \langle FGM_{SY,DB1}, \Theta[1, 0.3], \kappa_{DA}[0], ODM[\infty, 0], QRM_{BB}[2, 0, 2] \rangle \quad (2.4)$$

For the fragmentation, 30% of all points within each trajectory are deleted using the QRM_{BB} with

$$\mu_{BD} = 2, \sigma_{BD} = 0, \tau_{Seg,min} = 2.$$

Abbreviation		Description
QRM_{BB}	μ_{BD}	Mean bunch length
	σ_{BD}	Standard deviation of bunch length
	$\tau_{Seg,min}$	Minimal segment length
QRM_{MB}	p_{11}	Probability to stay in state 1
	p_{22}	Probability to stay in state 2
	p_{12}	Probability for the transmission from state 1 to state 2
	p_{21}	Probability for the transmission from state 2 to state 1
	$P_{del,S1}$	Probability to delete a point if the state is 1
	$P_{del,S2}$	Probability to delete a point if the state is 2
	ODM	ψ_{MODR}
ψ_{SODR}		Standard deviation of the object division rate
κ_{DA}		Probability for fragmentation in case of division event
Θ	$P_{t,del}$	Percentage of trajectories that are fragmented
	P_{del}	Percentage of points to delete

Table 2.2: Parameters for the quality reduction modules.

The trajectory database TDB_3 is generated based on purely synthetic trajectories that contain division events each 20 time steps.

$$TDB_3 = \langle FGM_{SY,DB1}, \Theta[0.8, 0.4], \kappa_{DA}[0], ODM[20, 0], QRM_{BB}[1, 0, 1] \rangle \quad (2.5)$$

Here only 80% of all trajectories are chosen to be fragmented by deleting 40% of all points using the bunch based deletion module with parameters

$$\mu_{BD} = 1, \sigma_{BD} = 0, \tau_{Seg,min} = 1.$$

For the validation benchmarks used in Chapters 3 and 4 of this thesis, the proposed notation is applied.

2.4.2 Benchmarks Used in this thesis

For the detailed quantitative evaluation of the different approaches in Chapter 3 and 4 several trajectory benchmark databases are generated (Figure 2.11). Regarding the detailed requirement of the validation task, different spatio-temporal characteristics described in Table B.1 in more detail are simulated and a bunch of trajectory benchmark databases is generated. Figure 2.11 represents the results of the fulltrack generation modules. Here, benchmark `TDB_S1` (Figure 2.11A) simulates trajectories moving on a sphere surface and benchmark `TDB_S1` (Figure 2.11B) contain trajectories with parallel movement characteristics at the beginning and a subsequent slight turn around movement to the sides. A clear internalization characteristics to one or both sides at several time points is represented within the benchmarks `TDB_S3-TDB_S5` (Figure 2.11C-E). Moreover, short local periods of undirected movement are represented in the benchmark databases `TDB_S6` and `TDB_S7` (Figure 2.11F-G). A combination of internalization movement and periods of undirected movement is contained within the benchmark databases `TDB_S8 - TDB_S11` (Figure 2.11H-K). Within Chapter 3 and 4, the required fragmentation characteristic representing the different problem classes (Section 2.1) is additionally added by setting the parameters for the different quality reduction modules (Section 2.3.2).

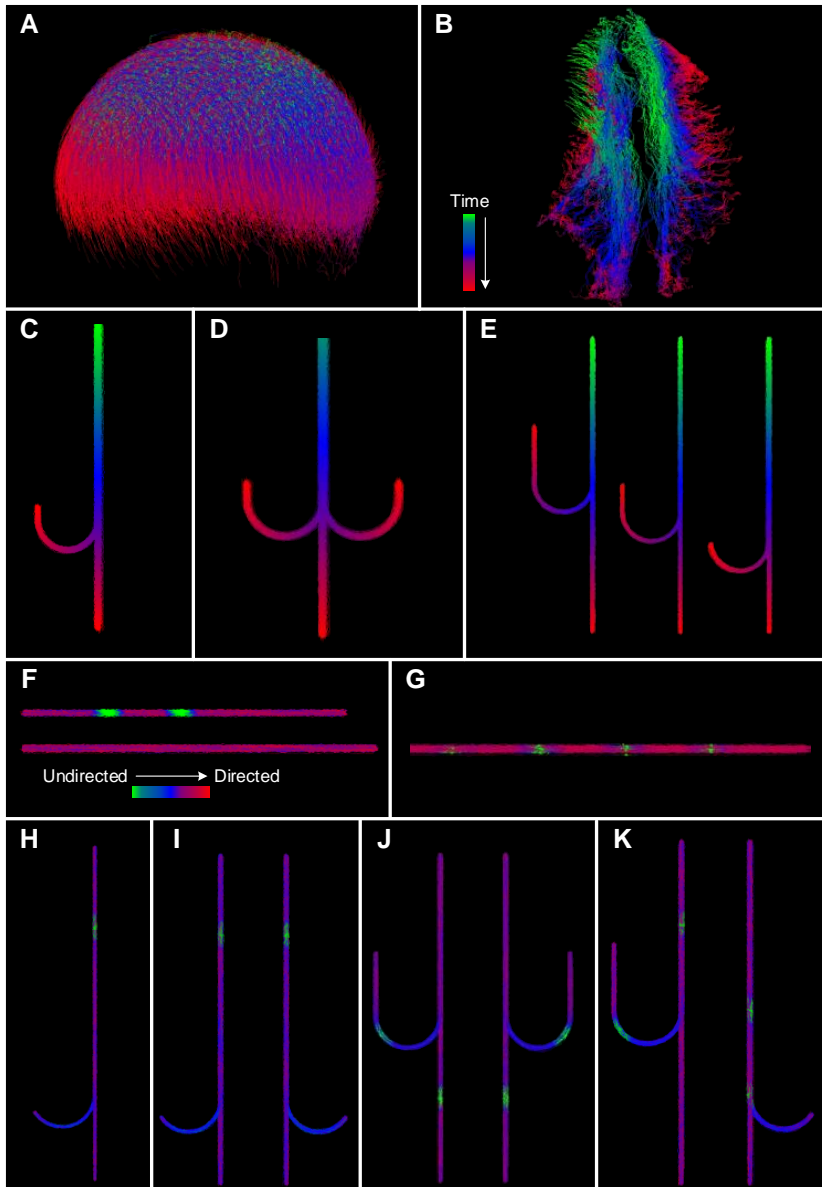


Figure 2.11: Generated trajectory benchmark databases for validation. The color-code in (A)-(E) indicates time and in (F)-(K) the directionality of movement. According to the notation in Table B.1 and B.2 the assignment of the benchmarks in this figure is: (A)-TDB.S1, (B)-TDB.S2, (C)-TDB.S3, (D)-TDB.S4, (E)-TDB.S5, (F)-TDB.S6, (G)-TDB.S7, (H)-TDB.S8, (I)-TDB.S9, (J)-TDB.S10, (K)-TDB.S11.

3 Integrating Prior Knowledge in the Discovery of Large-Scale 2D+t and 3D+t Trajectory Databases

Many automatically generated tracking databases containing a humongous amount of trajectories that offer an extremely valuable resource for in-depth analysis of object dynamics when made accessible properly. Therefore, Chapter 3 presents a variety of analysis methods tailored for tracking applications starting with the description of a systematic scheme for the knowledge discovery process in large-scale 2D+t and 3D+t trajectory databases to efficiently gain knowledge and quantify emerging phenomena with the focus on fragmented tracking data (Section 3.1). Furthermore, the systematic integration of existing expert-specific prior knowledge (Section 3.2) in the knowledge discovery process is described in a step-wise approach for the application-tailored efficient incorporation in consisting trajectory databases. Additionally, the possibility to incorporate new prior knowledge (Section 3.2.2) using a variety of conceptual approaches (Section 3.3) replenish the overall capabilities of efficiently analyzing existing trajectory databases of problem classes with varying complexity (Section 2.1). In case of fragmented tracking data several approaches to handle the circumstance of non-complete trajectories are described in Section 3.3.2 in detail, reaching from the calculation of representative tracks up to the allocation process of residual track fragments. To investigate trajectory databases in detail, Section 3.3.3 depicts distinct approaches for the quantitative description of moving object behavior. Focusing on local trajectory characteristics is further sustained using the methods described in Section 3.3.4. For the feature-based extraction of pertinent groups of possible interest, Section 3.3.5 delineates different possibilities. Furthermore, the efficient transfer of complete analysis pipelines is depicted in Section 3.3.6, allowing to extract the same relevant

knowledge from different datasets (Section 3.3.7). Moreover, the systematic handling of dividing object characteristics is detailed in Section 3.3.8.

3.1 General Design of the Knowledge Discovery Process

The knowledge discovery process (KD-process) is an iterative and interactive process to efficiently gain knowledge out of databases [346]. The tremendous amount of arising datasets yields to the problem that such datasets can hardly be evaluated and analyzed manually. The general aim is to generate a knowledge database containing relevant information about the dataset to answer scientific questions. Therefore, the KD-process is designed iteratively to access the information. Hereby, prior knowledge often exists at different levels of detail. The step-wise approach of the KD-process leads to an efficient analysis of the datasets and an optimal integration of the application-tailored prior knowledge.

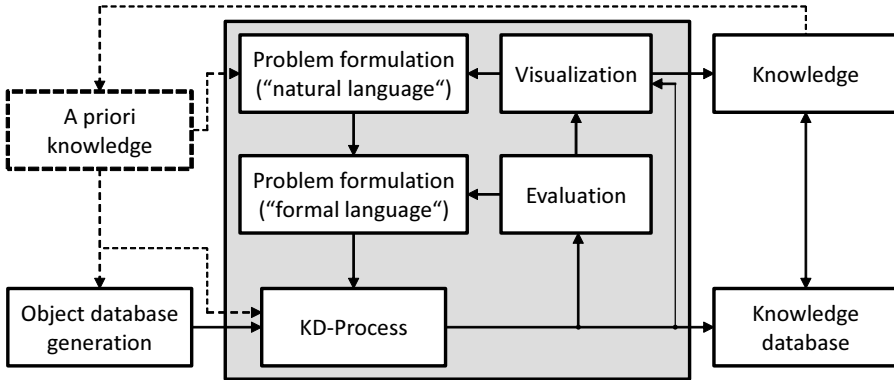


Figure 3.1: Scheme for the design of a knowledge discovery process. Here, the effective integration of prior knowledge to create a knowledge database is shown [346].

In case of applying the KD-process to tracking data of various fields, the database consists of hundreds of thousands to millions of objects that are tracked over time. Such highly complex problem classes can, for example, be

found in the field of developmental biology where the tracking of thousands of cells is a frequent analysis task [110, 141, 283, 347]. To gain knowledge out of this whole bunch of trajectories, the KD-process offers an opportunity to efficiently investigate the tracking database and integrate prior knowledge of experts to understand the behavior of objects and their accompanying interactions. To analyze application-specific tracking data (e.g., biological tracking data), the only change required for the KD-process is extending the available trajectory features to describe the dataset-specific trajectory characteristics. The complete architecture of the KD-process, as well as the computational analysis methods, remain the same, allowing to unfold a wide range of scientific questions in various application fields using the highly flexible and modular structure of the KD-Process. Several standard methods of the KD-process for extracting information out of the tracking data can be used (e.g., clustering, interactive filtering and classification). On this basis, the whole tracking databases can be iteratively investigated using the full power of the KD-process [346]. For instance, the KD process can be used to extract a spatial region of interest, to determine the time window of a specific process or to analyze a characteristic movement behavior of a group of objects by exploiting available prior knowledge and the intuition of the user for successive region of interest refinement.

Object Database Generation

The object database (Figure 3.1) generally contains the data of all included objects and serves as a basis within the KD-process [346]. The aim of the KD-process is to efficiently generate an interpretable knowledge database out of the original object database. According to the specific application of the KD-process, many different approaches to create the object database are possible reaching from raw object data to data-independent artificial simulated benchmarks, to name just a few. In case of spatio-temporal tracking data, the object database contains a set of trajectories \mathcal{T}_{set} . Chapter 2 therefore depicts a systematic overview of several fulltrack generation modules (*FGMs*) to serve as a basis for the generation of configurable object databases. The object databases used for validation purpose in this thesis are introduced in Section 2.4.2. Moreover, in Chapter 7 the analysis of real object databases is presented.

Generation of Problem Formulation

To apply the KD-process to the object database, the first step is to derive an applicable problem formulation in natural language as an important prerequisite for the extraction of meaningful knowledge out of an object database. For this task often an expert in the field is required to identify an appropriate problem formulation. Exemplary problem formulations could be "Find trajectories occurring in a given region", "Quantify spatio-temporal differences in the proportion of given trajectories" or "Find rare trajectories to detect occurring anomalies". Afterwards, the description in a natural language is transferred into a more formal formulation to be applicable for processing in mathematical algorithms and analysis pipelines.

Evaluation and Visualization

The evaluation and visualization modules embedded in the scheme of the knowledge discovery process in Figure 3.1 serve as a feedback to appropriately interact and optimize the output of the process. For the quantification of the recorded results, the evaluation module offers the possibility for human analysts to get quantitative descriptions of extracted groups and precisely describe specific characteristics in detail (Section 3.3.3). Therefore, extracted groups of objects can be investigated on the basis of their spatial, temporal or spatio-temporal characteristics. The results hereby serve as a basis to derive quantitative statements about specific object behavior and to compare several groups with the focus on predefined object characteristics (Section 3.3.7). The visualization module (Figure 3.2) complements the ability to interact with the KD-Process. Here, several visual representations support the human perception process and are enormous important for the reasoning process as well (Section 4.1 - 4.3). The visualization abilities serves as an essential basis to assist the human analyst in understanding the object database on a detailed level [346] (see Chapter 4 for a detailed description of guided visualization possibilities to support human perception).

3.1.1 Information Allocation Process

The KD-process module, depicted in the scheme in Figure 3.1, consists of an information allocation process module and the ability to efficiently incorporate interactive modifications during the knowledge discovery process (Figure 3.2). Each step within the overall framework (Figure 3.1) can consist of several iterations of interactive modifications and information allocation to refine the result. The information allocation process assigns specific information such as group membership or feature characteristics to each object within the object database. The allocation process can be done purely automatically or by manual assignment. Different methods for the allocation process provide high flexibility in target-based knowledge discovery. The three basic methods to allocate information to objects are cluster analysis, interactive filtering or manual allocation of specific characteristics. The cluster analysis approach forms groups of objects based on spatial, temporal, spatio-temporal or other quantitative features. Here, the allocation process assigns each object the membership to a specific group.

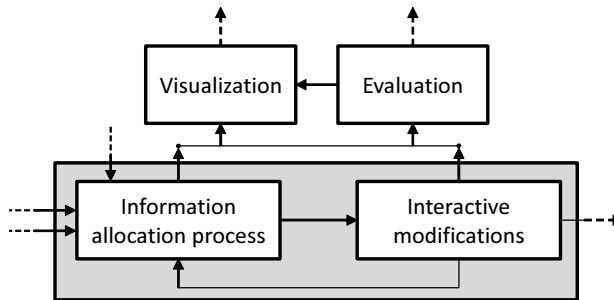


Figure 3.2: Scheme for the KD-process embedded within the general knowledge discovery framework in Figure 3.1 [346].

In cases where prior knowledge about features describing specific behavior is imprecise and object characteristics cannot be used adequately, clustering approaches for information allocation may not be appropriate [346]. Such circumstances often emerge on a detailed single object level with low occurrence of existing prior knowledge where the characteristics of single moving object is depicted. In such cases, there is the possibility to directly assign the informa-

tion such as group or cluster membership to the objects manually. The direct manual refinement of results (Section 4.2) is an appropriate approach suitable for highly detailed object levels containing only a few objects [346]. Interactive filtering, as a further method for the information allocation process, allows to apply several filtering methods to the object data reaching from spatial filtering by selecting a region of interest, temporal filtering by the selection of a time interval or characteristic-based filtering where object-specific characteristics are used to group the objects. The filtering, the clustering as well as the manual refinement can be flexibly applied in arbitrary order in different steps of the KD-process, allowing an efficient and straight forward extraction of underlying knowledge [346].

3.1.2 Interactive Modifications

To iteratively optimize the allocated information, the interactive modification module (Figure 3.2) provides several possibilities to efficiently incorporate prior knowledge (Section 3.2.2). In cases where the result of the information allocation module is not satisfying, the human analyst has the possibility of focused refinement. In the context of clustering the interactive modifications are often used to provide the user the ability to exclude objects associated with one or several subclusters, to fuse existing clusters or to apply a further clustering. The interactive modifications allow to effectively merge subclusters to one larger cluster or split existing subclusters by spreading their members among the remaining subclusters (for a detailed overview of newly developed interactive modification possibilities, see Chapter 4). Appropriate visualization techniques are required to support the whole interactive modification process and to facilitate interactions with the information allocation results to optimally apply the KD-process on existing object databases [346].

3.2 Systematization of Prior Knowledge

The incorporation of existing prior knowledge is an important element of the knowledge discovery process of trajectory databases in a variety of applications [146, 160]. In several knowledge discovery algorithms (Section 3.1) prior

knowledge is enormously helpful to improve and adjust the outcome of trajectory clustering algorithms (Section 3.3.5) and to extract adapted trajectory patterns on the basis of predefined movement dynamics. Especially, for the detailed quantification out of a tremendous amount of 2D+t and respectively 3D+t trajectory databases, prior knowledge is often necessary to refine intermediate results before applying subsequent analysis pipelines (Section 3.3.6).

3.2.1 Systematization of Different Sources of Prior Knowledge

Prior knowledge can be obtained from several resources reaching from expert knowledge, existing knowledge databases or from literature and published studies in the corresponding field. Often, a combination of these sources of prior knowledge is prevalent in application-specific knowledge discovery task. Additionally, the expert knowledge from different entities can be complementary as long as there is a common sense of data comprehension. In Table 3.1 an exemplary overview of different sources of prior knowledge in the context of object tracking and the analysis of the resulting trajectory data is depicted.

Here, the listed sources of prior knowledge provide a rough overview of potentially helpful characteristics to describe emergent phenomena in large-scale tracking data and have to be diligently chosen according the underlying problem class (Section 2.1) and the analysis task (Section 3.1).

3.2.2 Efficient Incorporation of New Prior Knowledge

The incorporation of new and additional prior knowledge from experts or from new experimental results is an important task to ensure flexibility and adaptability in the analysis of highly complex trajectory databases (Section 3.1). New prior knowledge with regard to the characteristic description of object movement behavior and global trajectory dynamics can be efficiently incorporated using the modular structure of the software framework (Section 5.1) with predefined interfaces. Here, new feature descriptions of single objects at given time points or the global spatio-temporal dynamics of the objects can be integrated without being an expert in programming the framework. A detailed schedule is depicted in Chapter 5. However, the incorporation of new prior

Prior knowledge	Description
Acquisition setting	Prior knowledge describing acquisition-specific influence factors such as image resolution, detection paths or illumination settings.
Spatial information	Information of the localization of the objects for each single object and for entire groups of objects defining neighborhood relations.
Interaction	The interaction of objects for predefined spatio-temporal conditions describing characteristic behaviors such as adhesion, division, clustering or density changes within the neighborhood.
Object morphology	Properties describing object characteristics such as volume, size, principal components and bounding volumes.
Geometry	Characteristics describing geometrical properties such as shape, dimensionality, proportions, feature localization within an object or symmetry specifications.
Dynamic Coherence	Object characteristic that change dynamically like speed, object expansion, direction of movement and the appearance and disappearance of objects.
Global dynamics	Object properties describing the global dynamics of a moving object's path, such as, the start or end region of objects as well as intermediate crossing regions.
Object progression	Spatio-temporal information about the objects and their movement pattern such as directional changes, resting points or phases of undirected movement.

Table 3.1: Identification of suitable prior knowledge. Extended from [84].

knowledge cannot be done in a fully automatic way. The user input is necessary to manually implement the new characteristic within the interface. In contrast, new prior knowledge based on existing intermodal visual representations can be easily integrated using the functionality of multi-linked visual representations introduced in Section 4.2.2 with the online propagation of relevant selections. Furthermore, the interactive access of the visual representation of the data often leads to new knowledge about the underlying processes and phenomena, such as group movement characteristics, allowing an ongoing deepening of the understanding of the present trajectory database. If new prior

knowledge is additionally based on new visual representations not integrated within the framework so far, there is the ability of incorporating new visualization instances that are automatically connected with all existing visualization windows to ensure multi-selection propagation (Section 5.1).

3.3 Conceptual Approaches to Analyze Large-Scale 3D+t Trajectory Databases

3.3.1 Overview of Conceptual Approaches

To efficiently analyze large-scale 2D and 3D trajectory databases and to gain quantitative knowledge (Section 3.1), several conceptual approaches are introduced in this Section. Hereby, the methodological approaches (MA) introduced in Section 3.3.2 - 3.3.9 depict automatic approaches to computationally analyze trajectory databases and extract quantifiable feature descriptions of the underlying trajectories (Figure 3.3).

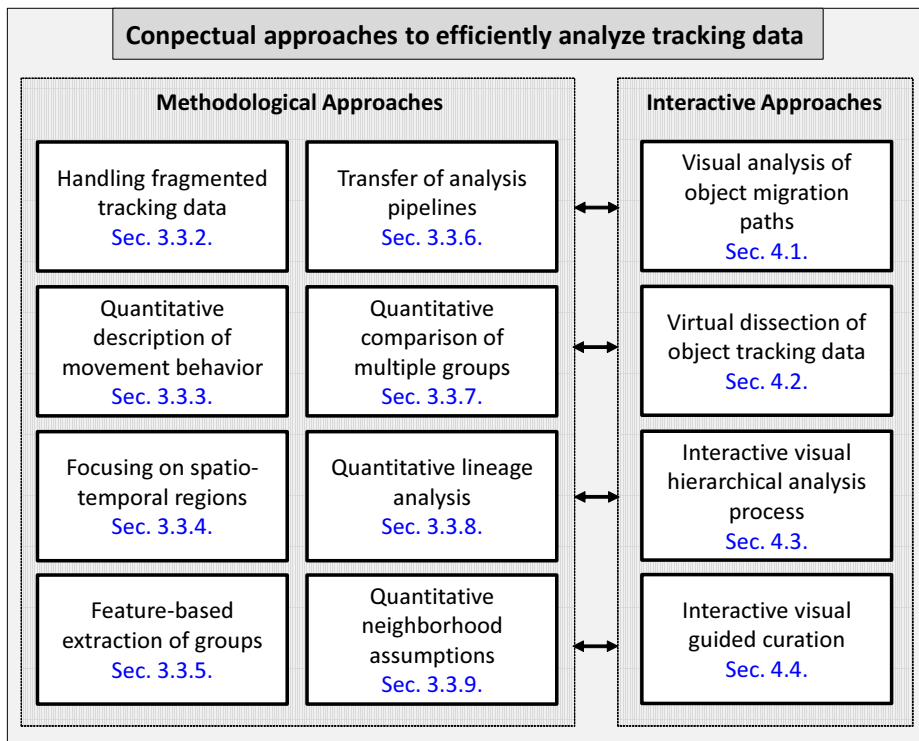


Figure 3.3: Overview of conceptual approaches to analyze large-scale trajectory databases. The methodological approaches introduced in Chapter 3 are combined with the interactive approaches developed in Chapter 4 to efficiently analyze existing trajectory databases.

In contrast to the methodological approaches, the interactive approaches (IA), delineated in Section 4.1 - 4.4, allow the efficient incorporation of existing prior knowledge from experts and to interactively guide the computational routines. Both, the methodological and interactive approaches in combination reveal full power of interactive guided computational analysis of huge trajectory databases. Regarding the application-specific purpose of analysis, the conceptual approaches cover the whole range of problem classes (Table 3.2) reaching from low complexity (Problem class 01) up to tremendous complexity levels (Problem class 20). In Table 2.1, the problem classes are described in detail. Each conceptual approach in Table 3.2 can be used in a stand-alone fashion or in combination with all other conceptual approaches providing enormous flexibility in addressing exceedingly application-specific analysis tasks. Exemplary the conceptual approach MA_{QDMB} handling the description of trajectory features can be applied to all twenty problem classes as shown in Table 3.2. The conceptual approach MA_{QLA} on the contrary is handling the division characteristics of moving objects making the application only meaningful to the problem classes containing dividing objects. The validation of these highly application-specific analysis tasks is in most cases only possible in a quantitative manner. Due to the enormous variations within the application-specific tasks, the validation result of one specific example is unlikely to be transferred to other scenarios.

3.3.2 Handling Fragmented Object Tracking Data

General Possibilities to Handle Fragmented Trajectory Data

Existing prior knowledge in the context of trajectory databases often pertains to trajectories reaching nearly over the complete time interval of consideration (Section 3.2). However, in highly complex tracking problem classes (Section 2.1) it is often not possible to continuously track the objects, inevitably leading to the fragmentation of the trajectories (Section 1.2.1). The fragmentation of the trajectories leads to a variety of problems reaching from the specification of time series feature calculated on the whole spatio-temporal course of the trajectory up to the fact that global statements on the individual object are not possible. Furthermore, group assignments based on global move-

Conceptual approaches	Description	Sec.	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20
MA _{HFD}	Handling fragmented tracking data	3.3.2			x		x		x		x	x	x	x	x	x	x	x	x	x	x	x
MA _{QDMB}	Quantitative description of movement behavior	3.3.3	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
MA _{HSTR}	Focusing on spatio-temporal regions	3.3.4	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
MA _{FBEG}	Feature-based extraction of groups	3.3.5	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
MA _{TAP}	Transfer of analysis pipelines	3.3.6	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
MA _{QCMG}	Quantitative comparison of multiple groups	3.3.7	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
MA _{QLA}	Quantitative lineage analysis	3.3.8							x	x	x	x	x	x	x	x	x	x	x	x	x	x
MA _{QNA}	Quantitative neighborhood assumptions	3.3.9	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
IA _{VAMP}	Visual analysis of object migration paths	4.1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
IA _{VDTD}	Visual dissection of object tracking data	4.2	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
IA _{VHAP}	Interactive visual hierarchical analysis process	4.3	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
IA _{VCC}	Interactive visual guided curation	4.4			x		x		x		x	x	x	x	x	x	x	x	x	x	x	x

Table 3.2: Application of the different conceptual approaches to problem classes 01-20 represented by the columns numbered from 01 to 20. A cross indicates that the given conceptual approach is reasonable for the given problem class.

ment characteristics can hardly be done in case of highly fragmented trajectory data. Therefore, approaches to handle fragmented tracking data are required to overcome the incisions in the analysis (Section 3.2). One approach purely takes only non-fragmented trajectories for subsequent analysis tasks requiring no processing steps, but lacking of interpretable results due to the elimination of a considerable proportion of trajectories. Another approach takes all trajectories into consideration for further investigation no matter if these trajectories are fragmented or not. Incorporating all trajectories without distinguishing the integrity of the data does not require any pre-processing steps but fails to extract reliable results of extracted trajectory features and the subsequent analysis (Section 3.3.3). Furthermore, the approach of forced fusion of fragmented trajectory using a quality measure (QM) to link the trajectory fragments leads to trajectories spreading a longer time interval but at the same time loses object maintenance by applying potentially incorrect linking decisions in all of the remaining cases. The quality measure is used to link the local nearest neighbors at each of the start points of a fragmented trajectory e.g., positions where a trajectory fragment was not successfully linked so far [348]. Here, two different neighborhoods (\mathcal{N}), namely \mathcal{N}_{All} and \mathcal{N}_{EP} , serve as a basis for the nearest neighbor search. The neighborhood \mathcal{N}_{All} contains all points of all trajectories for the nearest neighbor search whereas the neighborhood \mathcal{N}_{EP} contain only the end points of all trajectories in $\mathcal{T}_{set} = \{T_1, T_2, \dots, T_{N_T}\}$ for the search of the nearest neighbors.

One derivative of this quality measure, namely the QM_{EP} , ensures that only end points of possible trajectory fragments are considered for local neighbor search using the neighborhood \mathcal{N}_{EP} , leading to a more robust linking with the drawback not to be able to handle dividing object characteristics (see Section 2.1). Therefore, the nearest neighbor (NB) points \mathbf{p}_x to the actual start point of trajectory fragment \mathbf{p}_{act} are calculated:

$$NBs(\mathbf{p}_{act}) = \operatorname{argmin}_{\mathbf{p}_x \in \mathcal{N}_{EP}(\mathbf{p}_{act})} \sqrt{(\mathbf{p}_x - \mathbf{p}_{act})^2} \quad (3.1)$$

To also access problem classes comprising dividing objects, all points (in \mathcal{N}_{All}) not only the end points of the local nearest trajectories are used in the quality measure QM_{AP} to find possible predecessor trajectories. Hereby, the nearest

neighbor (NBs) points \mathbf{p}_x to the actual start points of the trajectory fragment \mathbf{p}_{act} are calculated:

$$NBs(\mathbf{p}_{act}) = \operatorname{argmin}_{\mathbf{p}_x \in \mathcal{N}_{All}(\mathbf{p}_{act})} \sqrt{(\mathbf{p}_x - \mathbf{p}_{act})^2} \quad (3.2)$$

To evaluate the effect of the complexity of problem classes (Section 2.1) to the result of the forced fusion using the proposed quality measures QM_{AP} and QM_{EP} , the percentage of deletion as well as the noise levels are simulated using benchmark database TDB_S3 (Section 2.4.2) to depict the degree of complexity. An increase in the percentage of deletion and noise level yields more complex tracking problems. The effect of the growing percentage of deletion is depicted in Figure 3.4 for smaller gaps and in Figure C.1 for even longer gaps.

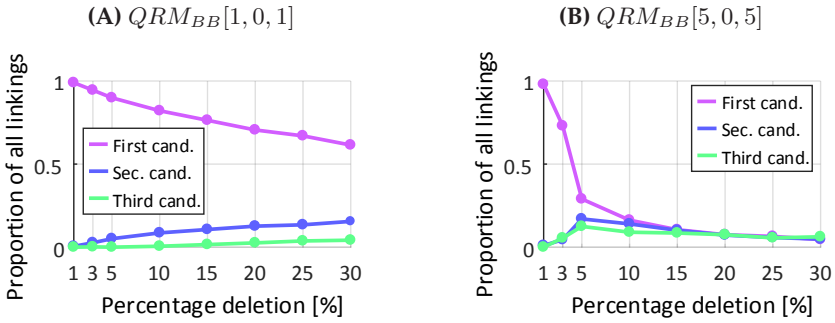


Figure 3.4: Impact of fragmentation to the linking quality using the quality measure QM_{EP} to rank the linking candidates. The first (magenta), second (blue) and third (green) candidate that are proposed by the quality measure QM_{EP} are shown. The percentage of overall correct linking decisions is hereby represented by the first candidate. There is no noise added to the trajectory positions. (A) For the fragmentation, a bunch deletion with a mean bunch length of 1 is used leading to short gaps within the trajectories. For different percentages of deletion, which describe the percentage of deleted points within each trajectory, the proportion of correct linkings is shown. (B) Here, a mean bunch length of 5 is used resulting in longer gaps leading to a fast decrease of the proportion of the correct linking candidates.

Here, the three best proposed candidate from the quality measure QM_{EP} are illustrated with the first candidate representing the correct linking of the forced fusion. For small deleted bunches with $QRM_{BB}[1, 0, 1]$ (Section 2.3.2), there is

an ongoing decrease of the correct linkings with an increased deletion percentage (Figure 3.4A). Moreover, a steep decrease of the correct linking percentage can be observed the larger the deleted gaps gets within the trajectories (Figure 3.4B, C.1A, B). Thus, a strong decrease of the linking quality is observed as the complexity of the problem class increases. The decrease of the linking quality is even more prominent when using the QM_{AP} quality measure (Figure C.2). The quality measure QM_{AP} has no information when objects are dividing and therefore always link to the nearest neighbor. The benchmark `TDB_S3` that is used in the validation does only contain single moving objects that do not divide and therefore do not increase the number of total objects contained in the dataset. However, the quality measure QM_{AP} contain the assumption of division events leading to an even worse linking result. Exemplary Figure C.4 shows, that an increasing in the standard deviation of the bunch lengths lead to similar curve characteristics with a vanishing between different mean bunch lengths. To systematically evaluate the effect of the noise within the datasets, different noise levels were simulated by adding Gaussian noise to the positions of the trajectories (Section 2.3) using two exemplary percentage of deletion (Figure 3.5A, B).

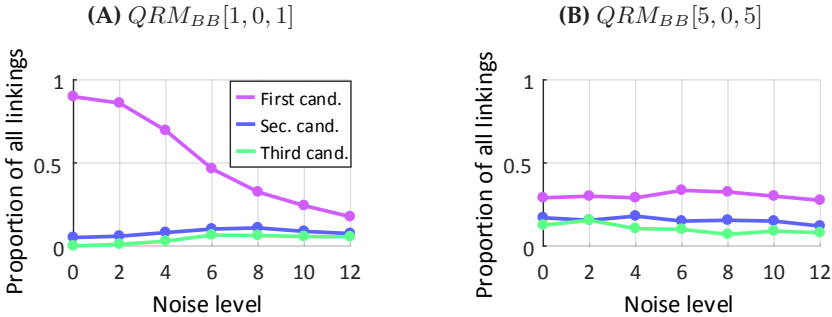


Figure 3.5: Impact of noise to the linking quality using the quality measure QM_{EP} . The first (magenta), second (blue) and third (green) candidate that are proposed by the quality measure QM_{EP} are shown. The percentage of overall correct linking decisions is hereby represented by the first candidate. Fragmentation with 5% of deleted points ($\Theta[1, 0.05]$). (A) For the fragmentation, a bunch deletion with a mean bunch length of 1 is used leading to short gaps within the trajectories. (B) Here, a mean bunch length of 5 is used resulting in longer gaps.

In case of short deleted gaps (Figure 3.5A, Section 2.3.2) the linking quality decreases strongly over the noise levels. Hereby, the combination of noise and deletion lead to a more precipitously decrease compared to a pure fragmentation without noise. If the deleted gaps (fragmentation of 5%) get bigger (Figure 3.5B, Figure C.3), the linking quality is reduced dramatically, leading to an invalid result of the forced fusion. With an additional increase in fragmentation to 10% (Figure C.5) the quality of linking also faster decrease, leading to nearly no valid linkings (Figure C.5C, D). The higher the underlying problem class represented by the level of fragmentation and noise, the worse the forced fusion approach to link the trajectory fragments.

Moreover there is one additional approach for fragmented tracking data described in the following. For a more detailed analysis of fragmented trajectory data, representative trajectories containing favored characteristics can be used for feature-based group extraction (Section 3.3.5). The remaining shorter trajectory fragments are then allocated to the predominant group within the spatial proximity of the neighborhood. A detailed description is depicted at the end of this section. Additionally, a fully error-free interactive curation of the fragmented trajectories leads to perfect trajectories spanning the whole time period leading to a perfect analysis of the trajectories but accompanying with an higher time effort (Section 4.4). The effect of different approaches to handle fragmented trajectory data on the result of analysis pipelines is depicted in Section 3.3.5 in detail.

Calculate Trajectory Neighborhood

To characterize trajectories according to their neighborhood, the nearest neighbors for each trajectory are calculated. The notation $\mathcal{T}_{\text{NBH}} = \{\text{NB}_1, \text{NB}_2, \dots, \text{NB}_{N_{\text{TNB}}}\}$ indicates a trajectory neighborhood consisting of N_{TNB} nearest neighbors NB. Superscript indices of the nearest neighbors NB further indicates the group membership of the actual neighbor, e.g. NB_1^{G1} points out that neighbor NB_1 belongs to Group1 ($G1$). For the calculation of these neighbor trajectories, a distance metric is used. Depending on this distance metric, neighbor trajectories are trajectories close in time, space or both. To calculate the neighborhood in a pure spatial context ($\mathcal{T}_{\text{NBH}}^{\text{SP}}$) trajectories located near in space are considered for the neighborhood (Figure 3.6). A sphere, in the form of a globe with radius r , is created around each trajectory. All trajectories that

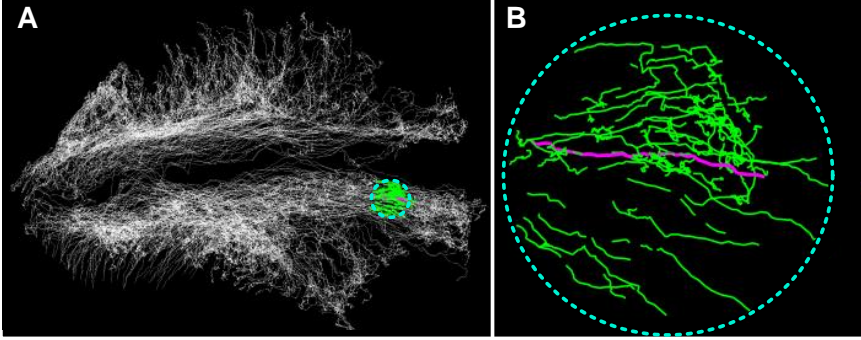


Figure 3.6: Neighborhood calculation for large-scale trajectory databases. Here, the simulated benchmark TDB_S2 is used to exemplarily show the calculation of a neighborhood for a selected trajectory. For the selected trajectory (magenta) the spatial neighborhood (green) is visualized. (A) The whole database with the marked region. (B) The corresponding zoomed region where the neighbors (green) for the selected trajectory (magenta) can be seen in more detail.

contain points that lie within that sphere belong to the neighborhood of the considered trajectory. The parameter r describing the sphere with the centroid coordinates x_c, y_c, z_c determines the neighborhood and can be adapted to specific analysis approaches interactively. Here, the spatial neighborhood for a trajectory \mathcal{T}_i is defined in the way:

$$\mathcal{T}_{NBH}^{SP} := \{ \mathcal{T}_i \in \mathcal{T}_{set} : \min_k (\text{dist}(\mathcal{T}_i[k], (x_c, y_c, z_c)^T)) < r \} \quad (3.3)$$

with

$$\text{dist}(\mathcal{T}_i[k], (x_c, y_c, z_c)^T) := \sqrt{(x_k^i - x_c)^2 + (y_k^i - y_c)^2 + (z_k^i - z_c)^2}. \quad (3.4)$$

To calculate the neighborhood in a combined spatio-temporal context (\mathcal{T}_{NBH}^{ST}), trajectories occurring at the same time within a spatial region around a trajectory are considered to be the neighbors (Figure 3.8). For this purpose, the mean Euclidean distance is calculated between all points of a considered trajectory and all points of the remaining trajectories. To define the neighborhood, the N_k nearest neighbors have to be selected. The parameter N_k can be chosen freely

to adapt to the specific approach. Here, Figure 3.7 depict the different cases for the calculation of the neighborhood ($\mathcal{T}_{\text{NBH}}^{\text{ST}}$), exemplarily on two trajectories \mathcal{T}_i and \mathcal{T}_j . In Case I, there is no neighborhood relation between \mathcal{T}_i and \mathcal{T}_j due to

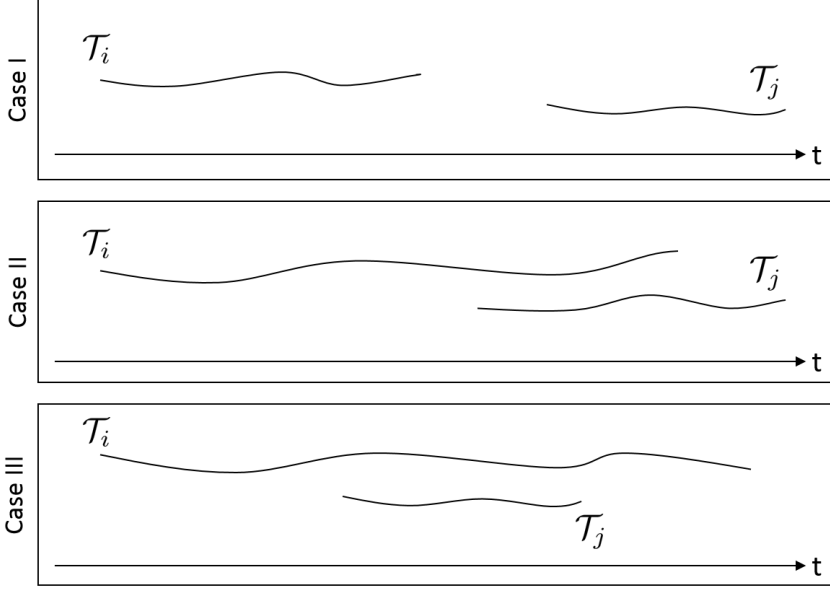


Figure 3.7: Distinction of cases for the calculation of the spatio-temporal neighborhood including three main cases from which all other can be derived. Here, only the time matters for the consideration of the cases. In Case I, the trajectories have no common time period in which both trajectories exist. Whereas, in Case II the trajectories \mathcal{T}_i and \mathcal{T}_j overlap. In Case III, one trajectory \mathcal{T}_j is completely contained within the time period of the other trajectory \mathcal{T}_i .

no overlap. For Case II, the mean distances $d_{\mathcal{T}_i, \mathcal{T}_j}$ between the two trajectories \mathcal{T}_i and \mathcal{T}_j is calculated as follows:

$$k_{\text{start}} = \underset{k}{\operatorname{argmin}} (|t_1^{\mathcal{T}_j} - t_k^{\mathcal{T}_i}|) \quad (3.5)$$

$$d_{\mathcal{T}_i, \mathcal{T}_j} = \frac{\sum_{k=k_{\text{start}}}^{k_{\text{NP}}} \left\| (\mathbf{p}_k^{\mathcal{T}_i} - \mathbf{p}_{(k-k_{\text{start}}+1)}^{\mathcal{T}_j}) \right\|}{k_{\text{NP}} - k_{\text{start}}} \quad (3.6)$$

For Case III, the calculation of the distances is:

$$k_{\text{start}} = \underset{k}{\operatorname{argmin}} (|t_1^{\mathcal{T}_j} - t_k^{\mathcal{T}_i}|) \quad (3.7)$$

$$k_{\text{end}} = \underset{k}{\operatorname{argmin}} (|t_{\text{NP}}^{\mathcal{T}_j} - t_k^{\mathcal{T}_i}|) \quad (3.8)$$

$$d_{\mathcal{T}_i, \mathcal{T}_j} = \frac{\sum_{k=k_{\text{start}}}^{k_{\text{end}}} \left\| (\mathbf{p}_k^{\mathcal{T}_i} - \mathbf{p}_{(k-k_{\text{start}}+1)}^{\mathcal{T}_j}) \right\|}{k_{\text{end}} - k_{\text{start}}} \quad (3.9)$$

The calculated distances between an actual trajectory and all other trajectories is saved in a list $D_{\mathcal{T}_i}$ containing all distances. This list is sorted in an ascending order and the N_k nearest trajectories are selected for the neighborhood. Further, the neighborhood of a trajectory can additionally be used to make statements about single trajectories in the context of its neighbors. One important scope of application is to make quantitative statements about the homogeneity of single trajectories in the context of their neighborhood according a given characteristic (Section 3.3.9). Not only trajectories themselves can be characterized by features, further valuable information can be derived about a trajectory in the context of its neighborhood. Analyzing a trajectory in the context of its neighbors for example may be further suitable to assess the occurrence frequency of extracted groups within the neighborhood for auto-assignment of short, fragmented trajectories. This leads to an additional possibility to analyze the tracking data.

Select Representative Trajectories

Often, not all tracks can be used for an analysis task due to artifacts or noise within the data. For this reason, representative tracks, containing the favored characteristic, are selected with regard to the subsequent analysis tasks. Depending on the analysis task, the selection of representative trajectories varies. To extract the representative trajectories, each trajectory is associated with a

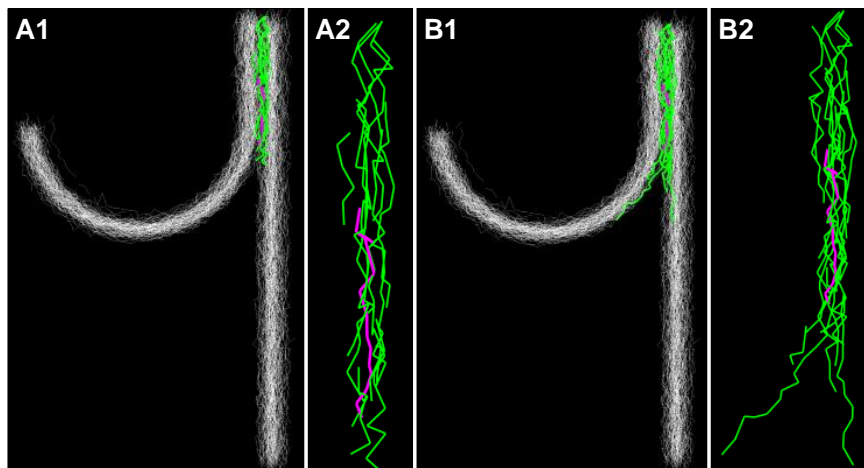


Figure 3.8: The effect of the size of the neighborhood N . Here, the simulated benchmark database TDB_S3 is used. Different parameter examples for spatio-temporal neighborhoods (green) of a single trajectory (magenta) are shown. The white trajectories thereby are not considered for the neighborhood. (A1) The $N=10$ nearest neighbors in global context and a zoomed region in (A2) respectively. (B1) The $N=20$ nearest neighbors and the corresponding zoomed region (B2).

set of predefined spatio-temporal characteristics like the length of a trajectory. Only trajectories that match the predefined criteria are chosen to be representative and are selected for further processing. It is also possible to initially correct a representative amount of tracks manually by linking subsequent fragments of a track (Section 4.4). Additionally, the use of automatic curation routines with predefined thresholds can also generate a set of representative trajectories. The following analysis pipeline is then unfailingly applied to the bunch of representative trajectories. In Figure 3.9, different percentages of representative tracks ($Q_{Alloc, PRT}$) are shown. Here, it can be observed that a decreasing percentage of representative tracks affects the problem of under-representation of groups (Figure 3.9A4, B4). This effect is more severe in cases where the amount of objects is not equally distributed over all different groups and in cases of fragmented tracks leading to leaving out minor groups in subsequent analysis (Section 3.3.5).

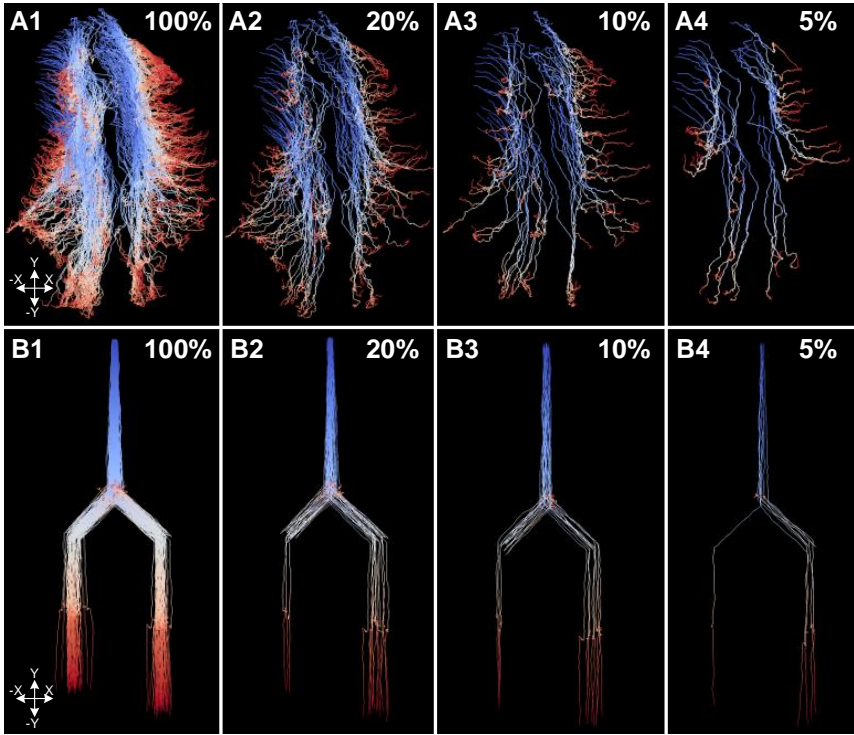


Figure 3.9: The effect of the choice of representative trajectories. For two benchmark databases (TDB_S2 show in A, TDB_S12 show in B) different percentages of representative tracks are illustrated reaching from $(Q_{Alloc,PRT}) = 100\%$ (A1,B1), $(Q_{Alloc,PRT}) = 20\%$ (A2,B2), 10% (A3,B3) and respectively $(Q_{Alloc,PRT}) = 5\%$ (A4,B4). Here, the percentage indicates the amount of trajectories within the benchmark database that are chosen to be used for a subsequent analysis task.

Allocation Process of Fragmented Trajectories

As an alternative to laborious manual correction of fragmented tracks, in this thesis an allocation process is developed, a strategy that can circumvent the problems arising with fragmented tracking data [346, 348]. Initially, a time interval of interest has to be defined and only tracks that sufficiently cover the desired time span are selected. Short fragments of the tracks are excluded from the initial analysis.

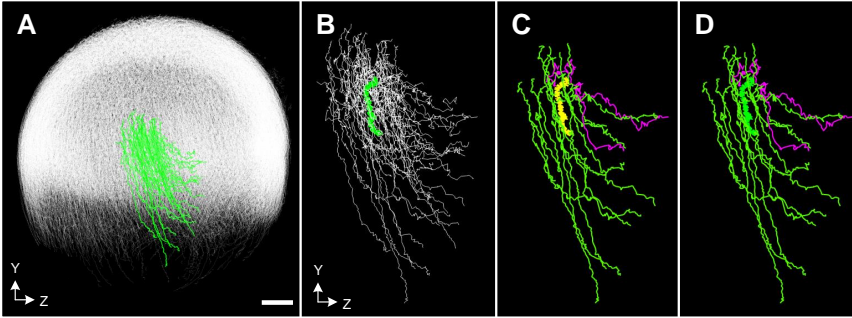


Figure 3.10: Allocation process as an alternative to manually correcting an entire tracking database. Here, the simulated benchmark database TDB_S1 is used (A). In (B) an exemplary selected short track fragment colored in green as well as other short track fragments and representative tracks (colored in white) are shown. First, only a subset of sufficiently long tracks is chosen to perform the region of interest selection of these tracks via filtering. Afterwards, all unlabeled tracks (exemplary one selected track colored in yellow) (C) are assigned to the predominant group of their spatio-temporally nearest neighbors (two groups exist colored in green and magenta) (D). In this exemplary case, the yellow unlabeled track fragment is assigned the predominant green group.

The basic idea of the allocation process is that some representative trajectories that are sufficiently long enough serve as representatives for the different groups contained in the dataset (Figure 3.10). These representative trajectories are used in the first analysis step to extract groups using all group selection strategies introduced in the KD-process (Section 3.1) to the subset of sufficiently long trajectories that allow analysis over the entire time interval of interest (Section 3.1). Afterwards, all remaining unlabeled track fragments (yellow track in Figure 3.10E) are allocated to the identified groups of interest. For the allocation process, each trajectory is classified based on the predominant group among its spatio-temporally nearest neighbors (NB). Based on a non-parametric k-nearest neighbor classification approach, the unclassified track fragments are then assigned the predominant group of their respective k-nearest neighbors, i.e., each track fragment is associated with the most frequently occurring class in its spatio-temporal neighborhood (\mathcal{T}_{NBH}^{ST}).

It is also possible to combine manual curation effort with representative trajectories and the allocation process. Thereby, initially a representative amount

of trajectories is linked manually and is subsequently used to identify groups of interest and finally assign all remaining track fragments to the group of the spatiotemporally closest neighbors. While the allocation approach works reasonably well for spatially separated groups and provides valuable first insights into the data without the need for manual curation effort, the automatic allocation of fragmented tracks may be ambiguous in transitional regions and may require further manual refinement. A reasonable approach is to first start with manual curations at the interfaces between different groups and let the automatic methods work on regions that can be unambiguously assigned. Generally, there are several parameters contributing to the result of the allocation process, namely the number of neighbors ($Q_{Alloc,NN}$), the percentage of representative tracks ($Q_{Alloc,PRT}$), the percentage of the most common group membership within a given neighborhood ($Q_{Alloc,PMC}$), the fuzzy membership (FM) and the spatial distribution of the representative tracks within the existing groups. In the following, the impact of various parameter settings is validated using database `TDB_S3` (Section 2.4.2) to deduce the impact to the results of the allocation process in a quantitative manner. To simulate the spatial separability, the distance parameter (D_{BG}) between the two groups in database `TDB_S3` (Section 2.4.2) is introduced. Therefore, Figure 3.11A-C exemplary depicts three different distances between the straight moving and the internalizing group representing different spatial separability. A good spatial separation of the groups (Figure 3.11A), a border case with spatial touching groups (Figure 3.11B) and spatial overlapping groups (Figure 3.11C) are exemplary visualized. In the following, the notation `TDB_S3`[$D_{BG} = 20$] exemplary points out the database `TDB_S3` with a distance D_{BG} between the groups equal to 20. Further, superscript indices offer an additional possibility to set the values of the given parameters described in the subscript indices. A notation of $Q_{Alloc,NN}^{10}$ points out the number of nearest neighbors ($Q_{Alloc,NN}$) is equal to 10.

The fuzzy membership (FM_{G_i}) of a specific group G_i within a given neighborhood \mathcal{T}_{NBH} indicates the percentage of the nearest neighbors (NB) belonging to the specific group G_i . Exemplary, the neighborhood

$$\mathcal{T}_{NBH} = \{NB_1^{G1}, NB_2^{G1}, NB_3^{G2}, NB_4^{G1}\} \quad (3.10)$$

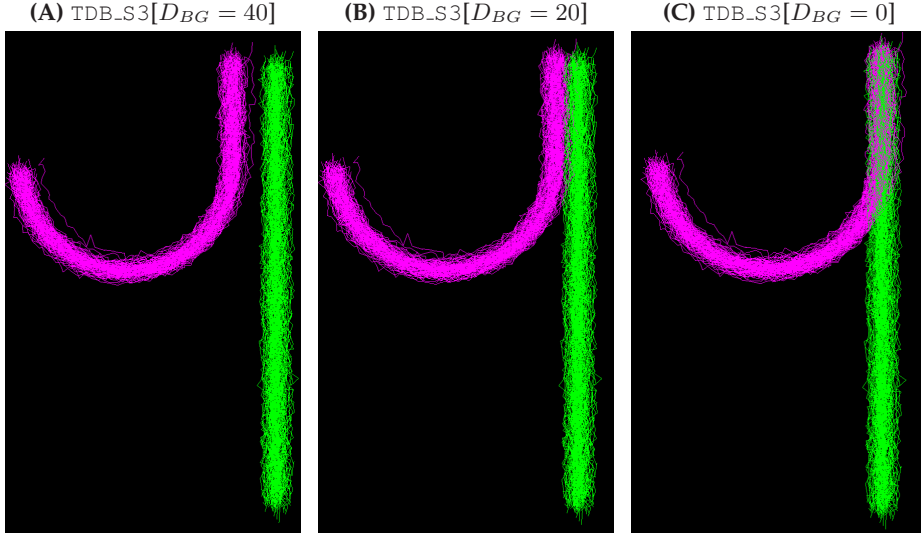


Figure 3.11: The impact of spatial separability in the allocation process using benchmark database TDB_S3. (A) Spatially well-separated groups. (B) Border case with spatially touching groups. (C) Spatially overlapping groups. For the benchmark fragmentation all trajectories are used with a deletion of 10% ($\Theta[1, 0.1]$, $\kappa_{DA}[0]$, $ODM[\infty, 0]$). Here, a bunch deletion module ($QRM_{BB}[1, 0, 1]$) with a mean bunch length of 1, leading to short gaps within the total $N_T = 1655$ trajectories is used.

contains four nearest neighbors. Three of the neighbors have a group membership of group G_1 and one neighbor has a group membership of group G_2 . In this case the fuzzy membership FM_{G_1} of group G_1 is 75%, represented by the notation $FM_{G_1}^{0.75}$. The fuzzy membership enables to make quantitative statements about the allocation process and to control the assignment of unlabeled track fragments. If the fuzzy membership is very high, the neighborhood of a selected trajectory is highly homogeneous regarding the given group membership resulting in a highly valid allocation result. Further, the fuzzy membership of a trajectory neighborhood can be used to assess border regions between different groups, allowing an efficient guidance of the manual curation process (Section 4.4) reducing the human effort. The impact of the fragmentation and the separability of groups to the fuzzy membership is evaluated for different parametrization of the bunch deletion module ($QRM_{BB}[1, 0, 1]$, $QRM_{BB}[5, 0, 5]$, $QRM_{BB}[10, 0, 10]$, Section 2.3.2) in Figure 3.12.

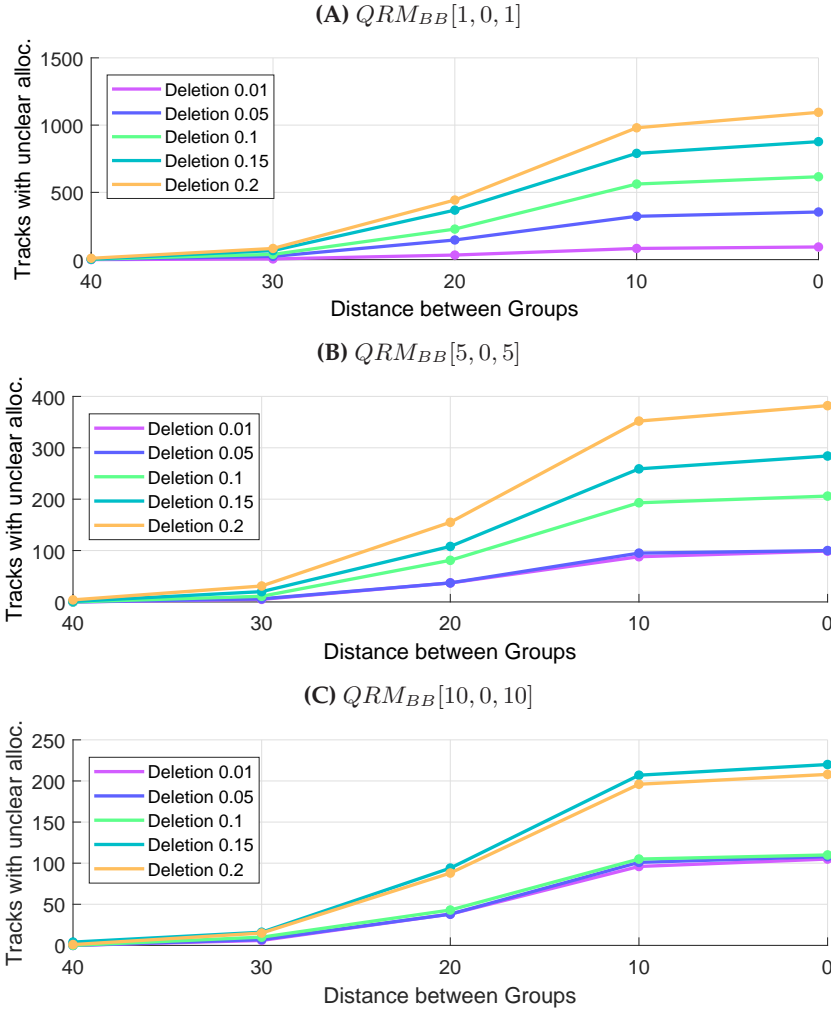


Figure 3.12: Fuzzy Membership in the allocation process. The distance between two groups is decreased step by step for different bunch deletion parametrization (A-C) with very short gaps (A), short gaps (B) and longer gaps of deletion (C). For each step the number of not 100 percent clear allocated tracks are counted. All trajectories are fragmented with different percentage of deletion ($\Theta[1, 0.1]$). Furthermore, there is no fragmentation through division events ($\kappa_{DA}[0, ODM[\infty, 0]]$). For the total number of $N_T = 1655$ trajectories the 50 nearest neighbors ($Q_{Alloc, NN} = 50$) are calculated.

Therefore, the number of tracks with unclear allocation (fuzzy membership less than 100%) is plotted against the distance between the groups (described in Fig-

ure 3.11). For all three bunch deletion parametrizations it can be observed that the number of unclear allocated tracks dramatically increases with the diminishing of the spatial separability of the groups. Also, the total number of unclear allocated tracks increase when the bunch lengths of the deletions within the fragments decrease. This effect is attributable to the fact that the smaller the bunch lengths are, the less the global characteristic of the tracks are represented.

Also, the percentage of representative tracks has a great influence on the outcome of the allocation process (Figure 3.13). Depending on the specific dataset, consisting of the separability, the spatial homogeneity and the amount and separation of existing groups, the effect of representative tracks on the allocation process varies. The allocation leads to very good results for spatially well separated groups (Figure 3.13A). Here, even small percentages of representative tracks ($Q_{Alloc,PRT}$) leads to perfect allocation results where all tracks are efficiently allocated the correct group. If the spatial separability of the groups decreases also the percentage of wrongly allocated tracks ($Q_{Alloc,PWAT}$) increases. In the case of no spatial separability (Figure 3.13C), a perfect allocation result as shown in Figure 3.13A is not even possible.

What can be derived from this result is, that the better the spatial separability of the groups within a database, the better the allocation process works, even for smaller amounts of representative tracks. Further, the additional influence of the combination of the number of nearest neighbors and the percentage of representative tracks is systematically investigated in Figure 3.13. For the exemplary databases `TDB.S3` (Section 2.4) it can be observed for spatial not separable groups (Figure 3.13C), that the higher the number of nearest neighbors within the neighborhood \mathcal{T}_{NBH} , the worse the allocation result. This is due to the fact that a greater neighborhood leads to the incorporation of tracks from adjoining groups and therefore leads to ambiguous allocation decisions. The better the spatial separability, the less the effect of the number of nearest neighbors. In case of spatially well separated groups (Figure 3.13A) different sizes of the neighborhood have no effect to the allocation result, except the greatest neighborhood containing 12 nearest neighbors increase the percentage of wrong allocated tracks. Summarized, the allocation leads to good results by roughly extract given groups of tracks. Once such rough groups are ex-

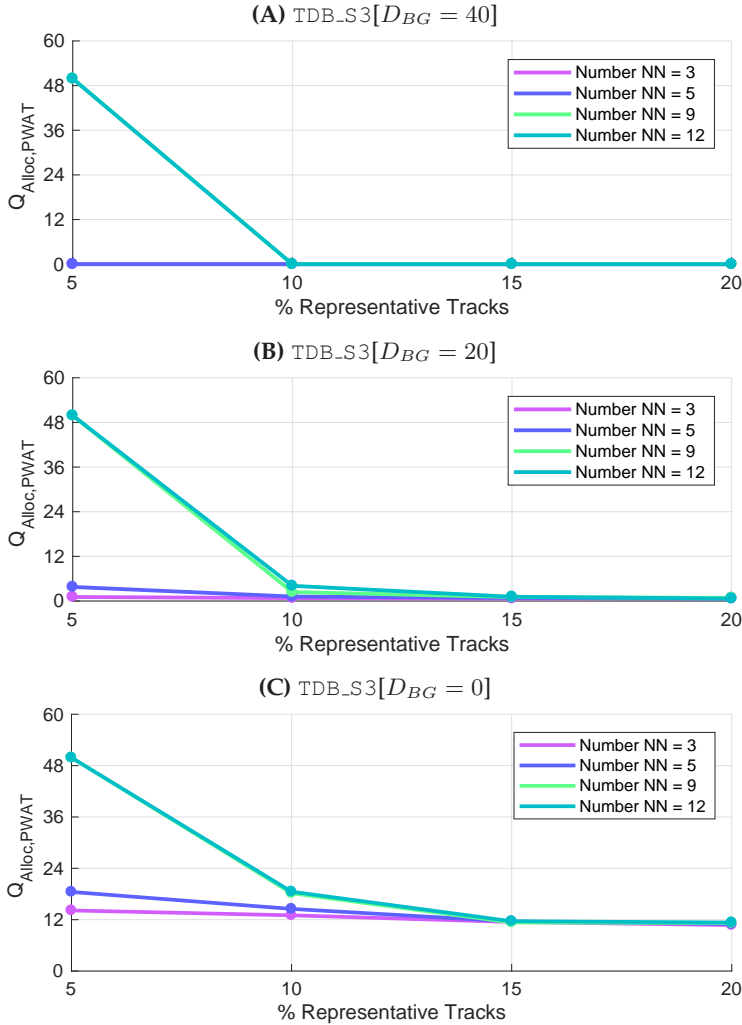


Figure 3.13: The percentage of wrong allocated tracks vs. the percentage of representative tracks. Benchmark parametrization: All trajectories are fragmented with a deletion rate of 10% ($\Theta[1, 0.1]$). There are no fragmentation artifacts through division events ($\kappa_{DA}[0, ODM[\infty, 0])$). For the fragmentation, a bunch-deletion module with a mean bunch length of 1 is used indicating short deleted gaps ($QRM_{BB}[1, 0, 1]$). In total $N_T = 1655$ are contained in the dataset. The benchmarks are displayed in Figure 3.11.

tracted a more detailed subsequent group-tailored analysis (Section 3.1) can be applied, enormously reducing the overall effort to analyze only subparts of

fragmented tracking data. Besides the percentage of representative tracks, the way in which the representative tracks are chosen has also a high impact on the allocation result. Either, the tracks are chosen automatically by e.g. using automatic curation routines or in an interactive manual fashion. The effect of the representative tracks selection mode is depicted in Figure 3.14 using database $TDB_S3[D_{BG} = 20]$.

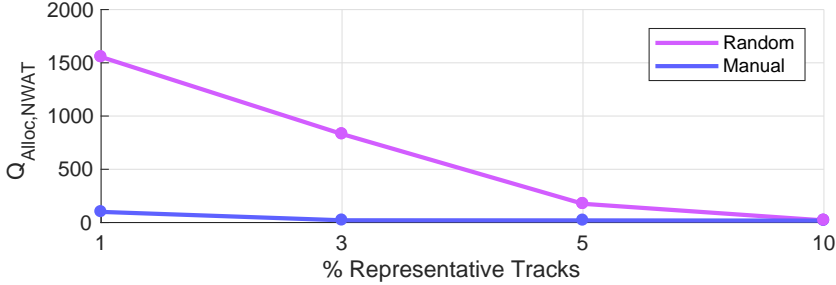


Figure 3.14: The impact of manual vs. automatic and random selection of representative tracks. Benchmark TDB_S3 from Figure 3.11 with overlapping groups is used for validation ($TDB_S3[D_{BG} = 0]$). For different percentages of representative tracks the number of wrong allocated trajectories is counted ($Q_{Alloc,NWAT}$). Here, all trajectories are fragmented with 10% of deletion ($\Theta[1, 0.1]$). There are no fragmentation artifacts through object division events ($\kappa_{DA}[0, ODM[\infty, 0]]$). For the total number of $N_T = 1655$ trajectories the 5 nearest neighbors are calculated ($Q_{Alloc,NN} = 5$).

Here, the number of wrongly allocated tracks ($Q_{Alloc,NWAT}$) is displayed for the selection mode (random or manual) and the corresponding percentage of representative tracks. For the manual selection mode the quality of allocation is very high even for small percentages of representative tracks. Whereas, the automatic selection of representative tracks in a random way leads to a much worse allocation result (Table 3.3). With an increase of the percentage of the representative tracks, the difference between the manual and random selection mode becoming increasingly smaller. However, the manual selection mode leads to the same allocation result using only 1% of representative tracks compared to the random selection mode using ten times of the same percentage (Table 3.3). This result lead to the conclusion that an interactive manual selection of representative tracks in an intelligent manner by e.g. manual picking representative tracks in border regions equally distributed over the existing groups, leads to a high reliability of the allocation result. Furthermore, less representative tracks

Method	$Q_{Alloc,PRT}^1$	$Q_{Alloc,PRT}^3$	$Q_{Alloc,PRT}^5$	$Q_{Alloc,PRT}^{10}$
Random	1555	831	176	20
Manual	100	21	20	18

Table 3.3: The impact of manual vs. random selection of representative tracks. The benchmark TDB_S3 with overlapping groups is used for validation (TDB_S3[$D_{BG} = 0$]). In the table the number of wrong allocated tracks ($Q_{Alloc,NWAT}$) is listed for different combinations of percentage representative tracks ($Q_{Alloc,PRT}$) and the selection mode (random and manual). For the benchmark parametrization all trajectories are fragmented with 10% of deletion ($\Theta[1, 0.1]$). The benchmark contain no division artifacts leading to fragmentation ($\kappa_{DA}[0, ODM[\infty, 0]$). Furthermore, a bunch deletion module with a mean bunch length of 1 ($QRM_{BB}[1, 0, 1]$) is used to fragment the benchmark leading to short gaps within the trajectories. For the total number of $N_T = 1655$ trajectories the 5 nearest neighbors are used to calculate the neighborhood ($Q_{Alloc,NN} = 5$).

are required to achieve the same quality of allocation in manual selection mode compared to the random selection mode (Figure 3.14). The allocation process offers not only the possibility to allocate a track fragment to the predominant class within its neighborhood, but further uses the dispensation of the groups within the neighborhood. Therefore, the percentage of the predominant group ($Q_{Alloc,PMC}$) within the neighborhood serves as a basis to further tune the allocation process to improve the validity of the assignment. Adapting $Q_{Alloc,PMC}$ it is possible to control the assignment of the track fragments. E.g. choosing $Q_{Alloc,PMC}$ to be 80% has the sense that only track fragments where the predominant groups is represented in more than 80% of the neighborhood \mathcal{T}_{NBH} are allocated. If not all fragmented tracks are assigned to the predominant groups within the neighborhood, not allocated track fragments ($Q_{Alloc,NNAT}$) are the result. To systematically evaluate the impact of the percentage of the predominant group ($Q_{Alloc,PMC}$), Figure 3.15 illustrates the effect of $Q_{Alloc,PMC}$ on the number of wrongly allocated ($Q_{Alloc,NWAT}$) and the number of not allocated tracks ($Q_{Alloc,NNAT}$) for the exemplary database TDB_S3 (Section 2.4) using 5% representative tracks.

It can be observed that the higher the required percentage of the predominant group within the neighborhood, the less tracks are allocated to the wrong group and at the same time the number of not allocated tracks in contrast increase (Figure 3.15A). In Figure 3.15B-C, the location of the not allocated tracks is vi-

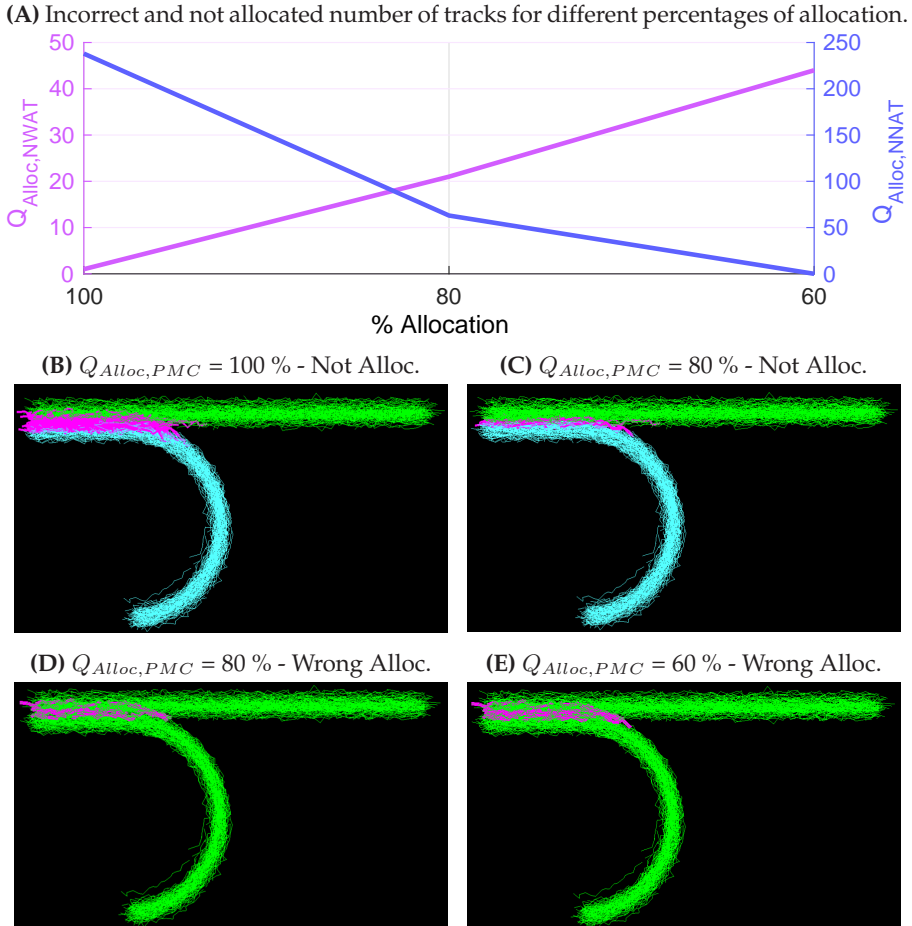


Figure 3.15: Evaluation of the percentage of nearest neighbors in the allocation process. (A) For different percentages of allocation the number of wrong allocated trajectories ($Q_{Alloc,NWAT}$), as well as the number of not allocated trajectories ($Q_{Alloc,NNAT}$) are displayed. (B-C) The visualization of not allocated trajectories (magenta) for different percentages of allocation. The two groups of trajectories are colored in cyan and green. (D-E) The visualization of wrong allocated trajectories (magenta trajectories located in the border region) for different percentages of allocation. For the benchmark parametrization the benchmark database `TDB_S3` with touching groups is used. For the fragmentation all trajectories are used with a 5% of deletion ($\Theta[1, 0.05]$, $\kappa_{DA}[0]$, $ODM[\infty, 0]$). Hereby, a bunch deletion module ($QRM_{BB}[1, 0, 1]$) with a mean bunch length of 1, leading to short gaps is used for the total number of $N_T = 1655$ trajectories. For the neighborhood calculation, the 5 nearest neighbors are used ($Q_{Alloc,PRT} = 5$).

sualized for different percentages of the predominant group ($Q_{Alloc,PMC}$). It can be clearly observed that the not allocated tracks are located only in border regions between different groups. Also, the wrongly allocated tracks are located in these transition regions between different groups (Figure 3.15D-E). To minimize the transition regions containing wrong and not allocated tracks the percentage of representative tracks has to be increased. Therefore, Figure C.6 evaluates the same characteristics as described in Figure 3.15, but with 10% instead of 5% representative tracks. Doubling the percentage of representative tracks lead to halve the number of not allocated tracks by at the same time keeping the number of wrong allocated tracks nearly constant (Table 3.4).

$Q_{Alloc,PercRT}$	100 % Alloc.	80 % Alloc.	60 % Alloc.
5 % RT	1/238	21/63	44/0
10 % RT	21/99	28/35	42/0

Table 3.4: Evaluation of quality of allocation according the choice of the percentage of representative tracks in combination with the allocation threshold. For two different percentages fo representative tracks (5% RT and 10 % RT) the wrong/ not allocated number of tracks are listed. Here, benchmark TDB_S3 with spatially touching groups is used for validation ($TDB_S3[D_{BG} = 20]$). For the fragmentation, all trajectories are used with a 10% of deletion ($\Theta[1, 0.1]$, $\kappa_{DA}[0]$, $ODM[\infty, 0]$). Hereby, a bunch deletion module ($QRM_{BB}[1, 0, 1]$) with a mean bunch length of 1, leading to short gaps is used for the total number of $N_T = 1655$ trajectories. For the neighborhood calculation, the 5 nearest neighbors are used ($Q_{Alloc,PRT} = 5$).

Further, the effect of the choice of the number of nearest neighbors ($Q_{Alloc,NN}$) within the neighborhood \mathcal{T}_{NBH} is depicted in Figure 3.16 and Table C.2 respectively. For different sizes of the neighborhood (5, 10, 15 and 20 neighbors) the total number of wrongly allocated ($Q_{Alloc,NWAT}$) and not allocated tracks ($Q_{Alloc,NNAT}$) are plotted for different percentages of the predominant group within the neighborhood. It can be observed that the higher the number of nearest neighbors, the lower the number of wrong allocated tracks along with at the same time a higher number of not allocated tracks (Figure 3.16). In the case where the neighborhood gets much too big (Figure 3.16D) all tracks are not allocated. In general, the choice of nearest neighbors and the strategy for allocation depends on the specific characteristics of the underlying dataset. To handle this circumstance, the interactive adaptation of the allocation paramete-

ters described in Chapter 4 leads to a database-tailored choice of the best allocation parameters.

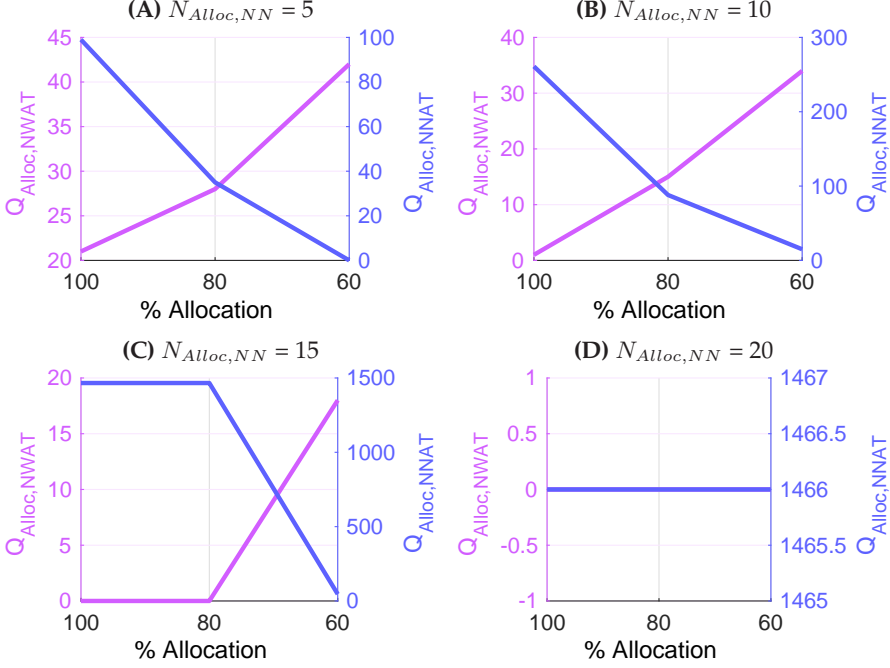


Figure 3.16: The effect of the choice of the number of nearest neighbors ($Q_{Alloc, NN}$) within the neighborhood \mathcal{T}_{NBH} is depicted for 5 nearest neighbors (A), 10 nearest neighbors (B), 15 nearest neighbors (C) and the 20 nearest neighbors (D). Here, for all number of nearest neighbors (A-D), the number of wrong allocated trajectories ($Q_{Alloc, NNAT}$) and the number of not allocated trajectories ($Q_{Alloc, NNAT}$) are displayed for several percentages of allocation. For evaluation benchmark database TDB.S3 with spatially touching groups (TDB.S3[$D_{BG} = 20$]) is used. All trajectories within the benchmark are fragmented using a deletion of 10% ($\Theta[1, 0.1]$). There are no fragmentation artifacts through object division ($\kappa_{DA}[0, ODM[\infty, 0]]$). For fragmentation a bunch deletion module with a mean bunch length of 1 ($QRM_{BB}[1, 0, 1]$) is used, indicating short gaps for the total number of $N_T = 1655$ trajectories.

3.3.3 Quantitative Description of Object Movement Behavior

Complementary to the qualitative visual analysis of the tracking data (Section 4.1), a major benefit of the digital representation of the object trajectories is the opportunity to quantitatively characterize the movement of individual objects or entire groups of objects. A trajectory feature therefore represents a given characteristic of a trajectory, such as the curvature or a characteristic of the spatial movement path. There are different possibilities to calculate the trajectory features. One possibility is to use only the spatial coordinates of a trajectory for the feature calculation like the spatial length. An other possibility is to further incorporate the temporal information of a trajectory such as the time interval of occurrence into the trajectory feature calculation. Considering both the spatial and the temporal information allow calculating features such as velocities, neighbor dependencies and global migration patterns of a trajectory. These trajectory features can be used to find and extract trajectories of moving objects exhibiting a predefined characteristic. This could be objects that move in a given direction, objects that move far away or objects that stay stationary. The advantage is that existing prior knowledge (Section 3.2) can be easily translated into trajectory features. The KD-process (Section 3.1) can then be used to efficiently find objects exhibiting a certain characteristic of interest. Furthermore, the interactive manual handling of the trajectory features to extract relevant groups described in Section 4.1.4 is an additional possibility. In Table 3.5, a set of single features (SF) and time series features (TS) are listed. Here, a single features contains one scalar value per trajectory comprising global measures, whereas time series features (TS) containing one separate scalar value for each time point of a trajectory that can be used to analyze temporally changing properties of a trajectory. Exemplary time series features are the speed, density, directional changes or continuous distance to a reference point. Additionally, a qualitative visual analysis of the extracted features using the methods described in Chapter 4 is also feasible (Figure C.9). Often, trajectory features are highly application-specific and therefore existing features are either adapted to new tasks (Section 1.2.3) or new features on the basis of application-dependent prior knowledge (Section 3.2) from experts are implemented.

In addition to global trajectory features, all single features (SF) provided in Table 3.5 can be used in a sliding window approach with a given window length

Feature Name	Feature Description	Type
Effective displacement	$d_{eff} = d(\mathbf{p}_1, \mathbf{p}_N)$	SF
Total displacement	$d_{tot} = \sum_{i=1}^{N-1} d(\mathbf{p}_i, \mathbf{p}_{i+1})$	SF
Ratio lengths	$r_l = d_{eff}/d_{tot}$	SF
Total time span	$t_{tot} = (N - 1)\Delta t$	SF
Current speed	$v_i = d(\mathbf{p}_i, \mathbf{p}_N)\Delta t$	SF
Mean curvilinear speed	$v_{mc} = \frac{1}{N-1} \sum_{i=1}^{N-1} v_i$	SF
Mean straight-line speed	$v_{sl} = \frac{d_{eff}}{t_{tot}}$	SF
Forward progression linearity	$r_{lin} = \frac{v_{lin}}{v_{sl}}$	SF
Trajectory direction angle	$\theta_{i,*} = \arccos\left(\frac{(\mathbf{p}_i - \mathbf{p}_{i-1}) \bullet \mathbf{u}_*}{\ \mathbf{p}_i - \mathbf{p}_{i-1}\ \ \mathbf{u}_*\ }\right)$	TS
Distance to reference point	$d_{i,ref} = \ \mathbf{p}_i - \mathbf{p}_{ref}\ $	TS
Speed towards reference point	$s_{i,ref} = \frac{(d_{i+1,ref} - d_{i-1,ref})}{2}$	TS
Instantaneous speed	$v_i = \frac{d(\mathbf{p}_i, \mathbf{p}_{i+1})}{\Delta t}$	TS
Instantaneous angle	$\alpha_i = \arctan\left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i}\right)$	TS
Directional change	$\gamma_i = \alpha_i - \alpha_{i-1}$	TS
Mean squared displacement	$MSD(n) = \frac{1}{N-n} \sum_{i=1}^{N-n} d^2(\mathbf{p}_i, \mathbf{p}_{i+n})$	TS
Derivative spherical coordinates	$r'_i = \frac{r_{i+1} - r_{i-1}}{2}$	TS
	$\phi'_i = \frac{\phi_{i+1} - \phi_{i-1}}{2}$	TS
	$\theta'_i = \frac{\theta_{i+1} - \theta_{i-1}}{2}$	TS

Table 3.5: Quantitative movement description of trajectories using single features (SF) or time series (TS). The table is extended from [4].

n_W to extract a time series for each trajectory containing the values of the single features calculated for the sliding window at each time point (Figure C.8). Here, the user can adapt the window length and can therefore control the impact of temporal and spatial characteristics. To further remove local fluctuations and to obtain a more robust extraction of trajectory features that visually match the observed global behavior, trajectories can optionally be temporally smoothed using a low-pass filter. In case of complete trajectories the calculation of trajectory features is well suited to depict given movement characteristics and serves

as a perfect basis to use the KD-process (Section 3.1) to efficiently gain knowledge about the underlying moving objects (Section 3.3.5).

However, more complex problem classes (Section 2.1) contain highly fragmented and noisy tracking data. The problem class has an enormous impact on the quality and the informative value of the extracted trajectory features. To systematically evaluate the effect of an increasing problem class on the validity and significance of the extracted features, the noise and the level of fragmentation are used to simulate the worsening of problem classes. Therefore, the benchmark database `TDB.S3` is used containing one group with a straight movement characteristic and another group with an internalizing movement characteristic (Section 2.4). For the evaluation two features are calculated for the different simulated problem classes: One features that describe the ratio of effective and total length r_l (Table 3.5) that is well suited to distinguish the two groups in database `TDB.S3` and one feature describing the mean curvilinear speed v_{mc} that is not appropriate to describe the differences between the two existing groups (Table 3.5). To address the complexity of problem classes, the level of fragmentation is simulated using the bunch-based deletion module with the parametrization $QRM_{BB}[1, 0, 1]$ (Section 2.3.2) and a varying percentage of deletion. For the well-suited feature describing the ratio of lengths, Figure 3.17 (corresponding Table C.3) shows that without fragmentation (percentage deletion = 0%) the two groups are clearly distinguishable. An increasing fragmentation level represented by the growing percentage of deletion is leading to a rapid vanishing of the distinctiveness between the two groups. Even small percentages of deletion starting at 1% dramatically reduce the separability between the two groups (Figure 3.17). Depending on the individual application, the fragmentation of the tracking data leading to shorter fragments of trajectories enormously affect the ability to describe to underlying movement characteristic correctly. It is essential to choose the correct features based on prior knowledge (Section 3.2) and use effective methods to handle the fragmented trajectory data (Section 3.3.2). The increasing noise (simulated using the adding of Gaussian noise) is another aspect of problem classes with higher complexity. The reason leading to this noise artifact is depicted in Section 2.1 in detail. The effect on the distinguishability of groups is exemplary depicted in Figure 3.18 (corresponding Tables C.4 and C.5) using different noise levels. For the well suited feature r_l an increasing noise level leads to a strong vanish-

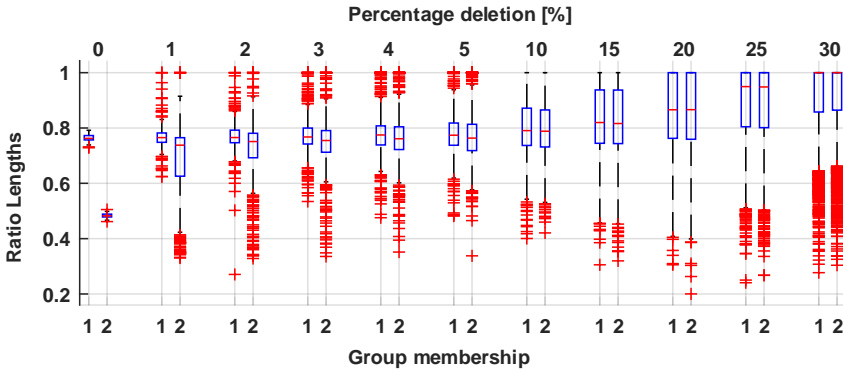


Figure 3.17: Evaluation of group distinguishability in relation to the percentage of deletion for each trajectory using benchmark *TDB.S3*. For different percentage of deletion reaching from 0% up to 30% the distribution of the features ‘ratio lengths’ is visualized for two groups (group membership 1 and 2). For the deletion, a bunch-deletion module with parametrization $QRM_{BB}[1, 0, 1]$ is used, leading to short deleted gaps.

ing of the separability of the two groups (Mean and standard deviation time series in Figure 3.18A). In case of the not appropriate feature v_{mc} the mean and standard deviation time series in Figure 3.18B point out that only the scattering of the features increase with an increasing noise level but the distinguishability remains poor. In conclusion, the complexity of the problem class has an enormous effect on the quantitative feature description. If fragmentation of the trajectories occurs, efficient methods to handle such fragmented tracks are required (Section 3.3.2) to obtain the validity of feature description. In case of high noise artifacts, methods such as trajectory smoothing are required to get rid of local fluctuations and maintain the global movement path to obtain the separability of existing groups.

3.3.4 Focusing on Spatio-Temporal Regions

Not all objects and respectively their trajectories within one database are of interest for a given analysis task. Therefore, only predefined trajectories within a region of interest (*ROI*) are considered for subsequent analysis. To focus on

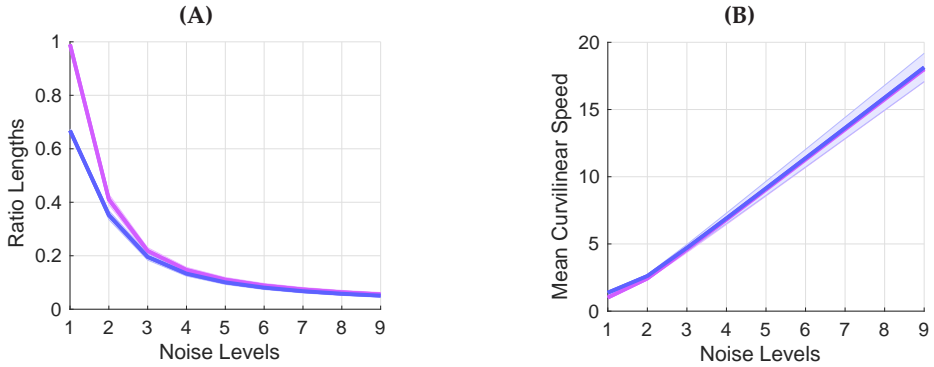


Figure 3.18: Impact of noise artifacts to group distinguishability for each trajectory using benchmark `TDB_S3`. For different noise levels the mean time series for the two groups within benchmark `TDB_S3` (colored in magenta and blue) for the features ‘ratio lengths’ (A) and the features ‘mean curvilinear speed’ (B) are illustrated.

the region of interest, the database can be globally truncated to a fixed spatial region and by specifying the time interval of interest. The knowledge of the spatio-temporal occurrence of the groups of interest in advance reduces the data size dramatically [346]. Focusing on a region of interest is often used because specific phenomena or behaviors of objects do not occur in the whole database but rather in specific regions. Therefore, the focusing enables to analyze object movement pattern in a given region. For the analysis of the tracking data often features (Section 3.3.3) are used to describe characteristics of the trajectories. However, these features are only meaningful if they are calculated within a region of interest where the specific behavior occurs. If the features are calculated for the whole dataset, given characteristics are averaged out due to the global context for the calculation. It is also possible to use a feature-based region of interest selection approach (see Chapter 4 for more details) to improve the objectivity and reproducibility. Several influencing factors such as the size of the *ROI*, the choice of given features as well as the noise level of the underlying problem class have a great impact on the result of region of interest selection with subsequent analysis. In the following, these influencing factors are evaluated systematically using derivatives of database `TDB_S6` (Section 2.4). The database `TDB_S6_D1` contains one undirected movement period of length 10 at time point 10 (Figure C.10A) whereas the database `TDB_S6_D2` contains two undirected movement periods of length 10 at time point 50 and

time point 100 (Figure C.10B). The effect of the choice of the correct features is depicted in Figure 3.19. The corresponding statistical significance of the two

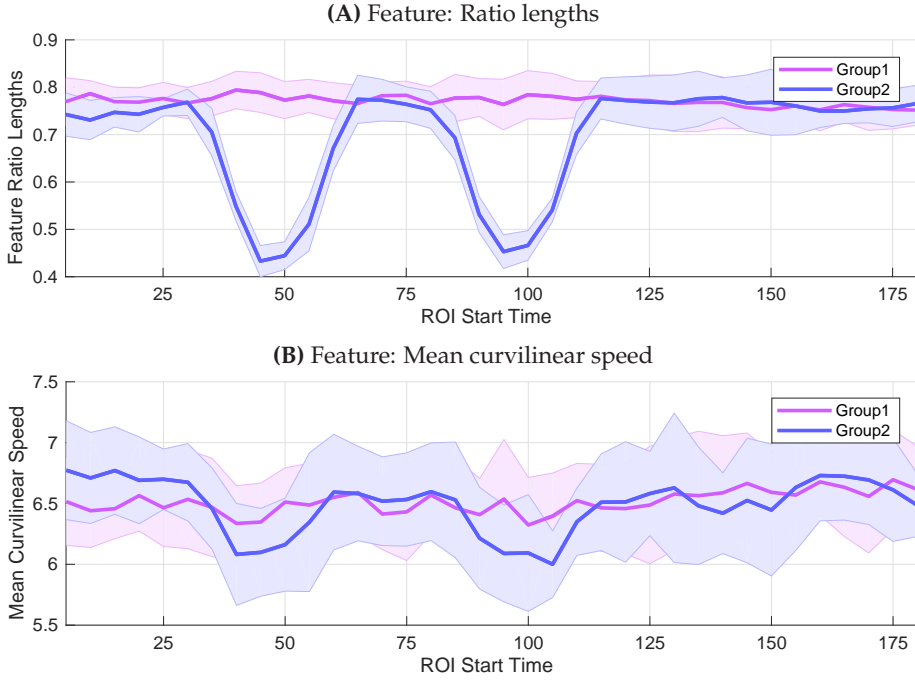


Figure 3.19: Choice of region of interest with predefined appropriate size using benchmark TDB_S6 . The window size of the ROI is set to be 20 time points. There is no noise added to the benchmark ($Q_{A,Noise} = 0$). (A) Mean time series of the two groups within the benchmark (colored in magenta and blue) for the feature 'ratio lengths'. (B) Mean time series of the two groups within the benchmark (colored in magenta and blue) for the features 'mean curvilinear speed'.

groups is listed in Table C.6. The feature describing the ratio of the effective and absolute displacement of the trajectories (Figure 3.19A) is generally well-suited to address such a characteristic and find significant differences within the existing groups (Table C.6). However, the mean curvilinear speed, representing a poorly suitable feature to describe the existing undirected movement patterns, is not able to well separate the two existing groups (Figure 3.19B). In the case of correctly chosen features, the choice of the size of the ROI has an enormous effect. Therefore, Figure C.11 shows the same characteristics as in Figure 3.19 with the only difference of choosing the size of the ROI too big. This leads

to a vanishing of the time-specific differences between the two groups (Table C.7). To systematically evaluate the effect of the size of the *ROI*, the database *TDB_S6_D1* with a single undirected movement period at time point 10 is used (Figure 3.20). Here, the size of the region of interest is successively incremented starting with a size of 5 time points and ending with 180 time points. For each size of the *ROI*, the well suited feature describing the ratio of effective and absolute length of the tracks is calculated (Figure 3.20). It can be observed, that the greater the size of the *ROI*, the smaller the differences of the two groups leading to a vanishing of the distinguishability (Table C.8). In general, the optimal size of the *ROI* is application-dependent and has to be adapted manually. When choosing the *ROI* too big or too small (Figure 3.20) the separability of different groups is vanishing.

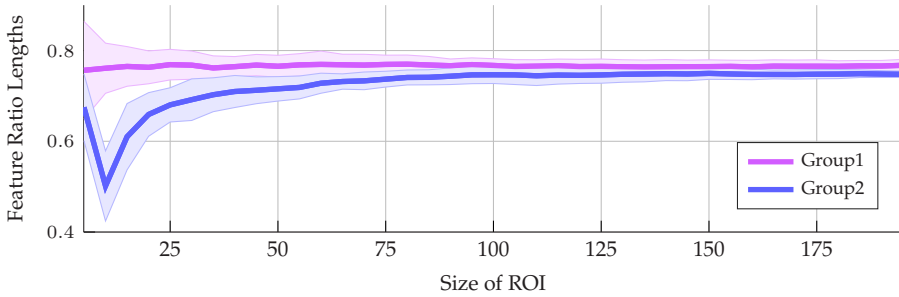


Figure 3.20: Impact of the correct size of the region of interest using benchmark *TDB_S6*. The size of the region of interest varies starting from 5 time points up to 190 time points. For the feature ‘ratio length’ the mean time series for the two groups (magenta and blue) are displayed for the varying size of the ROI. There is no noise added to the benchmark ($Q_{A,Noise} = 0$).

The complexity of the problem class of the tracking database (Section 2.1) has also a great impact for the region of interest extraction result. Here, the increasing noise added to the databases is exemplarily used to show the effect of an increasing difficulty of the problem classes to the *ROI* analysis. Different noise levels (Figure C.7) are applied to the dataset *TDB_S6_D2*. A subsequent extraction of the feature describing the ratio of effective and absolute lengths of the tracks is applied to the sliding *ROI* with different start points. The characteristic curves for the two groups is displayed in Figure C.7. It can be observed that an increase of the noise leads to a decreasing distinguishability of the two

groups pointing out that the complexity of the problem class has an enormous effect on the analysis result in the way that the higher the fragmentation and the noise level, the worse the distinguishability of extracted groups.

3.3.5 Feature-Based Extraction of Groups of Trajectories

Within the knowledge discovery process (Section 3.1) the feature-based group extraction approach is an essential component to access underlying effects such as movement characteristics that can not be found manually and to find groups of possible interest within large-scale trajectory databases. Representing a trajectory by a feature (Section 3.3.3) is an enormous reduction of the dimensionality facilitating the application of feature-based group extraction methods such as clustering and filtering (Figure 3.21). The exemplary application of a fuzzy c-means clustering method comprising four clusters is depicted in Figure 3.21A, whereas a filtering approach in Figure 3.21B is used to easily segregate two opposing sides within the benchmark database `TDB_S3`. Furthermore, the existing prior knowledge (Section 3.2) can be incorporated in these computational methods by exemplary setting the number of clusters or by specifying relevant filter intervals to efficiently extract groups of potential interest. Once the groups are extracted, interactive modifications can be applied elaborating the results by slight refinements of the allocated information (Section 4.2). The computational feature-based group extraction methods, therefore, allow to easily fathom huge trajectory databases in case of existing prior knowledge enabling the effective investigation of emerging effects (Section 3.2.2).

For the clustering approach in trajectory data, time series clustering using the time series representing the spatial location in X-, Y- and Z-direction is also possible. However, the time series clustering approach is inappropriate for trajectories exhibiting a non error-free attitude leading to the non-applicability to trajectory problem classes (Section 2.1) comprising fragmented tracking data. Even, if the trajectories are spanning the whole time interval without fragmentation artifacts equal to problem classes of lower complexity (Section 2.1), time series clustering is not suitable for all group extraction approaches. Exemplary, Figure 3.22A illustrates two groups of trajectories out of benchmark `TDB_S4` (Section 2.4.2): One group (green) that moves straight and the other group (magenta) displaying a turn-around movement. The result of time series clustering

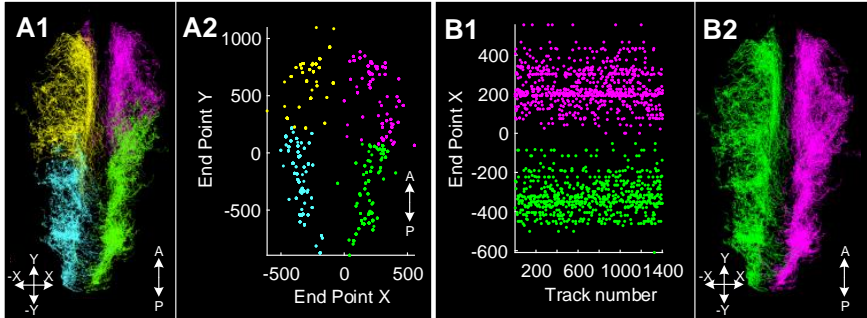


Figure 3.21: Feature-based group extraction methods. (A) Clustering the X- and Y-position of the end points comprising four clusters (A2) with the result applied to the 3D trajectory data (A1). (B) Filtering approach on the basis of the end point in X-direction easily segregating two opposite sides (B1) and the result applied to the 3D trajectory data (B2). Here, benchmark `TDB_S2` is used for validation purpose.

using the X-position, Y-position and Z-position time series is depicted in Figure 3.22B, separating two symmetrical groups instead of the feature distinguishable groups (Figure 3.22A). In conclusion, time series clustering of trajectory data is not possible in presence of fragmentation and higher levels of complexity (Section 2.1). Furthermore, not all characteristics can be separated using time series clustering. In these cases, extracted features for each trajectory (Section 3.3.3) can be used to access the underlying characteristics even in presence of fragmentation artifacts.

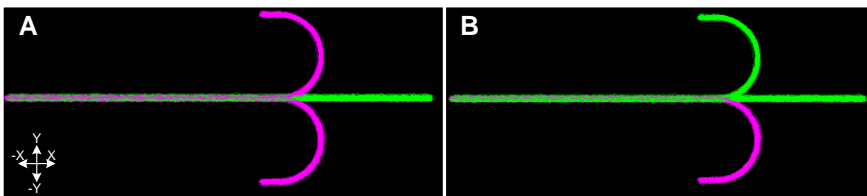


Figure 3.22: Time series clustering on trajectory data using benchmark databases `TDB_S4`. (A) Ground truth groups. Here, the green group is moving straight, whereas the magenta group performs a turn-around movement. (B) Result of time series clustering using the spatial coordinate time series in X-, Y- and Z-direction.

In feature-based group extraction approaches the incorporation of prior knowledge as described in Section 3.2, as well as the type of analysis approach, is highly relevant for the resulting quality and the required time effort. There are one-step approaches (OS) accessing the extracted information on a single analysis step, globally applied to the whole data, as well as multi-step approaches (MS) using a hierarchical analysis strategy allowing the use of several constructing methodological approaches. The resulting time effort and quality are depicted in Figure 3.23. Here, four different benchmark databases belonging to highly complex problem classes introduced in Section 2.1, namely benchmark `TDB_S8`, `TDB_S9`, `TDB_S10` and benchmark `TDB_S11` are used for validation purpose. The complexity of the benchmark databases increases with the increasing number of the benchmark pointing out that benchmark `TDB_S11` is the most complex one and benchmark `TDB_S9` the one with the lowest complexity in comparison to all others. The quality is assessed using the accuracy describing the percentage of correctly assigned trajectories to the ground truth groups. It can be observed that fully automatic one-step approaches require a negligible effort in time compared to the manual incorporation of prior knowledge. However, the achieved accuracy of the automated OS approaches is much below the approaches incorporating prior knowledge (Figure 3.23). Furthermore, it can be observed that the higher the problem class, the lower the accuracy. Even in the presence of benchmarks with high complexity (Section 2.4.2) used for this validation, multi-step approaches incorporating prior knowledge lead to perfect group extractions with a 100% accuracy, but with the drawback of an increase in time effort (see Table C.9 for detailed numbers). Moreover, the total time effort in case of existing prior knowledge is at an acceptable level not exceeding 100 seconds. Even in presence of no prior knowledge the time effort is only maximal five times higher, not exceeding 550 seconds. The analysis was performed by an experienced user of the framework suggesting that a non-experienced user may need the twice or triple amount of time being in the range of approximately 3 to 5 minutes in total in case of existing prior knowledge, which is also very fast. In the cases without prior knowledge the multi-step approaches also lead to a 100% accuracy accompanying a much higher time effort (Figure 3.23). An increase within the complexity of the benchmark lead to a disproportionate rise of the required time effort. Concluding, even highly complex trajectory datasets can be investigated and relevant groups can

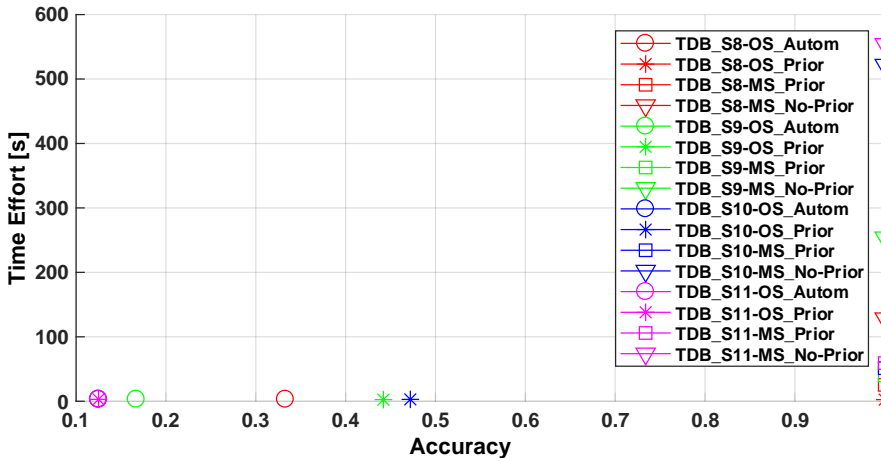


Figure 3.23: The impact of one-step and multi-step analysis approaches combined with the integration of prior knowledge. For four different benchmark databases TDB.S8, TDB.S9, TDB.S10 and TDB.S11 a fully automatic approach (circles), a one-step approach using prior knowledge (stars) and a multi-step approach using prior knowledge (rectangles) and a multi-step approach without the existence of prior knowledge (triangles) are validated using the required time effort and the accuracy of group assignments. In Table C.9 the corresponding numbers of time effort and accuracy are listed.

be extracted with a 100% accuracy. The time effort varies a lot depending on the extent of existing prior knowledge and the complexity of the underlying database (Figure 3.23).

3.3.6 Transfer of Analysis Pipelines to New Trajectory Databases

The KD-process (Section 3.1) is used to extract relevant groups and allows modifications to optimize the results till the expert is satisfied with the analysis result. Then, the whole process is applied to new databases (Figure 3.24). For the analysis of a database all single steps within the KD-process are logged and saved.

Due to inhomogeneity and differences between the trajectory databases, the user can, if necessary, modify the parameters within each step of the applied KD-process (Section 4.1 - 4.3). As a result, the analysis pipelines on the basis

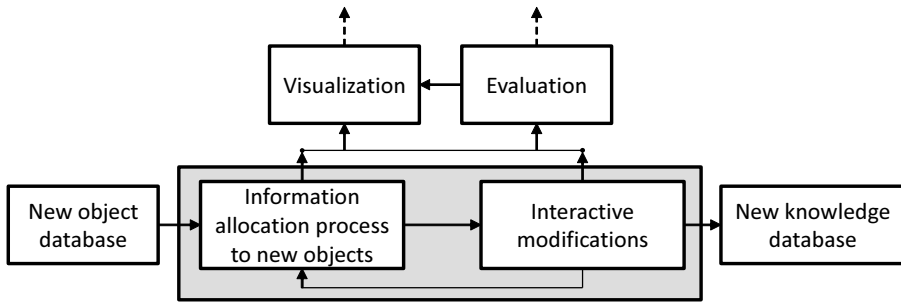


Figure 3.24: General scheme for the application of a KD-process to new databases. Here, the process is displayed to create a new knowledge database out of a new generated object database [346].

of the KD-process can be applied to any other databases efficiently by transferring the logged analysis steps. Databases have often slight variations among each other, resulting from different acquisition settings or other ambient conditions. Therefore, interactive modifications offer the possibility to modify the results of each analysis step interactively, to match the properties of the new dataset. In addition, the direct visual feedback of the knowledge discovery process helps to minimize the inter- and intra-expert variation that usually causes a bias between different users (Section 4.1). It is reasonable to quantify the extracted relevant groups within one single database first, before applying the whole KD-process to other databases. The reason is to analyze the results in a first step in one database in more detail. Through the detailed consideration of the results, the experts get further insights in the data and can investigate assumptions in more detail. Therefore, a slight modification of the KD-process can be necessary (Section 3.1). Finally, when the expert is satisfied with the analysis result of the complete KD-process, the whole process can be applied to other databases to extract the same relevant groups in an efficient way improving the reproducibility and reliability of the performed comparisons. For each trajectory database, a knowledge database is created using the KD-process (Figure 3.24). These knowledge databases can be compared and evaluated finally (Section 3.3.7). As a result, experts can make statistical statements about quantitative differences within and across databases and the respective groups of interest. To evaluate the transfer of complete analysis pipelines to new trajectory databases, two exemplary scenarios are depicted to address different

aspects and difficulties. Therefore, derivatives of databases `TDB_S3` with different internalization time points (Figure 3.25A1-A3) and `TDB_S6` containing short intervals of non-directed movement (Figure 3.25B1-B3) are used. In the

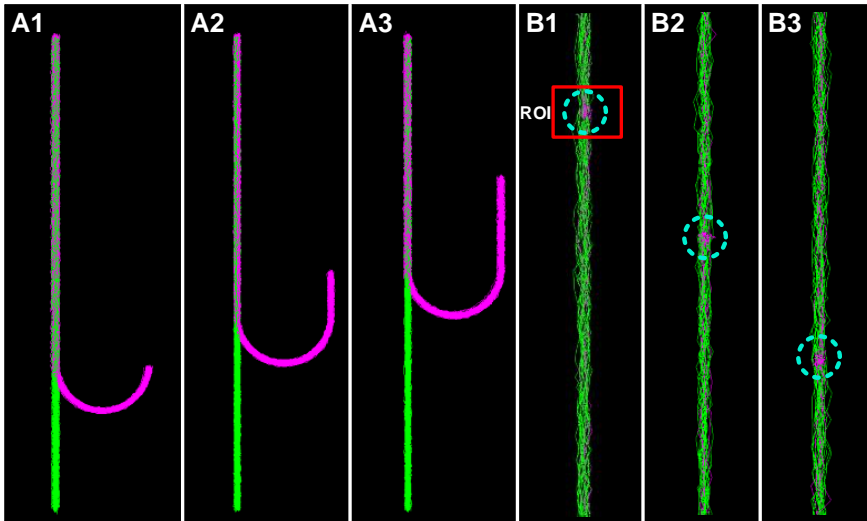


Figure 3.25: Transfer of analysis pipelines to new trajectory databases. (A) An analysis pipeline suited to separate the straight moving trajectories (green) and the trajectories performing an internalization movement (magenta) is designed on one trajectory database (A1) and is applied to two further databases (A2, A3) to extract the same relevant groups (benchmark `TDB_S5`). (B) Here, an analysis pipeline is used to extract one straight moving group (green) and another group having a short period of undirected movement before moving straight again (magenta) in benchmark database `TDB_S6`. The pipeline is transferred to two further databases (B2, B3).

first evaluation, the two different groups, colored in green and magenta (Figure 3.25A), are originally extracted in one database using only one analysis step in which the feature describing the ratio between the effective and total displacement (Section 3.3.3) is clustered. This analysis step is applied to database `TDB_S3, D1` and `TDB_S3, D2`.

3.3.7 Quantitative Comparison of Multiple Large-Scale Trajectory Databases

Once relevant groups are extracted using the KD-process (Section 3.1), these groups can be characterized further in a quantitative way (Figure 3.26). The quantitative comparison of groups can be done either using a time series representation (Figure 3.26B) where each time series indicates the progress of a single characteristic within a trajectory or by using the mean and standard deviation time series (Figure 3.26C) containing the mean time series averaged over all groups.

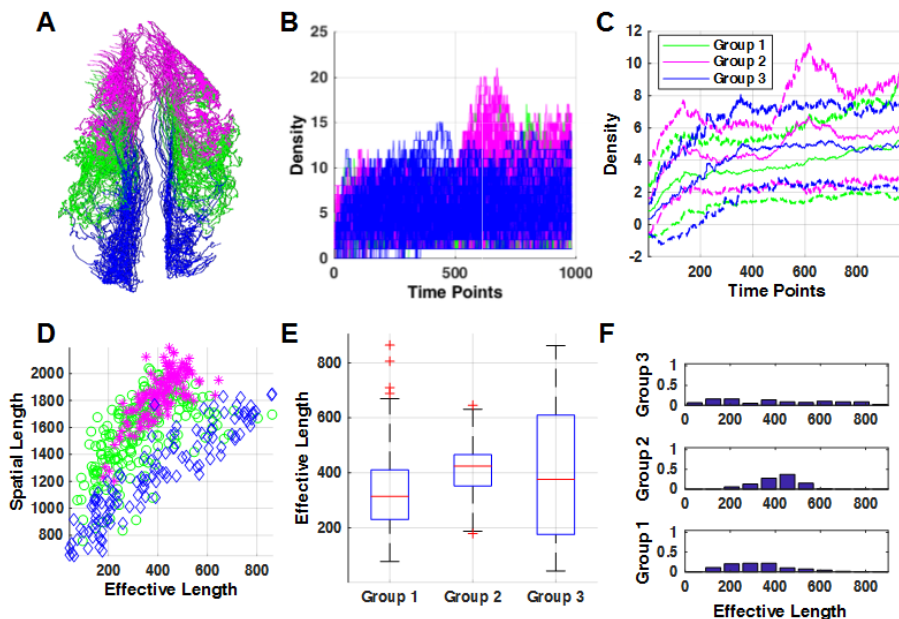


Figure 3.26: Efficient quantitative comparison of multiple groups within trajectory databases using benchmark databases *TDB_S2*. (A) Three manually selected groups (blue, green, magenta) in the 3D trajectory visualization. (B) Density time series for all three groups over all time points. (C) Mean- and standard deviation time series for the extracted groups. (D) Scatter plot comprising the spatial length and the effective length of the trajectories. (E) Box plots describing the distribution of the three extracted groups using the feature describing the effective length. (F) Histograms for all three groups using the feature depicting the effective length. The color code in all panels is consistent.

In case of fragmented trajectory data the mean is calculated by using only the trajectory fragments that are present in the actual time point. Scatter plots can be further used to access global characteristics of the trajectories. Here, each point within the scatter plot represents the calculated characteristic for one trajectory (Figure 3.26D). To additionally address the distribution of the data, box plots (Figure 3.26E) as well as histograms (Figure 3.26F) are a valuable visual representation to get a fast overview of the uniformities within a dataset. Additionally, effects that are observed visually (Section 4.1) using the interactive handling of the data can be depicted in a quantitative way. Therefore, characteristics for the trajectories in each extracted group are calculated representing aspects within these groups (Section 3.3.3). This serves as a basis to make statistical statements, e.g. calculating the statistical variation of a characteristic within one group or between different groups of trajectories allowing to make quantitative statements about different groups within one database or about the behavior of the same groups across multiple datasets. Such a quantitative concept leads to the deeper investigation of relevant groups and the gaining of additional knowledge based on quantitative feature description of occurring events (Section 3.1). Also, further knowledge can be gained by especially using a combination of two or more quantitative data representations leading to the discovery of phenomena that are not ostensible. The newly gained knowledge can then be used for further validation of additional hypotheses derived by experts, to serve as a starting point for an in-depth analysis of the observed phenomena within the moving object characteristics or to generate a new tailored database to investigate emerging effects on a more detailed level. Additionally, assumptions made by experts are supported by underpinned quantitative results (Section 3.1.1).

3.3.8 Quantitative Lineage Analysis of Dividing Objects

Trajectory databases with a high level of complexity often contain dividing object characteristics that result from splitting events within the spatio-temporal dynamics of moving objects (Section 2.1). To systematically analyze these division characteristics in a detailed and profound manner and gain additional knowledge (Section 3.2.2) about the underlying effects, different tailored characteristics are calculated on the basis of the dividing object trajectories. Each of

the characteristics described in Figure 3.27 access different aspects of the division pattern in a quantitative way.

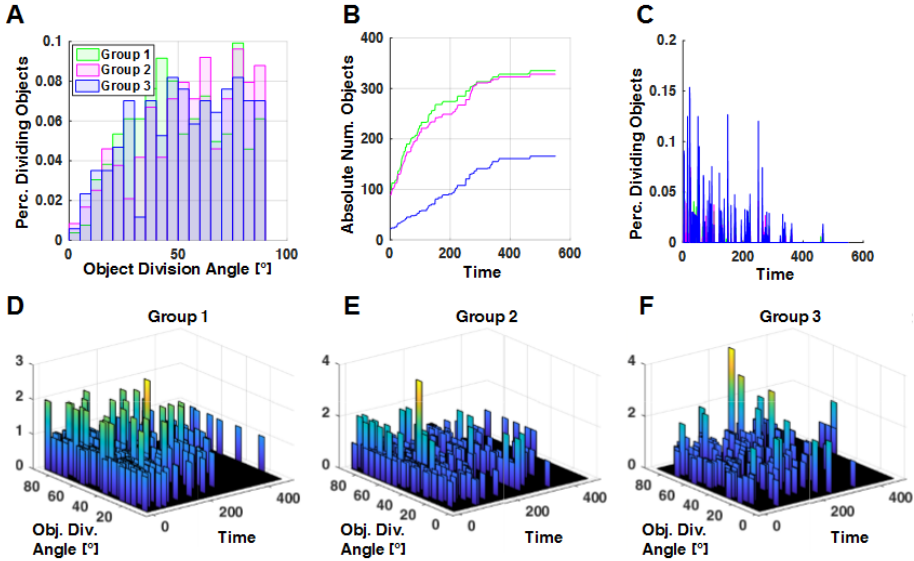


Figure 3.27: Quantitative line analysis of dividing objects. The groups and the corresponding coloring is equivalent to Figure 3.26. (A) Display the object division angle for all extracted groups. (B) Absolute number of objects for the three dedicated groups over time. (C) Percentage of dividing objects over time. (D)-(F) Object division angle over time for the three groups.

Through object division events the total number of objects increasingly grows with time. Therefore, Figure 3.27B depicts the correlation of the total number of objects contained within the dataset and the corresponding time interval. An additional characteristic to describe the division event is the division angles of the dividing migrating objects provide indications of application dependent spreading characteristics of objects (Figure 3.27A, E-F). As an extension of the quantitative feature description of the total object movement paths (Section 3.3.3) the division features sustain the qualitative and quantitative understanding of occurring phenomena interconnected with the moving object dynamics.

3.3.9 Quantitative Neighborhood Assumptions

Trajectories can not only be described on a single trajectory level as extensively described in Section 3.3.3. Moreover, a trajectory can also be described in the context of the surrounding environment. Therefore, a neighborhood (\mathcal{T}_{NBH}) containing the surrounding trajectories is calculated for each single trajectory (Section 3.3.2). Afterwards, a single feature (Table 3.5) is calculated for the current trajectory and the trajectories within its neighborhood. The feature values x of the trajectories in the neighborhood are then used to estimate a normal distribution:

$$f(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (3.11)$$

Moreover, in more complex cases also further distributions can be used. The parameters μ and σ of the normal distribution comprises information about the compactness of the neighborhood regarding the underlying feature. Furthermore, assumption of a selected trajectory in the context of the surrounding neighborhood can be used to depict the homogeneity aspect. Therefore, the feature value of a selected trajectory is set in context to the feature values of its neighbors by calculating the distance ($d_{Mean,NBH}$) to the mean of the approximated normal distribution (Figure 3.28).

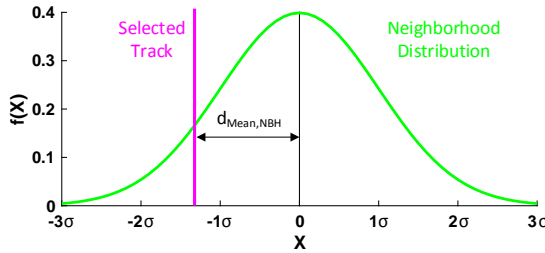


Figure 3.28: Schematic scheme to make homogeneity assumption for a selected trajectory within its neighborhood using the normal distribution of a feature represented within the nearest neighbors.

A high value of $d_{Mean,NBH}$ emphasize that the surrounding neighbors behave inhomogeneous to the actual selected trajectory regarding the chosen feature,

whereas a small value of $d_{Mean,NBH}$ indicates that the actual selected trajectory behaves the same way as the neighbors for the given feature description. Exemplarily, the homogeneity assumptions for benchmark database TDB_S3 on the basis of the feature r_l (Table 3.5) is depicted in Figure 3.29.

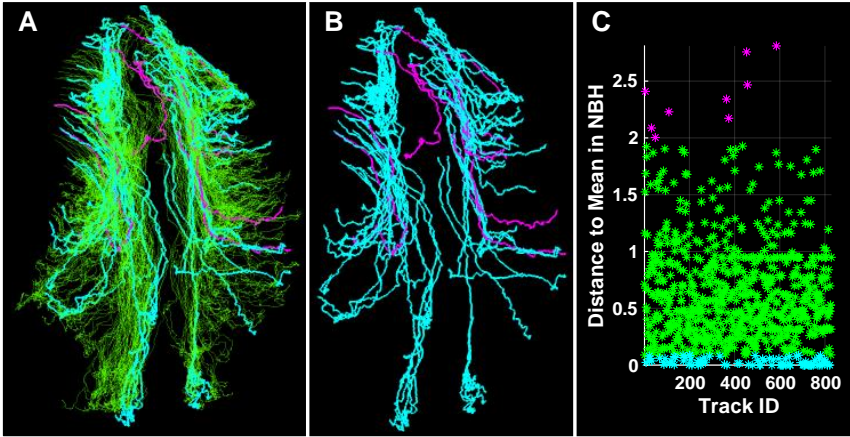


Figure 3.29: Neighborhood-based trajectory feature description using benchmark database TDB_S2. Here, the distance (feature is effective length) to the mean of the neighborhood trajectories is calculated (C). In magenta trajectories with a very high distance (inhomogeneity) and in cyan with a very low distance (homogeneity) are shown in the context of the whole trajectory database (A) and only the selected groups (B). For the neighborhood calculation the 10 nearest neighbors are incorporated.

Two groups of trajectories are selected using the interactive framework described in Section 4.2. The magenta group comprises trajectories with a high value of $d_{Mean,NBH}$ (Figure 3.29C) in reference to its neighborhood indicating trajectories that behave inhomogeneous in the context of their surrounding neighbors, whereas the cyan group has a very small value of $d_{Mean,NBH}$ depicting a high homogeneity within its neighborhood (Figure 3.29A,B).

3.4 Discussion

In this chapter a variety of knowledge discovery methods for large-scale 2D and 3D trajectory databases to efficiently incorporate existing prior knowledge are depicted in detail. Using these analysis methods tailored for tracking applications, underlying phenomena and occurring effects can be investigated on a detailed single trajectory level allowing to efficiently gain knowledge about the trajectory databases. The conceptual approaches developed in this chapter are also applicable to fragmented tracking data allowing to incorporate even highly complex problem classes in the analysis. To efficiently incorporate existing prior knowledge in the analysis of large-scale trajectory databases a new knowledge discovery process is introduced allowing to handle different origins of expert-tailored prior knowledge. Therefore, a wide variety of trajectory characteristics are introduced in this chapter depicting different aspects of prior defined trajectory characteristics. However, it is not feasible to automatically incorporate not formalized prior knowledge yet. To cope with such new emerging prior knowledge for application-specific trajectory datasets, a systematic procedure to efficiently incorporate new characteristics using the developed interface is described in this chapter. To efficiently handle the circumstance of error-prone fragmented tracking data an approach using representative trajectories resulting from initial automatic fusion routines or by region-specific manual curation effort was developed. This approach allows to allocate trajectory fragments to the predominant group frequency of the identified spatio-temporally nearest neighbors with a minimal effort of time. For spatially well separated groups the allocation approach provides good results. However, in highly dense regions with multiple groups intersecting especially in border regions, the allocation process is not suited well to perfectly allocate the remaining trajectory fragments. An interactive selection of representative trajectories located in the transition regions however can improve the allocation result, still allowing to make reasonable global assumptions about the fragmented trajectory dataset. To further analyze the trajectories in specific regions in detail, a feasible approach is introduced to globally truncate the trajectories used for analysis tasks to a fixed spatial region by specifying the time interval of interest allowing to evade to average out specific phenomena and behaviors of interest. Moreover, a feature-based group extraction approach suited for trajectory data

allows finding groups of possible interest by efficiently incorporating prevalent prior knowledge within clustering and classification task. However, not in all application-specific trajectory databases the groups of potential interest can be automatically extracted without the presence of convenient prior knowledge. To cope with the occurrence of sparse prior knowledge, Chapter 4 provides a variety of interactive visual representations to efficiently gain new knowledge and find underlying phenomena in large-scale trajectory databases. Once, relevant groups of potential interest are extracted within one 3D+t database using the knowledge discovery process developed in this chapter, the results can be efficiently transferred to new databases allowing the additionally handling of emerging inhomogeneities. However, an automatic transfer to new database comprising strong inhomogeneities is not possible, instead application tailored manual interactive adaptations are used to efficiently cope with this problem. Moreover, this chapter provides a wide range of methods to quantitatively describe extracted groups within trajectory databases, allowing to make profound assumptions of inter- and intra-group characteristics leading to the discovery of phenomena that are not ostensible. To additionally access trajectory databases with an enormous complexity containing dividing object, this chapter offers methods for a quantitative lineage analysis comprising spatial and temporal aspects. These methods, however require the preceding curation of fragmented trajectory data to obtain valid assumptions about the underlying dividing object characteristics. Moreover, a systematic approach to classify a trajectory in the context of the surrounding neighborhood is depicted in this chapter allowing to make statements about the homogeneity of the embedded environment of each trajectory. In future work, trajectory group pattern such as flock and convoy behavior can be integrated in the existing interface provided by the actual knowledge discovery framework to complement the trajectory features catalogue. Furthermore, approaches to automatically extract relevant groups of potential interest without requiring existing prior knowledge and human effort are the challenging tasks for the future.

4 Enhancing Interactivity Through Prior Knowledge Integration in Trajectory Knowledge Discovery

The systematical integration of prior knowledge in the knowledge discovery process of large-scale trajectory databases was depicted in the previous chapter. Here, in Chapter 4, visual interactive approaches are presented for the efficient guidance of the knowledge discovery process and the corresponding prior knowledge allocation reaching from rough group assignments up to single object-tailored characteristics. The methods presented in Chapter 3 mainly focus on the automated knowledge discovery process in 3D+t trajectory databases, whereas Chapter 4 introduces interactive possibilities to incorporate existing prior knowledge in the trajectory knowledge discovery process (Figure 3.3). Several visual representations of trajectory data are presented in Section 4.1 for a detailed investigation from multiple perspectives allowing to focus on a wide variety of attributes. Therefore, an efficient 3D representation of trajectories is introduced in Section 4.1.1. Furthermore, a maximum intensity projection overlay (Section 4.1.2), an efficient 3D+t visualization of migrating objects (Section 4.1.3) as well as a feature-based visual trajectory representation (Section 4.1.4) assure full power in interactively accessing trajectory databases. The visual handling of dividing object characteristics is described in Section 4.1.5. Furthermore, the incorporation of additional visual representation in the modular concept is depicted in Section 4.1.6. To interactively dissect the existing trajectories and directly allocate the predefined prior knowledge to the intended location within the spatio-temporal paths of the objects, Section 4.2.1 yields a variety of approaches including online propagation of performed selections (Section 4.2.2). For an efficient visually guided hierarchical analysis,

Section 4.3 depicts a newly developed visualization possibility allowing the expert to easily derive hypotheses and analyze phenomena of highly complex trajectory databases. In case of fragmented tracking data, Section 4.4 additionally provides the possibility to interactively curate tracking errors in an efficient way using a visually guided approach. The combination of the proposed visual representations tailored for large-scale 3D+t trajectory data combined with the possibilities of multi-linked selection strategies is new and pave the way to new approaches in the field of trajectory analysis (Chapter 7). The single visual representation themselves are not new from scratch. However, the ability to online propagate selection results between all mentioned visualizations is new, allowing to incorporate prior knowledge from several sources in the trajectory knowledge discovery process. Furthermore, the visualizations are implemented in the way (Chapter 5) that analysis of large-scale trajectory databases incorporating hundreds of thousands of trajectories gets applicable. The evaluation of the methods presented in this chapter is only useful in combination with the knowledge-discovery possibilities of Chapter 3 and the implementation of the whole framework depicted in Chapter 5. Therefore, Chapter 6 presents a user study to evaluate the overall possibilities and interactive approaches in combination.

4.1 Visual Analysis of Object Migration Patterns

The identification of spatio-temporal patterns and the corresponding analysis of these patterns in large-scale trajectory databases is a nearly impossible challenge without a suitable visual framework that allows focusing on specific regions of potential interest. Therefore, in Sections 4.1.1–4.1.5, a set of highly interactive visual trajectory representations are introduced enabling the efficient interaction with enormously large 3D+t datasets [348].

4.1.1 3D Rendering of Large-Scale Object Trajectory Data

The visualization of the tracks in 3D allows getting insights into the global movement of the objects. This leads to the advantage that users can judge single objects not only from their spatial information at a given time point, but

further using the complete movement path over the whole time span including the possibility of rotating, translating and zooming to a desired region of interest. The 3D visualization of the benchmark `TDB_S2` (Section 2.4) is displayed in Figure 4.1A1, A2 representing two different projections of the 3D trajectories.

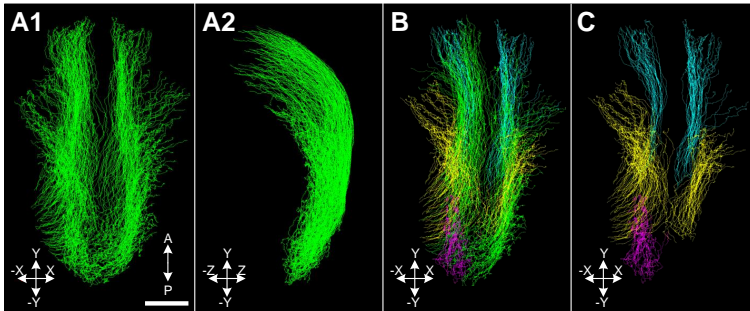


Figure 4.1: Highly responsive 3D rendering of trajectory data allows analyzing the data from arbitrary orientations in an interactive manner. (A1-A2) 3D trajectories of original data from different view angles. (B-C) Visualization of selected groups in 3D rendering. For the visualization, benchmark database `TDB_S2` was used.

This allows the user to freely choose between different levels of detail which is fundamental to get familiar with underlying phenomena and characteristics of the data. Moreover, the interactive 3D visualization is suitable to derive new hypotheses on-the-fly or to interpret upcoming changes in the database from different perspectives simultaneously. To allow focusing on a particular group of objects in isolation, a separate 3D window allows visualizing only a subset of trajectories that were obtained, *e.g.*, by manual selection or based on specific trajectory criteria as described in the following sections in detail (Figure 4.1B,C). Here, the user gets a better clarity regarding the selected groups and is able to inspect them in a stand-alone manner in different levels of detail [348]. The 3D visualization of the tracks in combination with the maximum intensity projection of the raw data (Section 4.1.2) reveals full power to investigate the tracking data interactively.

Example 1

Starting with the three-dimensional representation of the trajectories (Figure 4.1), the aim is to extract three groups manually according prior defined characteristics and spatial location (Figure 4.1C). In this example, one group of straight moving tracks (cyan), one group of tracks drifting to the sides (yellow) and one group that is located at the bottom side of the benchmark on the left side (magenta). Using the 3D representation, the user is able to modify the selection of the three groups from arbitrary perspectives (Figure 4.1A) and select and deselect single trajectories manually, not belonging to the prior defined groups. The process of selection is iterative and not straight-forward. Therefore, fully automatic methods as depicted in Chapter 3 are not feasible due to the absence of meaningful features.

4.1.2 Maximum Intensity Projection with Raw Images

The maximum projection overlay is a combined visualization where for each time point, the detected object centroids from the segmentation are superimposed on the raw image data that was projected along one of the coordinate axes (Figure 4.2A). This enables a detailed consideration of the data on a single object level. Within the maximum projection, the user can easily navigate through the time by scrolling and zooming. This functionality provides an overview of the entire dataset and simultaneously facilitates analyzing object migration at the single object level. Furthermore, the superposition of the whole trajectories with the maximum projection in a backward, forward and combined for- and backward mode reveals additional possibilities in analyzing the tracking data efficiently (Figure 4.2B,C). Additionally, the maximum intensity projection can be helpful to curate tracking errors and therefore assess single object movements visually (Section 4.4). While the maximum intensity projections along the three coordinate axes (Figure D.1) provide a useful way of visualizing the original data for tracking validation, in some cases objects may occlude each other. This occlusion is especially occurring in highly complex databases, where the data points are distributed over a lot of spatial layers. Often, a priori knowledge (Section 3.2) in the application-dependent context is gained on the basis of such maximum intensity projections. Therefore, this visualization is also well suited to efficiently incorporate this knowledge in the knowledge discovery process.

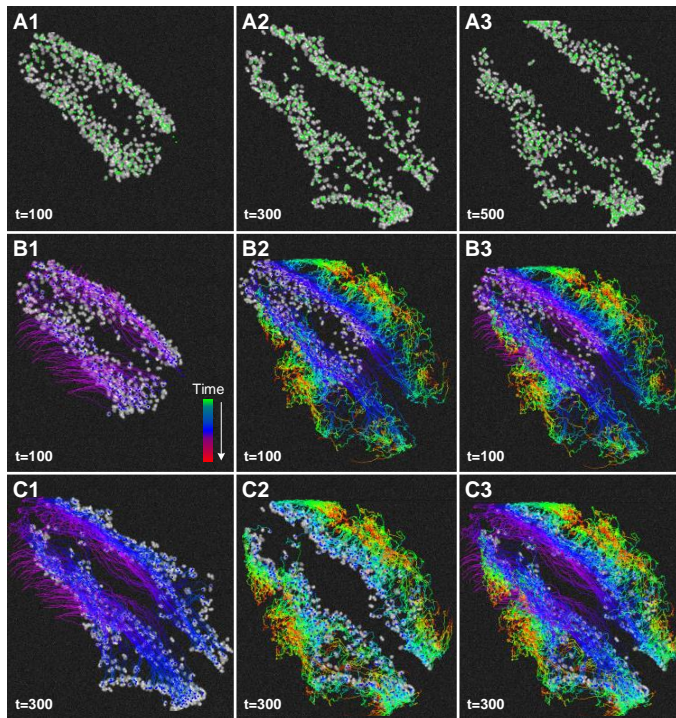


Figure 4.2: Maximum intensity projection overlay. (A) Temporally scrollable maximum intensity projections for benchmark database `TDB.S2` with superimposed segmentation results from detected objects (green dots) displayed for the time point 100 (A1), time point 300 (A2) and respectively time point 500 (A3). (B) For time point 100, the superimposed trajectories in the backward (B1), forward (B2) and back- and forward mode (B3) are visualized. (C) Superimposed trajectories for time point 300. Backward (C1), forward (C2) and back- and forward mode (C3). The color coding indicates the time.

Example 2

Different time points of the simulated benchmark `TDB.S2` (Section 2.4) containing the tracked object superimposed with a projection of the original raw data is shown in Figure 4.2. If the user wants to get an impression of the developing object movement for single objects alone, or in the context of the whole databases, the user can scroll through time to visually follow the migration of the objects (Figure 4.2A1-A3). If the user is further interested in the past or future migration paths of objects

at a given time point, the superimposed backward and forward tracks can be additionally visualized (Figure 4.2B, C). The superimposition of the tracks allows to easily play around and get impressions of movement paths for objects selected at a specific time point (Figure 4.2A).

4.1.3 3D+t Visualization of Migrating Objects

In case of objects migrating in a three-dimensional context, the 3D+t volume rendering visualization of migrating objects (Figure 4.3) is supplemental to the maximum projection overlay (Section 4.1.2) to display the object on a detailed 3D level. Here, arbitrary zooming and scrolling through time allows an efficient handling of the large-scale 3D trajectory datasets on a detailed single object level. On this basis also new prior knowledge (Section 3.2) can be gained by investigating the original data from several perspectives in 3D simultaneously. Additionally, the responsive volume rendering can be used to resolve tracking errors on a single object level (Section 4.4). To exemplary show the 3D volume rendering (Figure 4.3) the dataset *Pm1* from [283] is used because the benchmark databases introduced in Section 2.4.2 do not contain 3D images stacks.

Example 3

To exemplary show the 3D volume rendering (Figure 4.3), the dataset *Pm1* from [283] is used because the benchmark databases introduced in Section 2.4.2 do not contain 3D images stacks. Exemplary, the user wants to get an impression of the migration behavior of either single objects that are visually identified or the entire database. Therefore, the user can scroll through time (Figure 4.3C-E) and follow the single objects from different perspectives by rotating the 3D view. Additionally, the 3D trajectories gives a global overview about the movement paths over the complete time interval. The level of consideration can be changed using the zoom functionality.

4.1.4 Scatter Plot for Track-Based Features

The scatter plot visualization (Figure 4.4) allows plotting arbitrary trajectory features against each other. The advantage is that not only objects and respec-

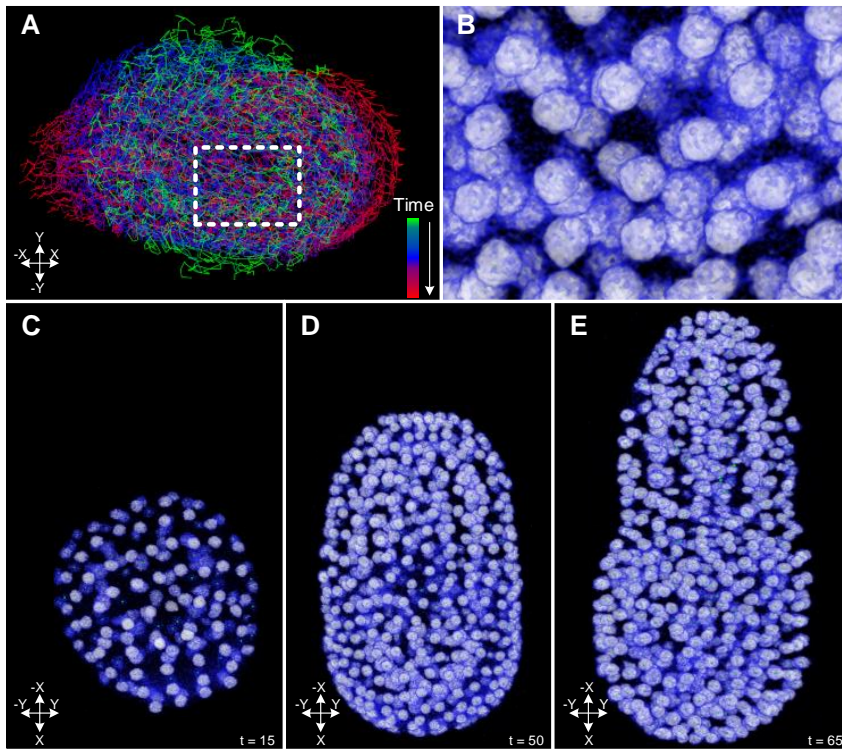


Figure 4.3: Superimposed segmentation result on 3D volume rendering. (A) The 3D trajectory data of the *Phallusia mammillata* (Pm1) dataset from [283, 298]. (B) Zoomed region (indicated by the white rectangle in (A)) with superimposed object centroids on interactive 3D volume rendering of original image data. (C)-(E) 3D rendering visualization for three different time points ($t=15$, $t=50$ and $t=65$).

tively their trajectories can be visualized, but further any chosen characteristic of the tracks can be visualized and interactively modified as well. In Figure 4.4, the result of a filtering approach (Figure 4.4A), a clustering approach (Figure 4.4B, C) and a manual selection output (Figure 4.4D) is visualized. One point within the scatter plot hereby represents one of the given features of one single object and its corresponding trajectory. The different approaches hereby offers several possibilities to extract valuable information out of the trajectory database.

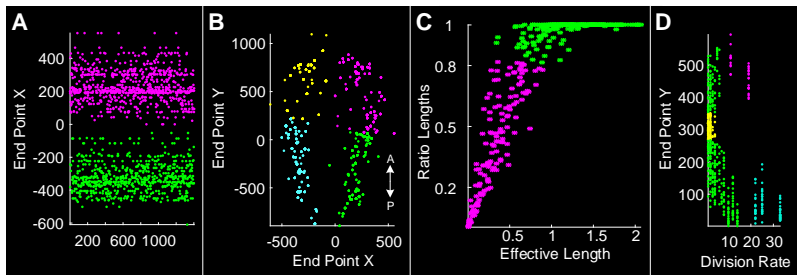


Figure 4.4: Scatter plot for trajectory features displayed using benchmark database `TDB_S2` (Section 2.4.2). Quantitative features associated with each of the trajectories allow feature-based selections. (A) The feature describing the end point of each trajectory is used to filter the trajectories. (B) Shows how trajectories are split into different spatial regions using the end points of trajectories in the investigated time interval using a clustering approach. (C) Two features are used to find two clusters within the scatter plot. (D) Manual selection of four different groups according to the two features describing the division rate of the trajectories and the end point of each trajectory.

In combination with the maximum projection (Section 4.1.2) and the 3D trajectory visualization (Section 4.1.1), this allows not only to consider global behavior and detailed single object movement but to further incorporate any given characteristic of the trajectories within the visual analysis task. Moreover, based on the 2D scatter plot window, feature-based data selection strategies can be applied leading to an efficient integration of prior knowledge (Section 3.2) by selecting only trajectories with specific quantitative movement characteristics.

Example 4

Given the original 3D+t trajectory databases, the user can refine filtering and clustering approaches manually by interactive selection and deselection strategies. Exemplary, Figure 4.4A, B illustrate the separation of trajectory groups according their spatial destination. If the user wants for example to separate two groups according the curvature of their trajectories, a clustering approach applied on two suitable features (ratio lengths, and effective length - listed in detail in Table 3.5) can be applied (Figure 4.4C). Afterwards, the user can refine the group assignment within the border regions where often wrong allocation results occur. Therefore, the prior knowledge of the user is important to manually fine-tune the clustering result using the scatter plot visualization. If the user wants directly to select objects

comprising a combination of spatial destination and moving object characteristics, Figure 4.4D shows the manual selection of three different groups.

4.1.5 Lineage Visualization for Dividing Objects

In case of problem classes containing dividing objects (Table 2.1) the interactive lineage visualization provides a complementary, visually guided possibility to investigate specific characteristics of dividing objects (Section 3.3.8).

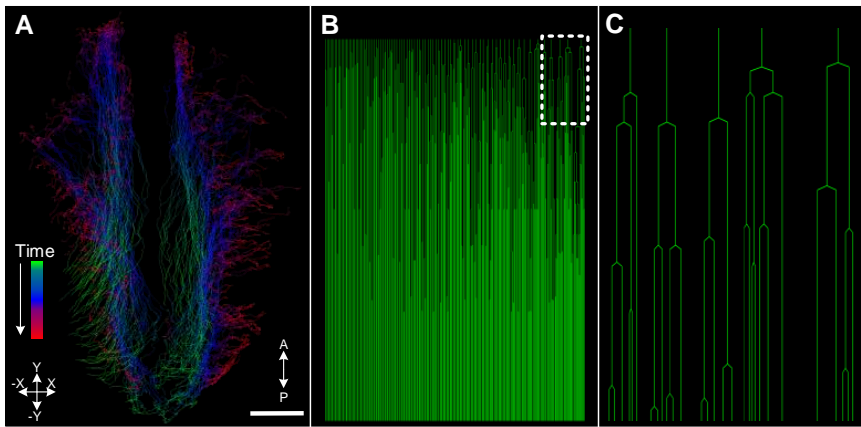


Figure 4.5: Visualization of dividing object characteristics. (A) The original 3D trajectory data of benchmark database `TDB_S2` color-coded using the time. (B) Lineage of the trajectories describing the object division events that occur at different time points. Here, the branches depict the division events. (C) Zoomed region (marked with a with rectangle in (B)) out of the whole lineage displayed in (B).

For each trajectory database (Figure 4.5A) containing dividing objects, the lineage visualization is displaying the division events on a detailed time scale (Figure 4.5B,C). Through interactively zooming and focusing on specific division events, the characteristics of the dividing objects can be effectively addressed. Additionally, the selection possibilities introduced in Section 4.2.1 allow focusing on single division characteristics and linking these events with the 3D trajectory data to gain knowledge about the underlying processes.

Example 5

Exemplary, the user wants to investigate benchmark database `TDB_S2` (Figure 4.5 A) with the specific focus of getting an impression of the division characteristics of the objects contained in the database. Therefore, the lineage (Figure 4.5B) of the original 3D trajectory visualization (Figure 4.5A) is calculated and visualized. Here, the user can see that the division events are not fully synchronized over the database. Furthermore, the user can use the zooming and focusing abilities to investigate division events at specific time points on a more detailed level (Figure 4.5C) allowing to gain additional knowledge about the division characteristics contained in the database.

4.1.6 Modular Incorporation of New Visualizations

The generic design of the visualization framework allows integrating new visualization windows, representing specific visual aspects of the dataset, in an effective way with custom-tailored data representations. The new visualization windows are automatically connected to all other windows to use the full power of interactive modifications by using the existing selection and propagation strategies. The easy integration of new visualizations in the existing visualization framework (see Section 5.1 for details) leads to full power of incorporating new a priori knowledge (Section 3.2) efficiently.

4.2 Virtual Dissection of Large-Scale Object Tracking Data

To analyze specific parts of the trajectory data in isolation and virtually dissect the trajectory database, the user needs the ability to interactively extract groups of interest in any of the data representations. In addition to the quantitative feature-based description of trajectory properties, manual selection possibilities in arbitrary visualization windows are required to interact with the data in an intuitive and efficient way [348].

4.2.1 Process of Interactive Selection in Trajectory Visualizations

All different visual representations of the tracking data introduced in Section 4.1 serve as a basis for an interactive selection process (Figure 4.6) to facilitate the incorporation of existing prior knowledge (Section 3.2). The prior knowledge can be either integrated using feature-based techniques or through direct manual allocation of the information (Section 3.1).

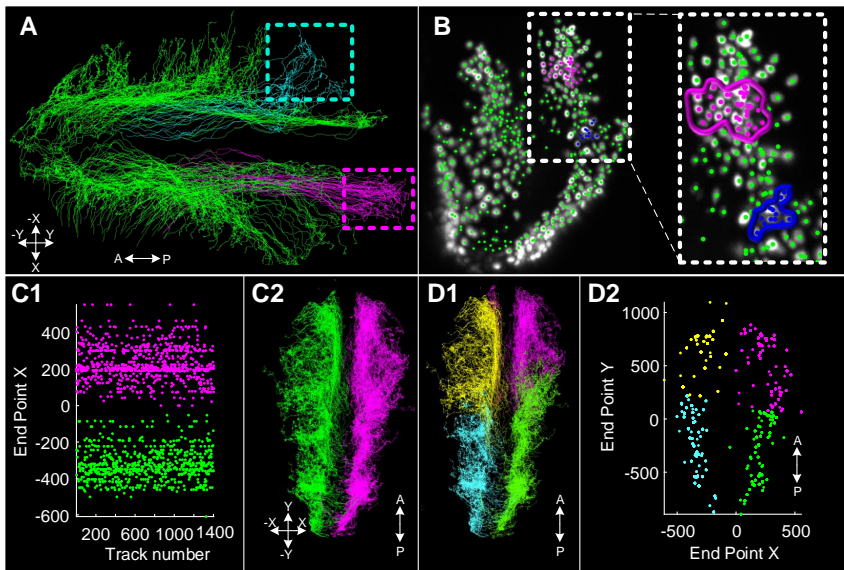


Figure 4.6: Interactive selections possibilities demonstrated using benchmark database TDB.S2. (A) Interactive freehand selection of two groups (cyan and magenta) within the 3D trajectory visualization. (B) Interactive freehand selection of arbitrary shaped groups within the maximum intensity projection overlay. (C1) Quantitative features associated with each of the trajectories allows feature-based selections. (C2) Feature-based separation of the left and right part. (D) Cluster algorithms can be used to automatically group the data. The example shows four identified clusters using the end point locations in the XY-plane at a selected time point as a 3D rendering (D1) and a scatter plot (D2).

Using the 2D and 3D visualizations (Section 4.1.1, Figure 4.6A) as well as the maximum intensity projection overlay (Section 4.1.2, Figure 4.6B) a freehand tool allows selecting single objects at specific time points or entire trajectories

from an arbitrary perspective (Figure 4.6A, B). Moreover, it is possible to perform the selection based on any of the precomputed trajectory features (Section 3.3.3) using a 2D scatter plot visualization (Section 4.1.4). In addition to the manual selection, the feature-based selection possibility allows to readily apply data mining methods such as clustering, interactive filtering and classification to group the dataset based on quantitative feature descriptions of the trajectories. The simplest way of feature-based selection process is to use a manually specified threshold on one of the single features represented in the scatter plot (Section 4.1.4). Generally, the user can apply the selection and deselection process in an arbitrary order, allowing for a highly flexible manual refinement of the selected group of interest. Furthermore, intermediate selection steps may also be based on data mining methods as described in Chapter 3.

Example 6

Using the 3D trajectory visualization, the user can select different groups by freely chose rectangles covering the spatial coordinates of the trajectories (Figure 4.6A). If the user is interested in groups of objects that occur at a specific time point in spatial locations of interest, a freehand selection strategy can be used within the maximum projection overlay (Figure 4.6B). The groups can be refined by selection and deselection strategies to iteratively approach the desired group memberships. If the user is furthermore interested in the separation of trajectories according one predefined characteristics, a manual threshold can be applied. In Figure 4.6C, such a manual threshold (threshold = 0) is applied on the horizontal coordinate of the TDB_S2 benchmark database (X-position), in order to separate the left and the right parts of the trajectories. If the user is interested to automatically find groups of interest according some characteristics, a clustering approach with subsequent manual refinement can be used. A clustering process is exemplary depicted in Figure 4.6D showing four identified clusters using the end point locations in the XY-plane at a selected time point with a predefined number of four clusters.

4.2.2 Online Propagation in Connected Trajectory Visualizations

To unveil the full scope of functions to integrate prior knowledge from different sources and various characteristics, an online propagation of intermediate selection results to other visual representation is developed. Here, all data visualization windows are synchronized in the way that the effect of a selection or

deselection is directly updated in all visualization windows. The synchronization of multiple views allows increasing the objectivity through *e.g.* performing the initial selection based on a given precomputed quantitative trajectory feature (Section 3.3.3) using the 2D scatter plot visualization or by using fully automatic methods such as clustering [348]. Detailed refinement steps can then be applied in other visual representation by incorporating prior knowledge available in the existing visual representations.

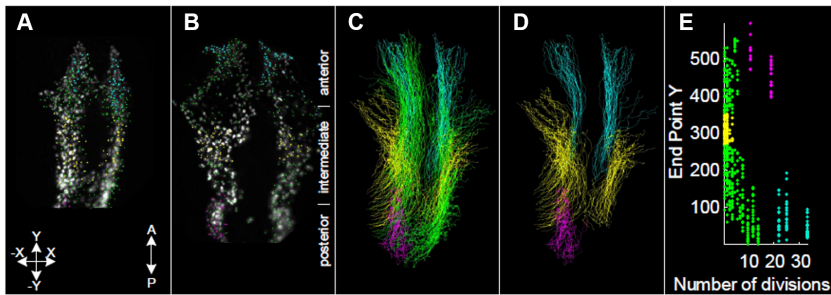


Figure 4.7: Online propagation of selections using the TDB_S2 benchmark database. Here, the selection of the three groups color-coded in magenta, yellow and cyan was done using freehand interactive selection within the scatter plot illustrated in (E). The non-selected trajectories are color-coded in green. All visualization windows allow selecting groups of interest using freehand selection tools. From the left: maximum intensity projection overlay of the raw images and object centroids for two time points (A, B); tracks of all cells in 3D (C); subset of selected tracks in 3D (D); scatter plot of track-based features (E). All visualization windows (A-E) are synchronized to obtain consistent selections in all views (see corresponding color-code of panels A-E) [348].

Example 7

Exemplary, Figure 4.7 illustrates three different selection approaches applied to the benchmark TDB_S2 using the visual representation of a maximum intensity projection (Figure 4.7A), a 3D trajectory visualization (Figure 4.7B) and a 2D scatter plot (Figure 4.7C). The initial selection comprising three groups (magenta, yellow and cyan) was done using the 2D scatter plot (Figure 4.7C). The user can now directly observe the effect of the performed selection strategy to all other visual representations. If the user is additionally interested to refine the executed group selection, the proposed selection and deselection strategies can be applied in an arbitrary order to the single visual representations. The initial group selection can also be done

using automatic approaches described in Chapter 3 with a subsequent manual refinement on the basis of visual inspection.

4.3 Interactive Visual Data Exploration in Hierarchical Analysis Processes

To keep track of selections and analytical operations applied on the trajectory databases, all results of the interposed analysis steps can be registered and visualized within a hierarchical tree structure depicted in Section 4.3.1–4.3.3.

4.3.1 General Scheme for the Hierarchical Analysis Process

The KD-process (Section 3.1) comprises an efficient step-by-step analysis of the data with several analysis steps that are sequentially applied to the data. Furthermore, analysis steps are often applied only to a specific part of the data leading to the branching of the complete analysis pipeline [348]. To interactively handle complete analysis pipelines in an efficient way and to keep track of different selections and operations applied on the data, the results of all intermediate analysis steps of the KD-process can be recorded and visualized in a hierarchical selection tree structure (Figure 4.8). All nodes of the tree can be merged to new groups of interest. To be able to perform multiple analysis tasks simultaneously, operations can be selectively applied on specific branches of the tree, e.g., to separately focus on border regions or left/right parts of the trajectory dataset. Hereby, in each node of the tree, all visualization possibilities developed in Section 4.1 can be used to visually represent the data leading to an enormous flexibility in addressing and handling the data. Besides visualizing intermediate results, all performed steps can be interactively modified and it is possible to combine different nodes to obtain a joint selection. The interactive visual tree structure tool provides a complete overview of the whole analysis pipeline, allowing users to concentrate only on a subset of the data by interactively choosing the desired tree branch. Furthermore, each step within the analysis pipeline can be interactively corrected afterwards resulting in the effective integration of complementary prior knowledge of additional experts for example (Section 3.2).

4.3.2 Visual Realization of the Hierarchical Analysis Process

The selection possibilities described in Section 4.2.1 in combination with the interactive hierarchical tree structure (Section 4.3.1) allow to virtually dissect an entire trajectory database and to quantitatively analyze a subset of the data. To unveil the full interactive potential to incorporate diverse existing prior knowledge all visualization are connected to each other (Figure 4.8) and modifications are therefore automatically propagated through the whole tree [348].

Example 8

Exemplary, Figure 4.8 illustrates several possibilities of using the visual hierarchical analysis process. Depending on the investigation task of the user, the hierarchical tree structure allows to depict several approaches of the KD-process (Section 3.1) In Figure 4.8A, the groups are selected manually using existing prior knowledge. In the first layer, the two sides of the benchmark are separated using a filter approach on the basis of the end points of the trajectories. In the following second layer, two groups in the front part are selected manually in the trajectory window, one group with straight moving trajectories and the other group with trajectories that moves to the sides. Moreover, Figure 4.8B illustrate the filtering on time series features. Here, the y-coordinate of the trajectory position is used the time series feature. To automatically find subgroups with similar characteristics, Figure 4.8C1 exemplary cluster the trajectories using their start points in X,Y,Z-space. A further clustering on the feature describing the mean curvilinear speed is illustrated in Figure 4.8C2. Moreover, a filtering approach applied on single features description of trajectories is illustrated in Figure 4.8D using the feature ratio lengths to separate two groups.

4.3.3 Interactive Hierarchical Analysis Process and Transfer of Analysis Pipelines

Moreover, the interactive visual tree structure serves as a generalized template of complete analysis pipelines and can be interactively applied to new databases. Reproducing an entire analysis pipeline on a new dataset can be used to efficiently extract the same information in different datasets. This allows quantifying variations of relevant groups of objects in different scenarios

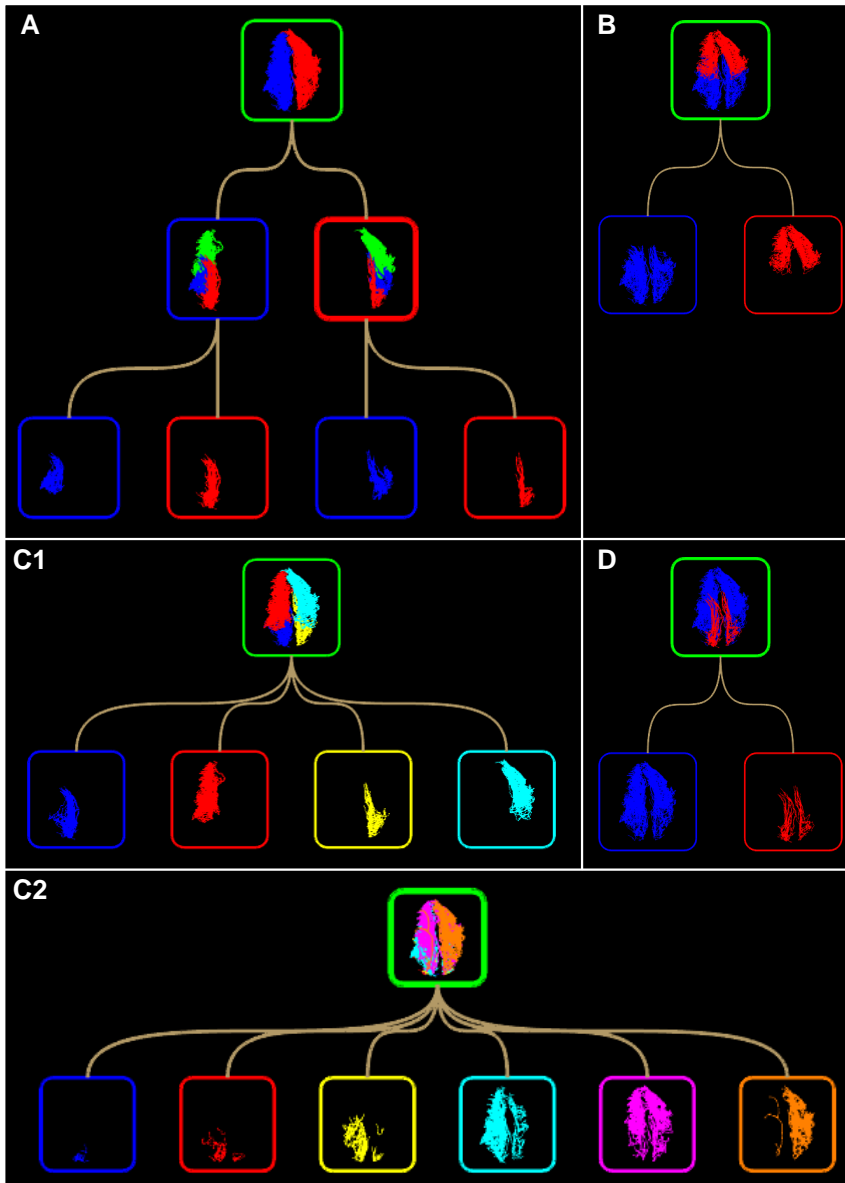


Figure 4.8: Hierarchical selection tree structure using benchmark database TDB_S2. (A) Hierarchical manual selection, (B) time series filtering using the Y-position of the trajectories, (C1) clustering using the start points, (C2) clustering using the feature describing the mean curvilinear speed and (D) filtering using the single features ratio of lengths (Section 3.3.3).

(Section 3.3.7). As each analysis step in the tree view allows modifications, parameters can be optionally adjusted and fine-tuned to each single database to compensate the natural variation inherent to different datasets or acquisition.

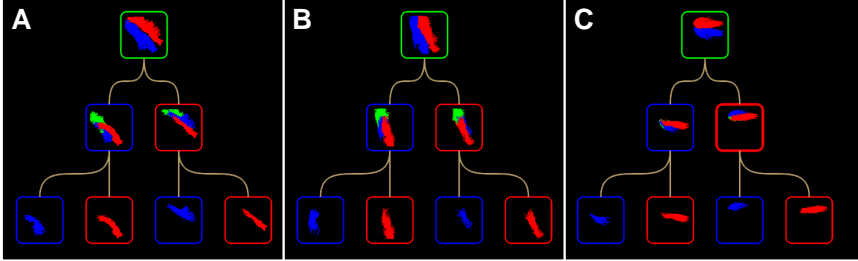


Figure 4.9: Interactive transfer of analysis pipelines. Here different simulated benchmark databases, namely $TDB_S2, 1$ (A), $TDB_S2, 2$ (B) and $TDB_S2, 3$ (C) serve as a basis to extract the same straight moving objects (color-coded in red) separately for the left and right sides within each database.

Example 9

Exemplary, in Figure 4.9A one group comprising a straight movement characteristic is extracted in the first hierarchy layer using the curvature of the tracks. In the subsequent second layer, the extracted straight moving group is additionally separated into two sides of the benchmark databases using a clustering approach based on the spatial destination of the tracks. The designed analysis pipeline is then transferred to two further databases ($TDB_S2, 2$, $TDB_S2, 3$) (Figure 4.9B, C). Here, the same groups are extracted. Furthermore, the user has the ability to manually refine the group extraction results in each layer separately, allowing to adapt the whole pipeline to small inhomogeneities within the new databases.

To evaluate the impact of interactive modifications on the transfer process of analysis pipelines derivatives of the benchmark database TDB_S3 ($TDB_S3, D1$, $TDB_S3, D2$) and derivatives of the benchmark $TDB_S6, D1$ ($TDB_S6, D1$, $TDB_S6, D2$) are used as introduced in Section 3.3.6.

For the benchmark TDB_S3 , Table 4.1 lists the percentage of correctly assigned trajectories to the original groups. It can be observed, that all trajectories are as-

signed the correct group after transferring the analysis pipeline through applying the clustering to the new databases without the need of manual adaptation of the analysis pipeline.

Method	TDB_S3, D1	TDB_S3, D2	TDB_S6, D1	TDB_S6, D2
No interactive modification	100%	100%	55%	40%
Interactive modification	100%	100%	100%	100%

Table 4.1: The impact of interactive modifications required for the transfer of complex analysis pipelines to new trajectory databases comprising inhomogeneities. The percentage of correct assigned trajectories for each two derivatives of the benchmark database TDB_S3 (TDB_S3, D1, TDB_S3, D2) and TDB_S6, D1 (TDB_S6, D1, TDB_S6, D2) is listed with and without interactive modifications.

A more complex example is represented by a transfer of combined region of interest selection with subsequent feature extraction as shown in benchmark TDB_S6 (Figure 3.25B). To distinguish the two groups in database TDB_S6, the correct region of interest has to be chosen in the time interval where the undirected movement behavior occurs. Afterwards, the ratio of effective and total displacement is clustered into two groups. The result of the transfer of this analysis pipeline to the databases TDB_S6, D1 and TDB_S6, D2 is listed in Table 4.1. The higher complexity of the analysis pipeline due to region focused feature extracting and clustering leads to a smaller fraction of correctly assigned trajectories. However, if the region of interest is manually adjusted using the interactive visualization introduced in Chapter 4, all trajectories are assigned the correct groups (Table 4.1). In general, it can be concluded that even more complex analysis pipelines can be efficiently transferred to new databases containing inhomogeneities by using interactive modifications. Less complex analysis pipelines can often be applied without the need of adapting the parameters of the analysis pipeline.

4.4 Interactive Visually-Guided Tracklet Curation

In highly complex trajectory databases (Section 2.1) with an enormous density of objects, a lot of tracking errors occur. Therefore, analysis pipelines (Section 3.1) can often not be applied due to the bad data quality. To handle this circumstance, a subsequent curation after the tracking leads to longer tracks over the complete time span. Complete tracks are required to make reliable assumptions about global characteristics of the moving objects [348]. The knowledge discovery process (Section 3.1) is designed to handle fragmented tracks without the need for an extensive curation of all trajectories registered within the dataset. Therefore, a subsequent allocation process handles the fragmented tracks separately (Section 3.3.2). However, the more complete the tracks are available, the more valid assumptions about global characteristics of the moving objects can be made on the basis of the results. In practice, this means that in a first draft the user can apply the KD-process directly on the original tracking data and get rough insights without the requirement of having complete tracks over the whole time span. If this provides evidence of interesting phenomena the user can additionally apply the curation framework (Figure 4.10) based on the visualization possibilities introduced in Section 4.1 to improve the track quality and examine the findings in more detail to underpin existing findings in a quantitative way. The general scheme of the curation process is depicted in Figure 4.10.

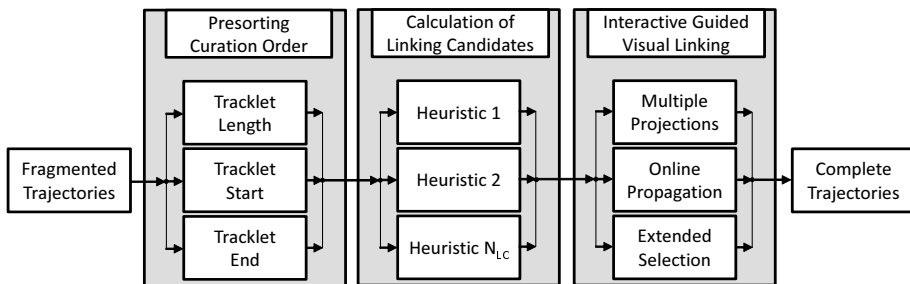


Figure 4.10: General scheme of the manual curation process. Starting with the fragmented trajectories, the fragments are sorted according a predefined order. Afterwards, the possible linking candidates are identified and the interactive guided visual framework assist the user in choosing the correct linking candidate.

In an additional case where only a sparse amount of representative trajectories are corrected manually, the remaining trajectory fragments can be automatically assigned to the extracted groups based on the majority of validated trajectories within the spatial proximity (Section 3.3.2). The interactively guided curation frameworks can be either applied on predefined groups of trajectories or on the entire database. For an efficient manual curation, a quality measure (Section 3.3.2) is used to calculate probabilities for the possible successor and predecessor trajectories on the basis of the local neighborhood, to efficiently guide the user with minimal time effort. The user then only has to identify the best linking candidate visually.

Example 10

Starting from a trajectory database containing fragmented tracking data (see Section 2.2 for possible reasons) the user wants to manually fuse two trajectory fragments that belong to each other with a minimal time effort. In Figure 4.11 a selected trajectory fragment (magenta) and three possible predecessors indicated by the corresponding numbers located next to the trajectories (color-coded in red, green and cyan) are illustrated. The user has the ability to interactively scroll through time in the different visual representations (maximum intensity projection of XY-, YZ- and XZ-plan as illustrated in Figure 4.11A-C and the 3D volume rendering shown in Figure 4.11D) and visually follow the objects to identify the correct linking candidate. Once, the correct linking candidate is found, the user can manually select this candidate by pressing the corresponding number on the keyboard. If the correct linking candidate is not covered within the first three suggested possible predecessors, the user can manually select the correct object.

4.5 Discussion

In this chapter, several approaches to interactively handle prior knowledge within automated analysis routines (Chapter 3) are depicted in a systematic way. The primary focus of the developed algorithms was to systematically embed and use the convenient prior knowledge within the analysis of highly complex 2D+t and 3D+t databases. Due to the fact that prior knowledge often

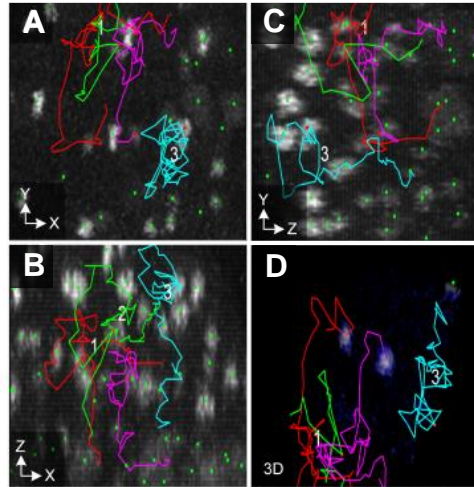


Figure 4.11: Exemplary curation process of erroneous tracks using the interactive curation framework. The chosen curation mode features interactive maximum intensity projections in the XY-plane (A), XZ-plane (B) and YZ-plane (C) as well as a 3D volume rendering (D).

exist on distinct levels depicting several characteristics of trajectory data, different possibilities were developed in this chapter to address the specific patterns in detail. Therefore, a 3D rendering approach, a maximum intensity projection overlay as well as an efficient 3D+t visualization of the migrating objects and an interactive possibility for feature-based track visualization are presented covering a wide range of capabilities to incorporate prior knowledge from different sources. The applicability of the visual interactive approaches to the problem classes defined in Section 2.1 are listed in Table 4.2. Additionally, all these visual representations can be used for an interactive selection process allowing to allocate prior knowledge intuitively to given objects. Therefore, the selection process can be applied to all proposed visual representations of the trajectory databases in the same manner. To additionally unveil the full potential of integrating prior knowledge from different sources and disparate characteristics, the selections done in one of the visual representations are propagated to all other visualizations allowing to observe directly the effect of a prior knowledge integration in the different aspects for an increased objectivity. The online prop-

agation of the intermediate selection results to all visual representations therefore enables a detailed incorporation process of prior knowledge from different sources. To depict even highly complex analysis pipelines comprising several hierarchical layers of analysis steps with a strong relation between the single analysis layers, an interactive visual tree structure was developed. This interactive visual tree structure provides a complete overview of the whole knowledge discovery process allowing to keep track of different analysis steps and propagated selections with the ability to modify intermediate analysis steps at any time by incorporating additional prior knowledge. Also, complete analysis pipelines can be interactively transferred to new trajectory databases using the tree structure to extract the same relevant information in several databases efficiently.

However, there is a drawback to automatically adapt the transfer process in the way that slight inhomogeneities within the databases are handled automatically to reduce the interactive manual effort. Therefore, specific algorithms that are able to automatically detect and alleviate the inhomogeneities are required for further improvements to achieve a highly automated transfer of analysis pipelines. In presence of less or no prior knowledge, fully automated analysis strategies provided in the previous chapter are not suited to investigate the data in detail. The interactive visual framework developed in this chapter however provides the ability to visually access the data and get familiar with different aspects from several perspectives allowing to gain new knowledge about the underlying effects by iteratively and interactively accessing the data. Furthermore, application dependent prior knowledge tailored for a specific context is often based on highly specific visual representations of the data. Therefore, the modular structure of the interactive framework allows to integrate new visualizations of the trajectory data in a short period of time using the predefined standardized interface introduced in Section 5.1. In highly complex trajectory databases a lot of tracking errors occur leading to the fragmentation of the trajectories hindering the application of analysis pipelines and the integration of prior knowledge tailored to the occurrence of complete trajectories. To cope with such highly complex and fragmented trajectory databases, a newly developed interactive curation scheme is introduced in this chapter, allowing to generate highly reliable linkings of the fragmented trajectories leading to repaired trajectories spanning the whole time interval. On this basis, the knowl-

edge discovery process developed in the previous section can be applied to the curated trajectories abetted by the interactive visual modifications introduced in this chapter. In future work, the quality of tracking algorithms for highly complex problem classes has to be increased to efficiently use the methodology introduced in this thesis with a minimal manual curation effort.

Visual representation	Sec.	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20
3D trajectory rendering	4.1.1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Maximum intensity projection	4.1.2	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
3D+t volume rendering	4.1.3	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Track-based features	4.1.4	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Lineage visualization	4.1.5								x	x	x	x			x	x	x	x	x	x	x
Hierarchical data exploration	4.3	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Tracklet curation	4.4			x		x		x		x	x	x	x			x	x	x	x	x	x

Table 4.2: Applicability of visual representation of 3D+t trajectory data for problem classes defined in Section 2.1. Here Problem class 01 till Problem class 20 are represented by the columns numbered from 01 to 20. A cross hereby indicates that the visualization is well suited for the corresponding problem class. A detailed definition of the problem classes is listed in Table 2.1.

5 New Implementations and Software Tools

The following sections delineate the detailed implementation of all developed methods introduced in Chapter 2, 3 and 4, handling the integration of existing prior knowledge in the discovery of phenomena in existing real-world large-scale 3D+t datasets. All of the proposed trajectory analysis methods described in Chapter 3 and all interactive visualizations depicted in Chapter 4 were implemented in the open-source MATLAB-toolbox SciXMiner [288] to provide the comprehensive range of trajectory analysis methods for large-scale 3D data to a wide audience. Table 5.1 list the most important functionality implemented for the analysis of 3D trajectory databases. In the following sections, the implementation details for the several functionalities are listed.

Functionality	Implementation	Theory
Handling fragmented object tracking data	5.1.4	3.3.2
Quantitative trajectory features	5.1.3	3.3.3
Feature-based group extraction	5.1.3	3.3.5
Quantitative lineage analysis	5.1.3	3.3.8
3D rendering of large-scale trajectory data	5.1.2	4.1.1
Maximum intensity overlay	5.1.2	4.1.2
Time resolved 3D visualization	5.1.2	4.1.3
Virtual dissection of tracking data	5.1.2	4.2
Hierarchical analysis process	5.1.5	4.3
Interactive guided curation process	5.1.4	4.4

Table 5.1: Overview of the implementations with the corresponding section of the methods described in Chapters 3 and 4.

5.1 EmbryoMiner: A new Toolbox for SciXMiner

5.1.1 Overview

To visualize and analyze large-scale 3D+t trajectory databases and to apply semi-automatic selection and data curation strategies by incorporating existing prior knowledge, the SciXMiner toolbox EmbryoMiner [348] was developed (Figure 5.1). Hereby, a bidirectional interface between MATLAB and the

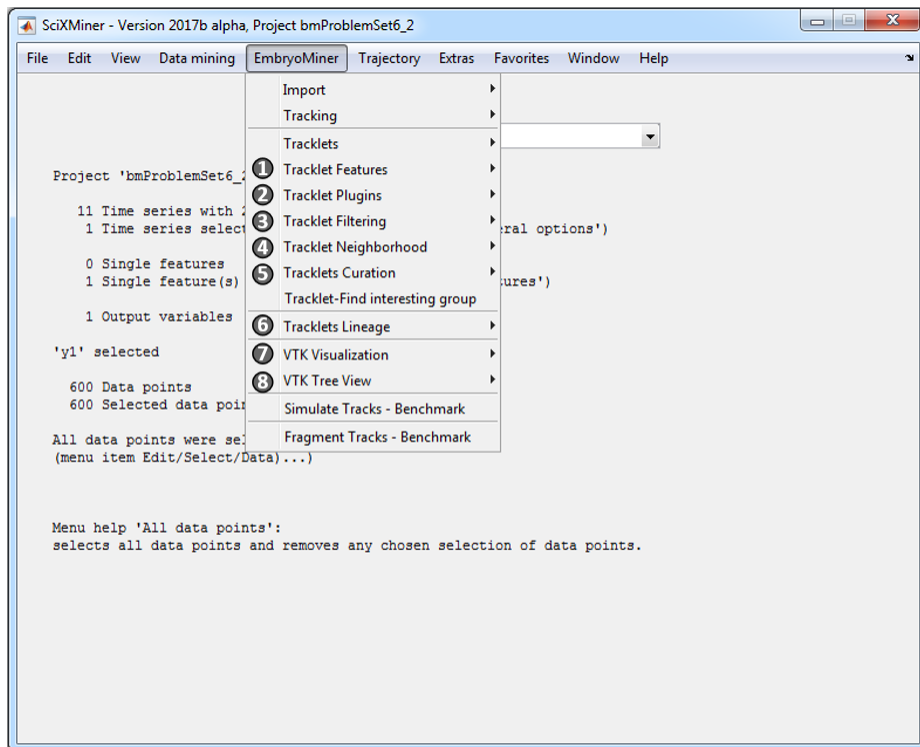


Figure 5.1: Exemplary screenshots of the SciXMiner toolbox EmbryoMiner. The main control elements of the GUI containing feature calculation (1), trajectory plugins (2), trajectory filtering (3), neighborhood calculation (4), curation scheme (5), lineage calculation (6), 3D+t visualizations (7) and hierarchical analysis process (8) are illustrated. The corresponding methodology for (1)-(6) can be found in Chapter 3 and for (7)-(8) in Chapter 4.

Visualization Toolkit (VTK ¹) was implemented using local TCP sockets and custom callbacks allowing to directly interact with 3D+t trajectory data from SciXMiner ². SciXMiner provides full analytical power to explore the data and the VTK interface is optimally suited for the efficient visualization of huge amounts of complex 3D+t data. A detailed description of the software structure and the implementation details can be found in [349]. The high responsiveness of the C++-based implementation, an immediate visual feedback of the render windows and a variety of selection possibilities allow virtually dissecting even unhandy cell tracking data in an intuitive way using the user-friendly graphical user interface (GUI) (Figure 5.1). For all proposed methods the parameters can be interactively adapted using the graphical user interface as shown in Figure E.1 at the appendix. Besides the ability to import and track seed points obtained with XPIWIT ³ [96, 295], importers are provided (Section 5.1.6) for converting the results of several existing tracking tools to a SciXMiner-compatible format. To incorporate new tracking data within the interface, the user has to create a new project in SciXMiner containing the tracking data in a predefined format with cell positions for each time point [348].

5.1.2 Interactive Visualization Framework for Large-Scale Trajectory Databases

To get an easily expandable and modular software interface for the visualization tasks, the VTK dependencies are separated in a C++ interface [348]. Here, the interface is totally independent from MATLAB potentially allowing to efficiently reuse the implemented algorithms with other software tools. To enhance interactivity (Chapter 4) and to incorporate prior knowledge (Section 3.2), a set of interactive trajectory data visualizations are implemented enabling the interactive dissection of huge 3D+t datasets. Hereby, the different types of visualizations illustrated in Figure 5.2 allow to focus on various aspects of the data within the multiple complementary data representations. A detailed software architecture of the visual representations can be found in [349]. The 2D and 3D visualizations created by the bidirectional interface (Section 5.1.1) are inter-connected using algorithmic linking of applied selections within the

¹<http://www.vtk.org/>

²<https://sourceforge.net/projects/scixminer>

³<https://bitbucket.org/jstegmaier/xpiwit/downloads>

software architecture. Selections within one visualization window are then instantly propagated to all other visualization windows (Section 4.2.2). Moreover,

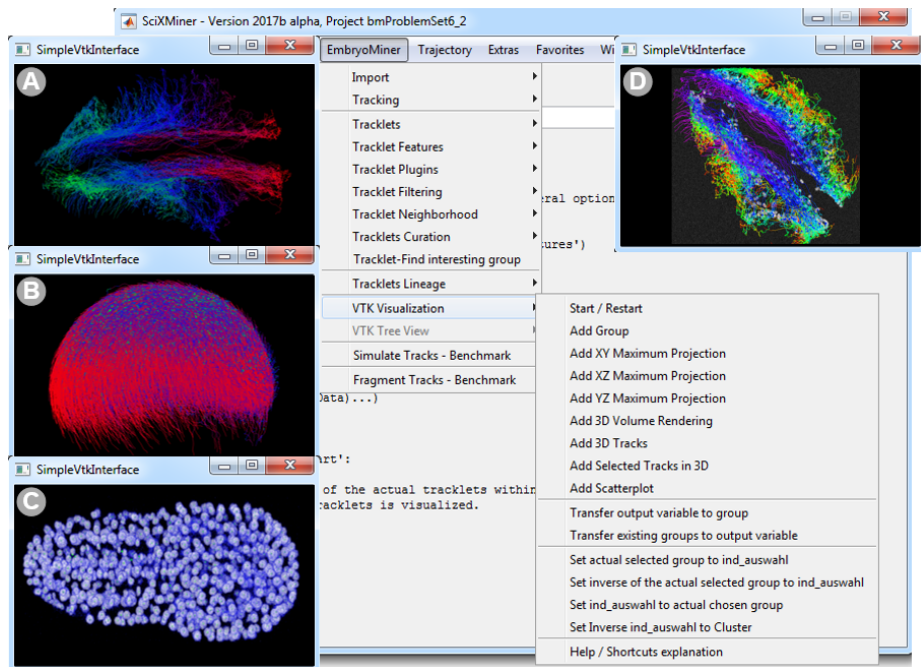


Figure 5.2: Exemplary screenshot of visualization windows. (A) Trajectories of the moving objects in 3D, (B) visualization containing only selected trajectories in 3D for a more detailed analysis of isolated groups of objects, (C) GPU-accelerated 3D volume rendering and (D) maximum intensity projection overlay of the raw images and the tracking results.

custom callbacks within the VTK interface are used to visualize the trajectory data in multiple formats allowing to outsource the functionality for better modular use. Here, the visualized trajectories are addressed using IDs, making it possible to easily request and change properties of, such as opacity, color and others (Section 4.1.1) and to create selections of groups of interest. Additionally, new application-dependent visualization windows, that are automatically connected to the existing windows, can be integrated using the generic design of the visualization framework [348, 349].

5.1.3 Feature-based Description of Movement Behavior

All trajectory features listed in Table 3.5 are newly implemented and computed entirely in the MATLAB-based data mining software SciXMiner, allowing users to make use of the MATLAB scripting language and a large repository of available algorithms. Once a project has been loaded to SciXMiner, all implemented features can be added to the project via the graphical user interface (Figure 5.3) interface and will be computed instantly.

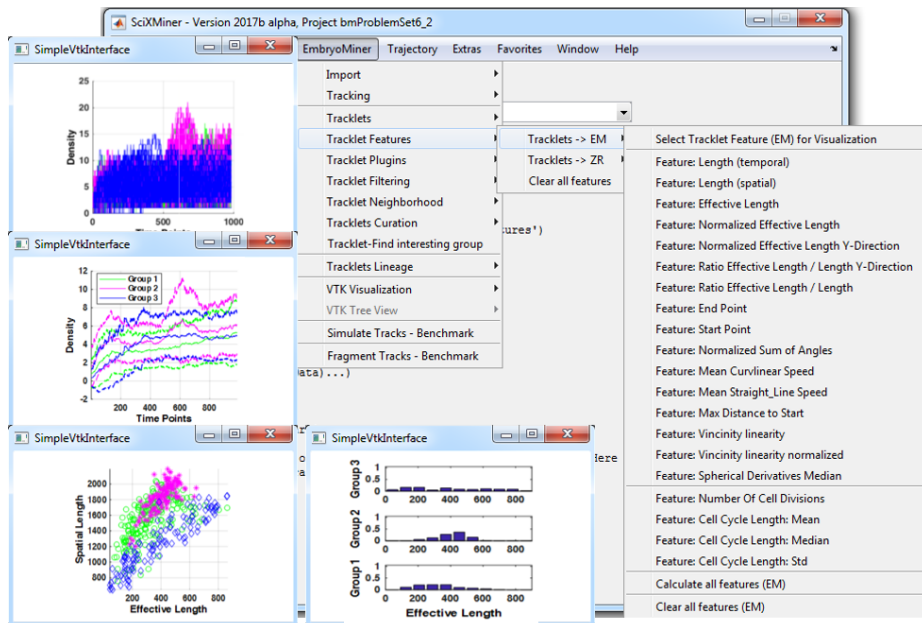


Figure 5.3: Exemplary, an overview of trajectory features already implemented in the EmbryoMiner framework is illustrated.

Optionally, prior knowledge-based new trajectory features can be additionally integrated in the EmbryoMiner framework using a pre-defined software interface. The software interface allows users to integrate new features without being in expert in the software architecture of the whole framework. Moreover, the user can also calculate features not only based on the spatial movement paths of the objects but on the division characteristics of the objects. Therefore, the lineage calculation and visualization of characteristics of dividing object,

such as division angles described in Section 3.3.8 can be applied to the data using the GUI (Figure 5.4).

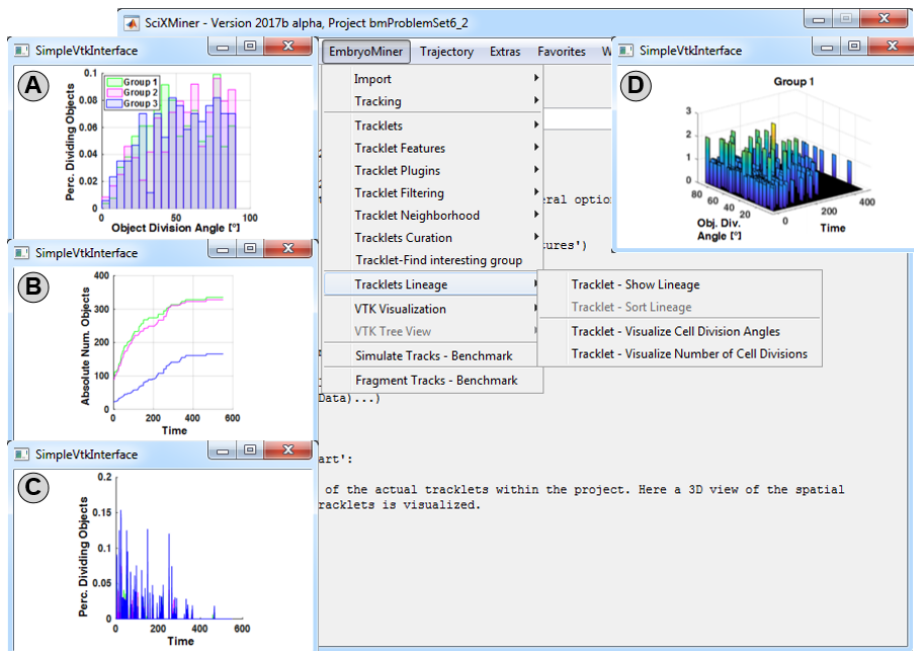


Figure 5.4: Functionality of lineage analysis in trajectory databases containing dividing objects. (A) Visualization of object division angles. (B) Absolute number of objects for all time points. (C) Percentage of dividing objects over the complete time. (D) Object division angles over time.

5.1.4 Interactive Manual Curation Framework

For the interactive manual curation scheme, the maximum intensity projection overlay and the 3D GPU-accelerated volume rendering (for implementation details see Section 5.1.2) are used. The presets for the curation can be done manually using the EmbryoMiner GUI (Figure E.1). Here, the user can choose the number of displayed neighbor candidates and set the threshold for the quality measure (described in Section 3.3.2). Depending on the respective scenario the user can adapt the settings for the curation. Furthermore, the manual curation scheme is launched via the GUI and the user can select the correct linking candidate using the corresponding keyboard shortcuts options. For the

actual implemented version of the curation framework the positions (X-,Y- and Z-coordinates) are incorporated within the quality measure providing the possible linking candidates. Further features (e.g. fluorescent channels or geometrical characteristics from the segmentation process) can be implemented in the curation scheme. A detailed overview of the implementation details can be found in [349].

5.1.5 Interactive Hierarchical Analysis Process

The visual hierarchical analysis process is an important possibility to graphically apply whole analysis pipelines with the ability to refine intermediate results by manually integrating further prior knowledge. The methodological concept is described in Section 4.3. Here, all proposed functionality is implemented in the SciXMiner-toolbox EmbryoMiner. The graphical hierarchical visualization can be started in the GUI (Figure 5.1). New nodes within the visualized tree can be added manually by clicking on the parent node. For each node an additional window is opened where several methodological approaches can be chosen to extract groups of trajectories (manually, filtering and clustering). Single branches of the tree can be locked, so that modifications are not further propagated through the whole tree. The locking of the tree allows to quickly change some intermediate results. When unlocking the tree, these changes are instantly propagated to the whole tree. All analysis steps are saved in an internal format allowing to transfer the complete analysis pipeline to new databases (Section 3.3.6).

5.1.6 Importing Existing Tracking Databases

To integrate existing tracking approaches in the field of developmental biology, the EmbryoMiner framework provides importers for the tracking data from BioEmergences [283], TGMM [141], TrackMate [292] and all algorithms that produce tracking result in the Cell Tracking Challenge Format [135, 350] (Table 5.2). Importing already existing tracking data from other frameworks allows to apply the proposed approaches of handling large-scale trajectory data introduced in Chapter 3 and 4. In future releases, additional importers for application-specific tracking frameworks can be implemented leading to the

efficient analysis of tracking data in a wide range of applications. Exemplary, Figure 5.5 shows imported tracking databases from BioEmergences (Figure 5.5 A-E) and the result of the Cell Tracking Challenge Format (Figure 5.5F-M).

5.2 Comparison to Existing Software Solutions

To compare the SciXMiner-toolbox EmbryoMiner [348] to further existing software tools for trajectory visualization and analysis, Table 5.2 lists several evaluation criteria [84, 349, 351]. Additionally, four evaluation criteria are added that are substantial for the interactive analysis and visualization of large-scale trajectory databases. One of these additional evaluation criteria is the possibility for hierarchical analysis processes required for complex pattern extraction and the transfer of analysis pipelines (*Hier.*). Multi-linked trajectory visual representations (*Mult.*) is another criterion that is necessary for efficient analysis. Moreover, the ability of finding interesting groups using clustering (*Clust.*) and the efficient handling of error-prone fragmented tracking data (*Frag.*), making the handling of a wide range of applications possible, is required as additional validation criterion.

Name	Usab.	Adj.	Open	Ext.	Appl.	Hier.	Mult.	Clust.	Frag.
Fiji	••	•••	••••	••	••	•	•	••	•
CATMAID	•••	••	••••	••	••	•	••	•	••••
Mov-IT	•••	•••	••••	••	•••	•	••	•	•••
MATLAB	•••	•••	••	••	•	••	•	•••	••
V-Analytics	•••	••	••	••	•••	••	•••	•••	•
EmbryoMiner	•••	•••	••••	••	••••	••••	••••	••••	••••

Table 5.2: Comparison of trajectory visualization and analysis software tools. The software tools Fiji [294], CATMAID [300], Mov-IT [283], MATLAB, V-Analytics [274] and EmbryoMiner [348] are compared. The criteria for evaluation are combined from [84, 349, 351] and include usability (*Usab.*), adjustability (*Adj.*), open-source (*Open*), extensibility (*Ext.*), applicability (*Appl.*), the possibility for hierarchical analysis strategies (*Hier.*), multiple linked trajectory visualizations (*Mult.*), clustering (*Clust.*) and the possibility to handle fragmented tracking data (*Frag.*). Each of the properties is rated from one to the maximum of four points. As already described in [84], it should be noticed that the assessments are done objectively. Each of the single applications comprises its own benefits and weak points in certain areas of application ([84, 351]).

Software tools that are mainly appropriate only for tracking purpose without trajectory analysis (CellTracker, CellProfiler Tracer, LineageTracer, ImarisTrack, GMinPro, ParticleTracker) are excluded from the comparison table. The comparison in Table 5.2 suggests the strengths of EmbryoMiner covering the whole range from hierarchical analysis methods up to the handling of fragmented tracking data and multi-linked visual representations. Future work for EmbryoMiner has to focus on the automation of the transfer of whole analysis pipelines. Moreover, the 3D volume rendering views for the microscopy data has to be improved allowing the online scrolling through the images stacks.

5.3 Discussion

Chapter 5 described the detailed implementations of the trajectory knowledge discovery methods and visual representations described in Chapter 3 and 4. The open-source MATLAB-toolbox SciXMiner serves as a basis for the implemented methods, where the extension package EmbryoMiner contain the whole implemented functionality described in this thesis. The overall functionality described in Chapter 3 and 4 can be used in menu callbacks in SciXMiner allowing the application-independent use of the proposed methods to analyze large-scale trajectory databases. However, for a detailed modification and extension of the core functionality of the EmbryoMiner toolbox programming expertise in MATLAB is required. To evaluate the performance of the implemented framework, entire zebrafish embryos with up to at least 25,000 objects per frame were used for testing purpose. The framework remains highly responsive also for such large-scale trajectory databases. All proposed methods in this thesis thus can be applied using a standard workstation. For databases containing more than 50,000 objects per frame, a more powerful workstation with sufficient memory and a suitable graphic card is required to fluently visualize the 3D volume rendering. As most analysis pipelines focus on a specific region of interest, spatio-temporal filtering approaches prior to the actual analysis can be used to reduce the object amount to a feasible range.

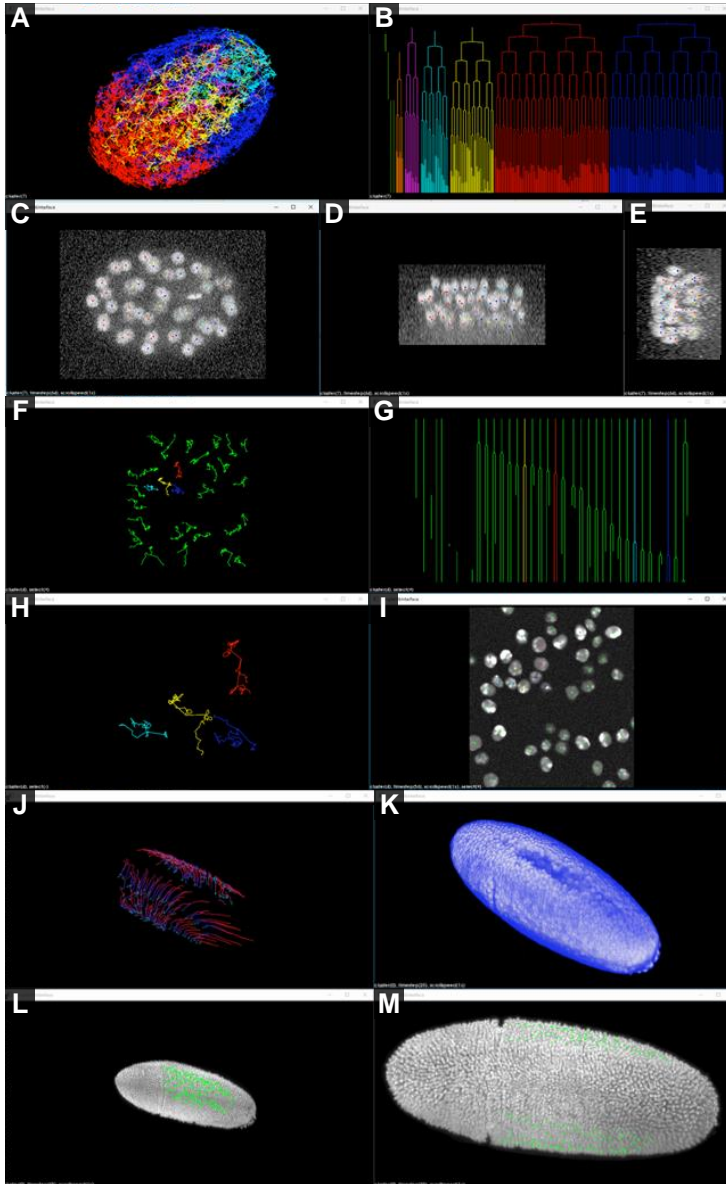


Figure 5.5: Exemplary imported databases from the Cell Tracking Challenge [135, 350]. (A)-(E): *C.Elegans* dataset, (F)-(I): *Fluo-N2DH-SIM* dataset, (J)-(M): *Fluo-N3DL-DRO* dataset.

6 User Study To Evaluate the Functionality of The Knowledge-Discovery Framework For Large-Scale 3D+t Tracking Data

A first evaluation of the knowledge discovery methods for large-scale 3D+t trajectory datasets (Chapter 3) and the interactive approaches (Chapter 4) implemented in the EmbryoMiner Toolbox for SciXMiner (Chapter 5) is presented in this chapter. The methods provided in Chapter 3 - 5 are evaluated in total because single evaluation, e.g. the pure methodological approaches without the interactivity to apply these approaches to new datasets, would not be reasonable. To provide ground truth datasets for the evaluation, the benchmark databases introduced in Chapter 2 are used. In the presented user study, five different tasks have to be fulfilled by the test users to evaluate the usability in terms of quality and time effort. An additional questionnaire further enables the quantitative description of single aspects of the overall framework presented in this thesis. As described in Chapter 5, there are already other software frameworks containing some of the functionality required for the user study. However, there is no software tool that is able to solve all task within one single workflow (Section 5.2), e.g. the tracking is handmade or the IDs of the trajectories have to be manually assigned to given groups.

6.1 Design of the User Study to Evaluate the Possibilities of the Knowledge Discovery Framework

Before starting the study, all users get a proband information sheet (Figure F.9). Here, the overall context of the study and the detailed description of the executed tasks is described in detail. Furthermore, the subjects get a privacy protection information as shown in Figure F.10. Here, the anonymization and the use of the data in a pure aggregated form is described. Additionally, a declaration of consent (Figure F.11 - F.12) with a summary of the major information about the study is handed over to each subject. An overview about the design of the user study to evaluate the possibilities of the knowledge discovery framework for large-scale 3D+t trajectory data developed in this thesis is presented in Figure 6.1. First, the users get a 20 minute introduction to interac-

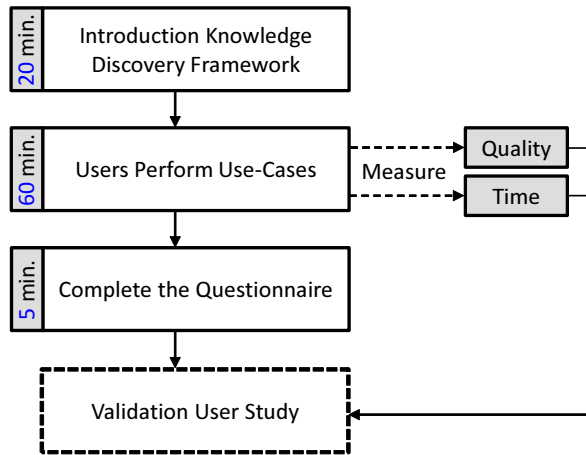


Figure 6.1: Design of the user study to evaluate the knowledge discovery framework for large-scale 3D+t trajectory data. Starting with a 20 minute introduction to the framework, the user gets an overview about the functionality by showing exemplary applications to benchmark datasets. After the introduction, the user has to perform five different use-cases. Here, the quality and time for each use-case is measured and noted. At the end, the users have to complete a questionnaire. Therefore, a time range of approximately 5 minutes is scheduled.

tive knowledge discovery framework (Chapter 3 and 4) and the corresponding

software framework (Chapter 5). Here, the different possibilities of the KD-framework are presented and are exemplarily shown on benchmark databases (Section 2.4.2). After the introduction the users have to perform five different tasks (Figure 6.2 and Figure F.5 - F.8) introduced by the study coordinator. For all performed use-cases the quality and time effort is measured and noted. After completion of the use-cases, the user has to fill in a questionnaire (Figure F.1 - F.2) indicating the end of the user study.

6.2 Content of the User Study

After introducing the test users to the framework as described in Section 6.1, the five use-cases are described one after the other. After successfully performing one task, the next task is introduced. The different tasks model different scenarios in the analysis of real-world large-scale tracking datasets. Hereby, the simulated benchmarks introduced in Section 2.4.2 are used to model the task with the advantage of having the complete ground truth for the underlying groups of objects. A description of the tasks with a detailed problem definition of the simulated real-world application and a listing of all required tools that are developed in this thesis are illustrated in Figure 6.2 (Use-Case 1), Figure F.5 (Use-Case 2), Figure F.6 (Use-Case 3), Figure F.7 (Use-Case 4) and Figure F.8 (Use-Case 5). Here, Use-Case 1 handles the interactive separation of groups according their 3D spatio-temporal movement path. The separation of groups with inhomogeneous movement characteristics in a multi-step approach using a hierarchical procedure is described in Use-Case 2 and Use-Case 3. The allocation of highly specific prior knowledge to single objects in a spatio-temporal context is modeled in Use-Case 4. Furthermore, in Use-Case 5 the feature-based extraction of 3D trajectories exhibiting a predefined spatio-temporal characteristic is investigated. For the subsequent questionnaire (Figure F.1 - F.4) four categories are provided to all test users containing the categories: *Prior experience*, *Usability*, *Potential of methods* and *Potential of interactivity*. Furthermore, an additional category tailored only for biologists is introduced using the framework to interactively investigate their tracking databases in the context of developmental biology. A detailed overview of all questions broken down to the defined categories is listed in Table 6.1.

Use-Case 1: One Layer 3D Trajectory Selection - Middle Complexity

Problem definition

Separate spatially located groups of trajectories in 3D for further ongoing analysis of selected groups in detail.

Task description

The user has to select the two opposing sides of benchmark `TDB_S2` (Section 2.4.2) in the 3D visualization of the trajectories.

Required tools

To successfully accomplish the presented task the following functionality developed in this thesis can be applied:

- 3D rendering of large-scale trajectory data (Section 4.1.1)
- Scatter plot for trajectory-based feature handling (Section 4.1.4)
- Interactive trajectory selection possibilities (Section 4.2.1)
- Online propagation of intermediate selection results (Section 4.2.2)

Target output

The requested output are two selected groups color-coded in green and magenta as shown in Figure 6.2.

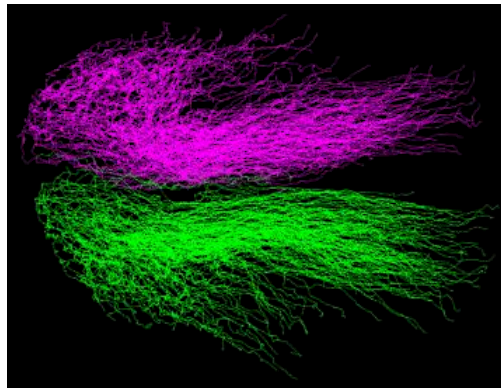


Figure 6.2: Target output to validate the interactive framework: Selection of two opposing groups in 3D (color-coded in green and magenta).

Question	Description
Prior experience:	
Question 1.1	Do you have prior experience in using the software SciXMiner?
Question 1.2	Do you have prior knowledge in analyzing tracking data?
Question 1.3	Do you have prior knowledge in software programming?
Usability:	
Question 2.1	How high was the training effort for the interactive framework?
Question 2.2	How high is the effort of transferring already applied approaches to new trajectory data?
Question 2.3	How big was the knowledge growth in using the interactive framework after applying the proposed methods in a 15-minute session yourself?
Potential of methods:	
Question 3.1	Feature-based description of trajectory data
Question 3.2	Focusing on spatio-temporal regions within trajectory databases
Question 3.3	Interactive curation of fragmented tracking data
Question 3.4	Feature-based extraction of groups using clustering and filtering approaches
Question 3.5	Allocation process to handle fragmented trajectory data within analysis
Question 3.6	Quantitative comparison of multiple groups
Question 3.7	Quantitative neighborhood calculation
Question 3.8	Transfer of analysis pipelines to new trajectory data
Potential of Interactivity:	
Question 4.1	3D rendering of large-scale object trajectory data
Question 4.2	Maximum intensity projection overlay with raw images
Question 4.3	3D+t visualization of migrating objects over time
Question 4.4	Scatter plot for track-based feature selections
Question 4.5	Lineage visualization of dividing object characteristics
Question 4.6	Interactive visual data exploration using the hierarchical analysis process
Question 4.7	Visual guided interactive curation of tracking error
Biology-tailored:	
Question 5.1	Are software approaches already used to analyze existing cell tracking data?
Question 5.1	Is there a need for software approaches handling cell tracking data?
Question 6	For how many percentage of your cell tracking applications is the interactive knowledge discovery framework useful?
Question 7	What are the new emerging possibilities in biological applications, especially cell tracking, using the interactive knowledge discovery framework?
Question 8	Using these new possibilities in cell tracking applications which new conclusions can be made on the newly gained quantitative results?
Question 9	Using the possibilities of the knowledge discovery framework, what may possibly change for future biological experiments compared to the current state?
Question 10	Are there any suggestions for improvement? If yes, which suggestions?

Table 6.1: Questions of the questionnaire for the user study illustrated in Figure F.1 - F.4 evaluating the experiences when using the EmbryoMiner software tool.

6.3 Results of the User Study

The results of the analyzed user study (Section 6.2) are listed in this section. Here, the results of the questionnaire are displayed in Figure 6.4, whereas the measured time effort for the five different use cases is visualized in Figure 6.3. For each use case the time effort and the quality was measured for all six users. The reached quality of the users were throughout all use cases 100%, indicating that the given tasks were perfectly fulfilled from all of the users. The time effort for the different use cases, as shown in Figure 6.3, reveals differences in the required time correlating to the prior experience of the users in software programming (Result of the questionnaire in Figure 6.4). The users with high

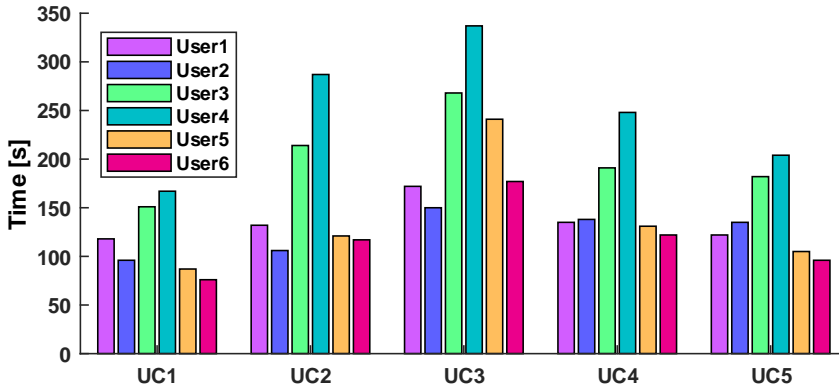


Figure 6.3: Evaluation of the user study containing five different use cases (UC) as described in Section 6.2. In total, six users took part in the user study. Here, the time effort (measured in seconds) for all users within the five different use cases is visualized in bar plots. UC1 handle the spatial separability of groups in 3D. UC2 and UC3 handle multi-step separation of groups. The allocation of specific prior knowledge is handled in UC4. The extraction of groups based on spatio-temporal movement patterns is depicted in UC5.

prior knowledge in software programming (User 1, User 2, User 5 and User 6) were throughout all task faster in comparison to users with less prior experience (User 3, User 4). Exemplary, User 4 had the lowest prior knowledge in software programming and also the highest time effort in all use cases. However, the maximum time effort for the most complex use case (Use-Case 4) was

only 337 *sec* in total. The fastest application to the use cases ranges from 76 *sec* for the easiest use case (Use-Case 1) to 172 seconds for the most complex use case (Use-Case 3) over all test users. For use cases 1,2,4 and 5 the time effort of the users with prior knowledge in software programming was nearly the same with a maximum differences within the users of 42 *sec* in use case 1, 26 *sec* in use case 2, 16 *sec* in use case 4 and 39 *sec* in use case 5. For the use case 3 comprising a multi-step hierarchical analysis of in total 8 different groups, the differences within both, the experienced (User 1,2,5,6) and inexperienced users (User 3,4) was maximal.

Question	Not at all	Little	Neutral	Much	Very much
Question 1.1	3	-	2	-	1
Question 1.2	4	1	1	-	-
Question 1.3	1	-	2	2	1
Question 2.1	4	2	-	-	-
Question 2.2	2	4	-	-	-
Question 2.3	-	-	-	2	4
Question 3.1	-	-	-	1	5
Question 3.2	-	-	1	2	3
Question 3.3	-	-	1	4	1
Question 3.4	-	-	-	-	6
Question 3.5	-	-	-	4	1
Question 3.6	-	-	-	1	5
Question 3.7	-	-	-	2	3
Question 3.8	-	-	-	2	4
Question 4.1	-	-	1	-	5
Question 4.2	-	-	-	2	3
Question 4.3	-	-	-	2	4
Question 4.4	-	-	-	3	3
Question 4.5	-	-	-	4	1
Question 4.6	-	-	-	-	6
Question 4.7	-	-	-	4	1

Table 6.2: Evaluation of the user study. Quantitative amount of answers of all test persons for the questionnaire described in Table 6.1. Here, five Likert categories [352] (Not at all, little, neutral, much, very much) are used. In total, six users take part at the questionnaire. In each of the rows with a total sum lower than 6, the remaining answers were "I don't know".

Regarding the evaluation of the questionnaire, Table 6.2 lists the answers of all users and Figure 6.4 visually illustrated the answers in the corresponding Likert scales. In the following the prominent statements are summarized for the evaluated questionnaire. Hereby, the term "Question X" is abbreviated with "Q. X". Most of the test users did not have any experience in analyzing tracking data (Q. 1.2). Regarding the usability, all of the users pointed out that the training effort for the knowledge discovery framework developed in this thesis was very low (Q. 2.1). Furthermore, the transfer of already applied approaches to new trajectory data was rated as low throughout all users (Q. 2.2). Moreover, all users pointed out that the knowledge growth was very high after using the proposed methods in a 15-minute session themselves (Q. 2.3). Considering the potentials of the proposed methodology (Q. 3.1 - Q. 3.8) all methods were classified with a high potential for future development. The feature-based description (Q. 3.1) and the resulting feature-based extraction approaches, such as clustering and filtering (Q. 3.4), in combination with the quantitative comparison of the extracted groups (Q. 3.6), were rated with the highest potential. With regard to the potential of the interactivity approaches, the hierarchical data exploration process (Q. 4.6) was rated from all users with a very high potential. Furthermore, the remaining approaches were rates with a high and very high potential by all users. Only the 3D volume rendering was rated by one user with a medium potential. A special category (Q. 5 - Q. 7) has addressed only biologists working with embryonic tracking data. The answers of one biological expert working a long time with biological tracking data is listed in Section F.1. Here, the biologist pointed out that at the moment no software approaches at all are used in the field of developmental biology to analyze the existing cell tracking data (Q. 5.1). Moreover, it was pointed out that there is an inherent need in software approaches that are able to handle large-scale 3D+t tracking data from developing embryos (Q. 5.2). The applicability of the KD-framework developed in this thesis to tracking applications in the field of developmental biology was rated to be 100 % (Q. 6) indicating a great number of future usages.

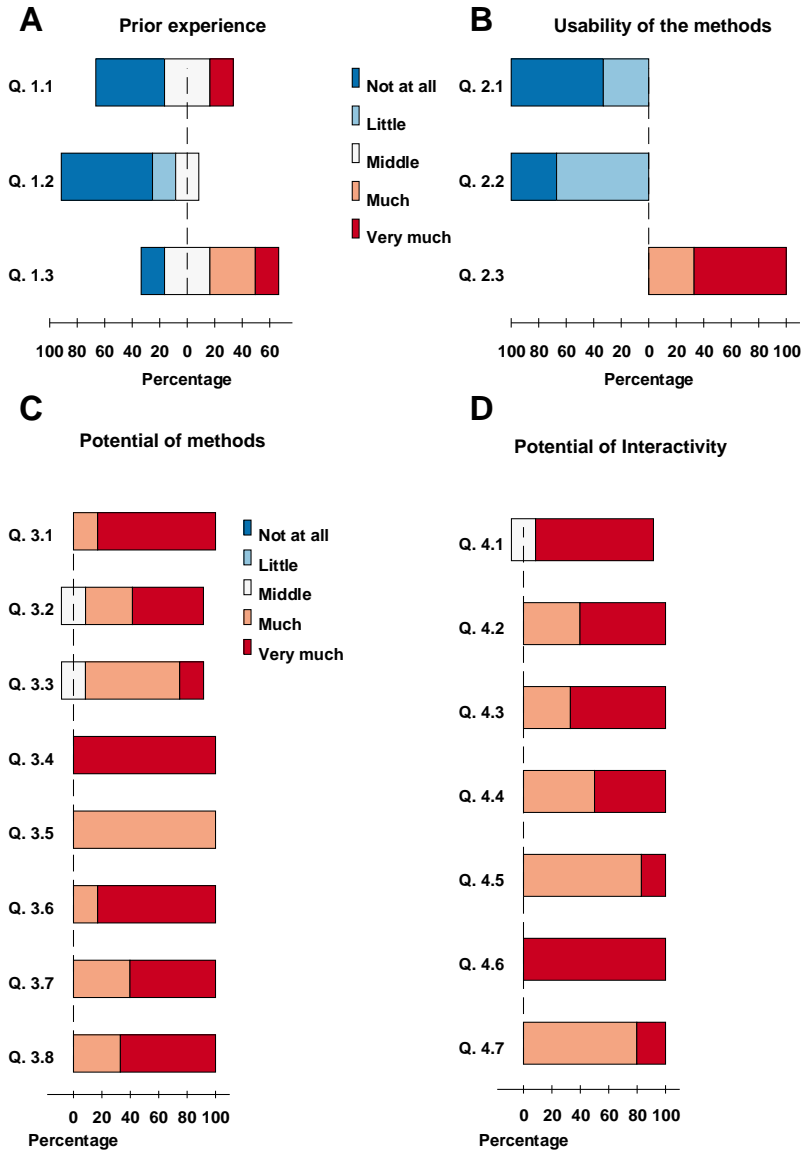


Figure 6.4: Evaluation of the questionnaire for the user study. For four categories (*Prior experience*, *Usability*, *Potential of methods*, *Potential of interactivity*) the answers of the users are visualized in Likert plots using a Likert scale from *Not at all* up to *Very much*. The corresponding questions of the questionnaire are listed in Table 6.1. In total 6 users take part at the study.

6.4 Discussion

In this chapter a study was conducted to evaluate the methods presented in Chapter 3 and 4 in combination with the corresponding implementation in the EmbryoMiner software framework as described in Chapter 5. The study has demonstrated that different task categories in the analysis of 3D+t trajectory data can be done by inexperienced users within a short period of time comprising a high quality of the results. It became apparent that users with a good background in software engineering were faster than users without this background. However, the total time of successfully applying to the given tasks was under 4 minutes for all test users in the study. Furthermore, it was observed that an increase in the complexity of the tasks leads to an increased time effort. However, there was no disproportionate correlation between the complexity and the time effort resulting in a nearly linear relation. Therefore, it is expected that further similar tasks in the analysis of 3D+t trajectory data can be done in similar time ranges of a few minutes. For new tasks in new application fields, the time may vary and through the interactive dissection of the data there might be more time needed. However, all users that participated in this study were inexperienced. It is expected that with a growing routine in using the framework, the time effort decreases quickly. The training effort was rated from all test users as very low and at the same time the knowledge growth was rated very high indicating the applicability of the framework to new areas of application. Concerning the result of the questionnaire regarding the potential of the proposed methods, future research may focus on the automated feature-based group extraction and the automatic comparison of given groups. In this chapter a pilot study was presented. The evaluation led to similar results for all users indicating that also studies with a higher number of subjects may lead to similar results. To investigate the performance of the methods developed in this thesis in more detail, a further study with more users in various application fields may be helpful to also study the impact of the application-specific background and the potential for application-tailored methods.

7 Application

In the previous chapters the newly developed methodology (Chapter 3 and 4) and the corresponding implementations (Chapter 5) of the methods have been presented. This chapter introduces several examples, where the developed methods are applied to real-world 3D+t tracking datasets in the field of developmental biology. State-of-the-art light-sheet microscopy imaging techniques offer the possibility to record fluorescently labeled nuclei and membranes of whole embryos in 3D over the time interval of many hours [86, 101, 297, 353] offering unprecedented possibilities to study biological phenomena on a detailed single-cell level [354]. A complete analysis framework for 3D+t biological tracking data on the basis of the methods developed in this thesis (Chapter 3 and 4) is illustrated in Figure 7.1 allowing in-depth analysis of cell tracking data. Here, the raw images are segmented and registered before tracking algorithms are applied [84]. The newly developed interactive knowledge discovery framework is then applied to analyze the tracking data. Starting at the whole-embryo level, the biologists can analyze the embryo by interactively focusing on regions of potential interest and develop or test trajectory-based hypotheses [348]. Furthermore, retrospective cell fate analyses can be performed. As a result a knowledge database is generated containing the gathered analysis result of the database, e.g. the extracted cell groups according chosen feature characteristics. On the basis of the extracted knowledge database, the whole developed analysis pipelines can be transferred to new biological datasets of the same specimen to extract the same relevant information in further embryos. Throughout the presented examples the observed challenges are the handling of the error-prone tracking results in embryos with a high cell density and the handling of tens of thousands of cells in these large-scale tracking databases produced by light-sheet microscopy imaging (Section 7.1 and 7.2).

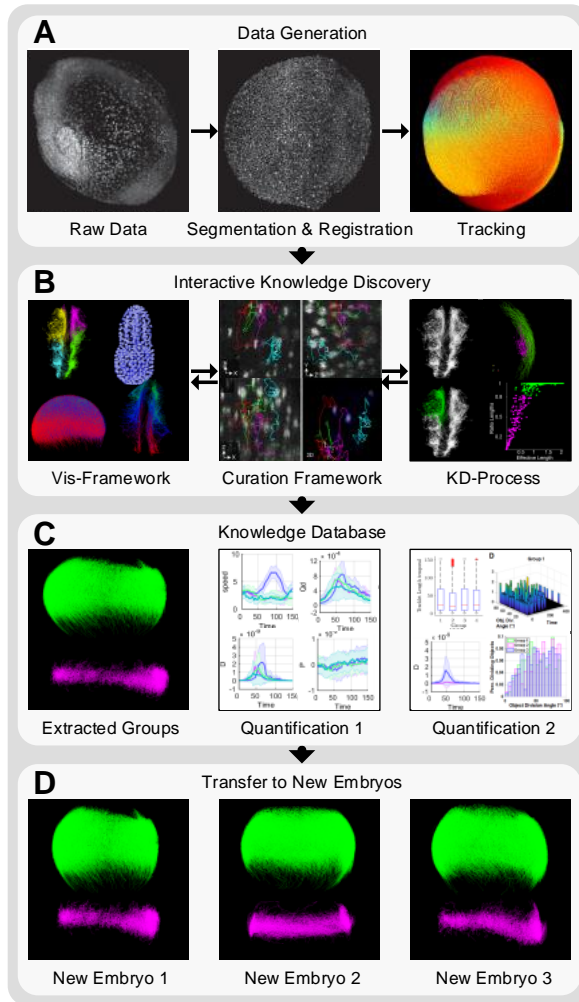


Figure 7.1: Analysis framework to analyze large-scale biological 3D+t tracking data. (A) The raw images are segmented, registered and subsequently tracked to generate trajectories for each single cell. (B) The interactive knowledge discovery framework (KD-framework) allows interactive handling and visualization of the data. The curation framework is used to correct erroneous tracks and the KD-process is applied to extract biologically relevant groups. (C) The KD-process leads to a knowledge database containing the extracted groups and the quantification of these groups by given characteristics. (D) The KD-process can be transferred to new embryos to extract the same biologically relevant groups in each embryo.

7.1 Application of the KD-Process to Separate Hypoblast and Epiblast Cell Groups in Zebrafish Embryos

In this section, tracking datasets on the basis of state-of-the-art light-sheet microscopy images of whole zebrafish embryos were used to extract groups according trajectory-based features descriptions. The interactive selection strategies (Section 4.2) in combination with suitable visualization strategies (Section 4.1) for the embryo tracking data, as described in this thesis, were applied to directly extract groups on the basis of prevalent prior knowledge (Section 4.2.1). The knowledge discovery process was applied on the large-scale embryo tracking datasets to demonstrate the successful use of multi-view selection strategies (Section 4.2.2), trajectory features-based clustering (Section 3.3.5) and interactive transfer of analysis pipelines (Section 3.3.6) on a biological problem, underlining the potential of quantitative assumptions on a single cell level in tracking approaches in developmental biology.

7.1.1 Dataset Description

The image dataset for the knowledge discovery in early embryonic development in zebrafish that is depicted in this section were acquired by using a custom-made digitally scanned laser light-sheet fluorescence microscope (DSLIM). For details regarding the image acquisition, data preparation and data storage see [108]. The complete preprocessing steps starting with the detection, segmentation and tracking of the fluorescently labeled cell nuclei is shown in Figure 7.2. The wild-type zebrafish (*Danio rerio*) embryos were imaged using fluorescent marker expressed in their cellular nuclei *Tg(h2afva:h2afva-GFP)kca66Tg* to analyze cellular dynamics of single cells and group of cells during the early gastrulation phase. The image resolution was 2560x2160x500 voxels pixels with a 60 seconds time period to capture one 3D stack [108]. In each of the recorded 3D images, the fluorescently labeled nuclei were detected separately using a Laplacian-of-Gaussian Scale Space Maximum Projection (LoGSSMP) approach [96]. For a detailed parametrization of the image filters and a detailed description of the complete image analysis pipeline see [348, 355]. The views of the datasets were manually registered using a custom-made graphical user

interface. Here, only the translation had to be estimated because the rotation was known to be 180° . The orientation of the embryos therefore was performed such that the animal-vegetal axis was extended from the positive y-axis to the negative y-axis [348]. Furthermore, the complete embryo datasets were temporally synchronized at the 256-cell stage. Afterwards, the single nuclei were tracked to get the movement paths of the detected objects over the complete time interval [84]. Here, the tracking was directly applied on the registered, filtered and fused segmentation result as described in [96].

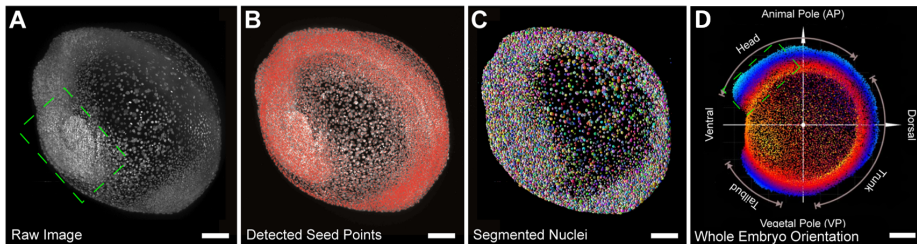


Figure 7.2: Preprocessing steps to extract tracking data for hypoblast and epiblast cell groups in light-sheet microscopy images of zebrafish embryos. (A) Maximum intensity projection overlay for a single time point in early zebrafish somitogenesis stages. (B) Superimposed centroids of detected cell nuclei with the maximum intensity projection overlay. (C) 3D volume rendering of the segmentation result for all cells displayed in a random color code. (D) Oriented whole-embryo data set with the animal-vegetal axis aligned with the x-axis (animal pole positioned on the positive y-axis) and the dorsoventral axis aligned with the x-axis (dorsal part of the positive x-axis [108]). Scale bar: $100 \mu\text{m}$. Figure from [348].

For the tracking approach, a nearest neighbor method was used, searching for the spatially nearest neighbor candidates. The gathered tracking results were saved as a SciXMiner project (Section 5.1) for further analysis.

7.1.2 Interactive Trajectory Knowledge Discovery Framework

The interactive trajectory knowledge discovery framework developed in this thesis (Chapter 3 and 4) was applied to extract and analyze epiblast (non-involuting) and hypoblast (involuting) cells during zebrafish gastrulation in four wild-type zebrafish embryos (a detailed description of the whole process

can be found in the papers [346, 348] that summarize the methodology developed in this thesis). The underlying dataset containing the tracking data of the embryonic cells corresponds to Problem class 20, as described in Section 2.1, indicating a highly complex level of difficulty in the analysis containing fragmented tracks in highly dense region of cells (for more details see Table 2.1). The conceptual approaches on the methodological and interactive side used for this task are listed in Figure 7.3. Here, a subset of the overall approaches (Figure 3.3) is used for the analysis of the described problem. In the following, the different approaches are explained in detail. To extract the epiblast and hypoblast

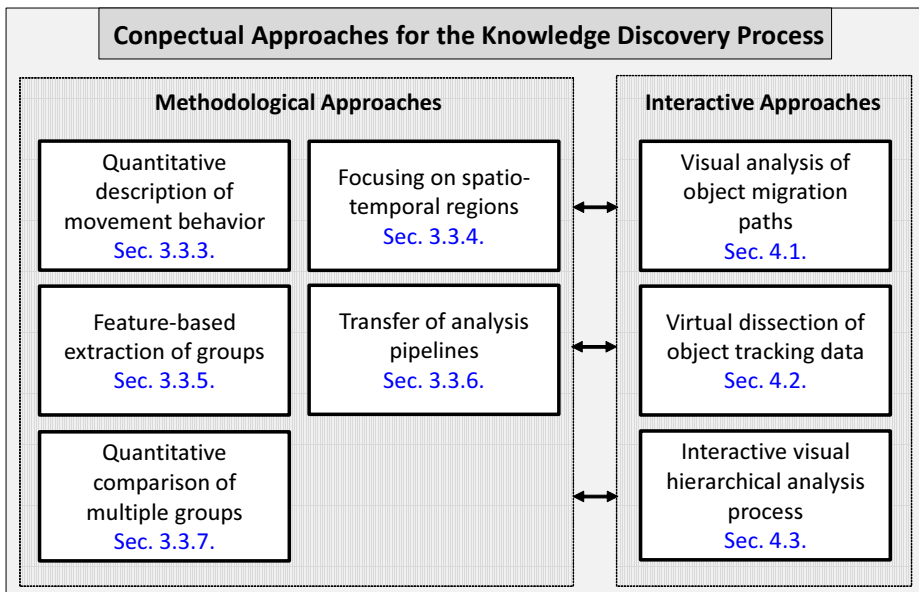


Figure 7.3: Conceptual approaches that are required to extract and analyze epiblast and hypoblast cells during zebrafish gastrulation in four wild-type embryos. Here, a subset of the overall conceptual approaches to analyze tracking data as described in Section 3.3 is used, pointing out the modularity of the overall framework.

cell groups, the first major gastrulation event is analyzed where a involution of cells at the germ ring margin occurs at about 5.5 hours post fertilization (hpf) [356]. For the analysis, the listed methodological and interactive approaches shown in Figure 7.3 are used out of the overall possibilities described in Section 3.3. The different analysis steps performed on the zebrafish embryos to extract the both groups are shown in Figure 7.5. On the basis of the whole em-

bryo dataset (Figure 7.5A) a time range of 5 - 7.25 hpf was cropped to cover the whole gastrulation period (Figure 7.5B). To further focus specifically on the region of interest where the involution happens, a spatial filtering was applied to the embryos (Figure 7.5C) to extract the region containing the germ ring margin with adequate coverage of the surrounding tissue towards the vegetal and animal pole [348]. The tracks of hypoblast cells that perform the involution movement at the germ ring margin form a U-shaped turn-around characteristic (Figure 7.4) where the effective displacement has a much lower value compared to the integrated travel path during the gastrulation phase (Figure 7.5E). On the basis of this criterion, two clusters of the tracks were identified that

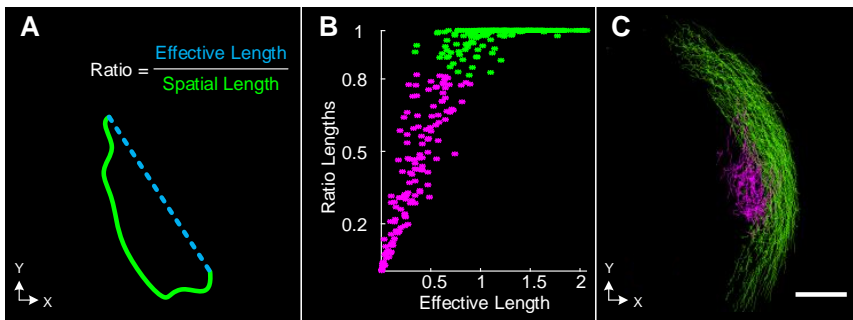


Figure 7.4: Cell movement characterization using trajectory features in early embryonic development. (A) Schematic illustration of the ratio of effective length (distance between start and end point) versus the spatial length (integrated path length). (B) The ratio illustrated in (A) is used to automatically identify two clusters of cell groups corresponding to the epiblast (color-coded in green) and the hypoblast cells (color-coded in magenta) visualized as a scatter plot. (C) 3D trajectory rendering of the extracted epiblast and hypoblast cells. The figure is modified from [348]. Scale bar: 100 μm .

were longer than 70% of the overall time interval. The remaining shorter track fragments were assigned to one of these two classes on the basis of the most common class membership within the labeled tracks in their spatio-temporal proximity (Figure 7.5B-F). The complete analysis pipeline was developed on one single embryo (Embryo 01 in Figure 7.5) and transferred to three further embryos (Embryo 02-04) to extract the same epiblast and hypoblast groups. For methodological details regarding the transfer of the analysis pipeline see Section 3.3.6. For the quantification of the movement behavior of the separated

hypoblast and epiblast cells, mechanical deformation features were used as described in [357]. Here, the volume change rate (P), the distortion rate (Q_d), the speed and the rotation discriminant (D) were computed. The calculated features are all based on the analysis of the local deformation of the groups of cells with regard to a cell reference in the center of the corresponding region.

7.1.3 Results

The result of extracting the hypoblast and epiblast cell groups within four wild-type zebrafish is shown in Figure 7.5. All results of this section were published by Schott et al. in the paper [348]. Here, the extracted hypoblast cell group is shown in Figure 7.5D-E. The color-coded visualization of both cell groups, the epiblast and hypoblast cells, is illustrated in Figure 7.5F qualitatively. A region with an increased cell density at the future dorsal side of the embryo can be observed in the shield region within the hypoblast cells as marked in Figure 7.5D. The visual comparison of the hypoblast groups in all four wild-type embryos indicates that the groups are highly similar between all considered embryos.

The quantification of the movement behavior for the separated epiblast and hypoblast cells, using the mechanical deformation features described in [357], is illustrated in Figure 7.6 and 7.7. For each embryo (Embryo 01-04), the quantification of the epiblast and hypoblast groups is depicted in Figure 7.6, whereas Figure 7.7 compares the separated groups between the considered embryos. The volume change rate (P) with values larger/smaller than zero indicate expansion/compression of the tissue. Furthermore, the speed of single cells is represented by the magnitude of the velocity vector. The distortion rate (Q_d) indicates that objects located within a small neighborhood perform a change of their relative position while the cells within the neighborhood remains constant in motion. Moreover, the rotation discriminant (D) with values larger than zero indicating a rotation movement around a reference center. All these characteristics are used to analyze similarities and differences within the two cell groups. It was observed, that the hypoblast cell group moves slightly faster at 5 hpf compared to the epiblast cells. This velocity difference changes as soon as the hypoblast cells start to involute whereby cells located at the interior side of the germ ring formation perform a directionality change and afterwards move upwards to the animal pole of the embryo (Figure 7.6). Additionally, the rotation

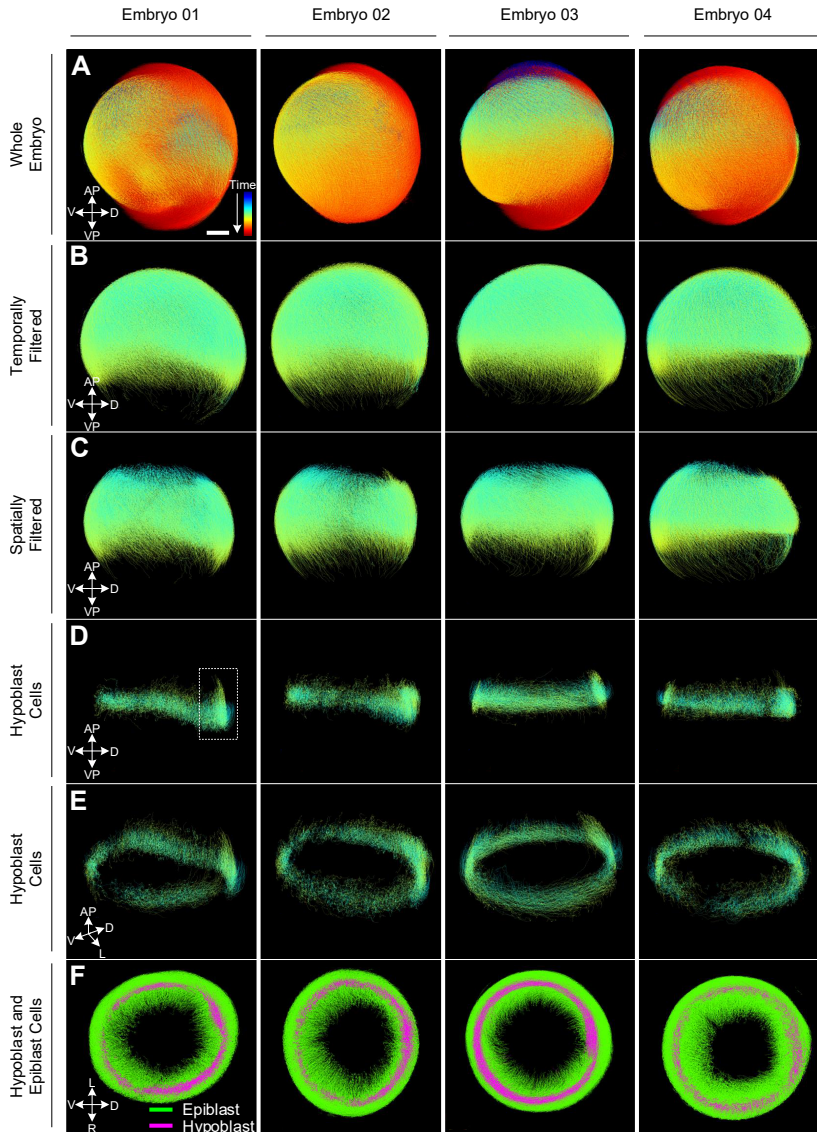


Figure 7.5: Steps that were performed for extracting hypoblast cells in four different wild-type zebrafish embryos (A). The knowledge discovery process was designed interactively on Embryo 01. First, the embryo was filtered temporally and spatially to focus on the region of interest (B, C). Using the KD process, the two groups of cells were separated using a feature-based clustering approach (D-F). The whole analysis pipeline was then applied to Embryos 02-04 resulting in the extraction of the same internalizing cells in all embryos. The color code in panels (A-E) indicates time from 2-14 hpf and the group association to hypoblast (magenta) or epiblast (green) in panel (F). The figure and the caption are taken from [348].

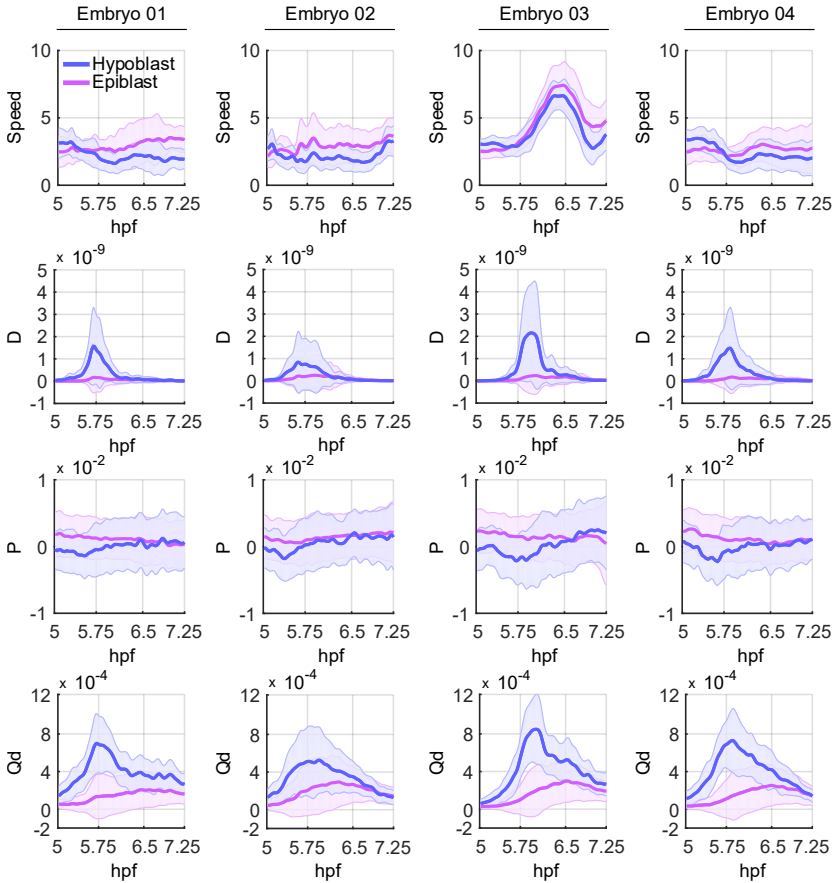


Figure 7.6: Quantitative comparison of selected tissue deformation features recently published in [357] measured for hypoblast (blue) and epiblast (magenta) cells. Each column contains the results obtained on one of four wild-type zebrafish embryos in a time interval spanning early gastrulation from 5 – 7.25 hpf. The selected features comprise speed (magnitude of the velocity vector), the rotation discriminant D (values larger than zero indicate rotation around a reference cell), the volume change rate denoted by P (values larger/smaller than zero indicate expansion/compression of the tissue, respectively) and the distortion rate Q_d (indicates that centroids around a reference point change without affecting the surrounding volume). Note that Embryo 03 physically moved during image acquisition, which caused the increased total speed. Despite this global speed difference, all other quantitative features were nicely captured and revealed comparable patterns among all analyzed embryos. The figure and the caption are taken from [348].

discriminant D indicates peaks between 5.5 hpf and 5.9 hpf. These peaks indicate an increased rotational movement at the margin of the germ ring. At the same time, the rotation discriminant and the speed of the epiblast cells remain more or less on a constant rate during the considered time interval. Furthermore, it can be observed that the volume change rate P is larger than zero for the epiblast cells pointing out an extension movement of the epiblast cell group (Figure 7.7). At the same time the volume change rate for the hypoblast cell group remains nearly around zero with a short phase of negative values just before the peaks of the rotation discriminant. Based on this behavior, the hypoblast cell group seems to have a short period of temporal compression located at the germ ring margin right before the involution process starts. Starting at the time interval around 7.25 hpf, the volume change rates P converge on each other while still remaining positive, suggesting that similar extension movements occur in both, the hypoblast and epiblast, cell groups. Increasing Q_d values of the hypoblast cell group with a peak at 5.75 hpf indicate that most of the cells change their corresponding neighbors without changing the surrounding volume. This behavior indicates an increased number of cell intercalations that occur within the hypoblast group. In general, the result obtained for all four embryos is summarized in Figure 7.7. Here, it can be observed that the qualitative behavior is highly preserved as shown by the largely overlapping feature curves. Moreover, the quantitative analyses are only reasonable, if the region of interest is chosen correctly or the cells has been chosen appropriately (last two columns in Figure 7.7). Otherwise, superimposing effects of multiple objects affects the graphs, leading to no characteristic feature curves.

7.1.4 Discussion

In this section, an exemplary trajectory knowledge discovery pipeline was presented to analyze the resulting tracking data from large-scale light-sheet microscopy image datasets for four different wild-type zebrafish embryos. The proposed scenario allowed the biologist to quantitatively and qualitatively investigate cell tracking datasets for whole embryos at a single cell level and use existing prior knowledge to interactively extract groups of interest. Here, the extracted cell trajectories were used to extract two groups of cells, namely the hypoblast and epiblast cells. The KD-process described in Chapter 3 was further

used to quantitatively compare the extracted groups according a wide range of mechanical deformation features. The quantitative comparison of groups of cells on such a single cell level allows to support hypotheses of biologist and find detailed similarities and differences within groups of cells automatically. Furthermore, investigating the four embryos, a slight physically movement of Embryo 03 was noted during the microscopy imaging process leading to global effect on the speed features as illustrated in Figure 7.6 and 7.7. These panels were intentionally left for demonstration purposes, because these subtle movement characteristics could not be identified with pure visual inspection of the maximum intensity projections of the embryo. An automatic alignment of the embryo before the initial acquisition process and the subsequent features extraction would be easily able to get rid of such global drifting effects. Nonetheless, the global drifting movement only affected the speed feature. All other features still remained in accordance to the other three embryos. The visual inspection of the result using the knowledge discovery framework still remain indispensable and has to be done very carefully to avoid artifacts and increase the validity of the results.

The major challenge in analyzing such tracking data for whole embryo datasets is still the occurring fragmentation of the tracks due to segmentation errors and further artifacts described in Section 2.2. In this application the allocation process was used to handle the fragmented tracking data. For a perfect lineage reconstruction that is required to make valid assumptions about the division characteristics of cells, such as division angles or the synchronization of division waves over the whole specimen, non-fragmented tracks are needed. Therefore, a curation at the beginning of the analysis is required, leading to tracks over the complete considered time interval.

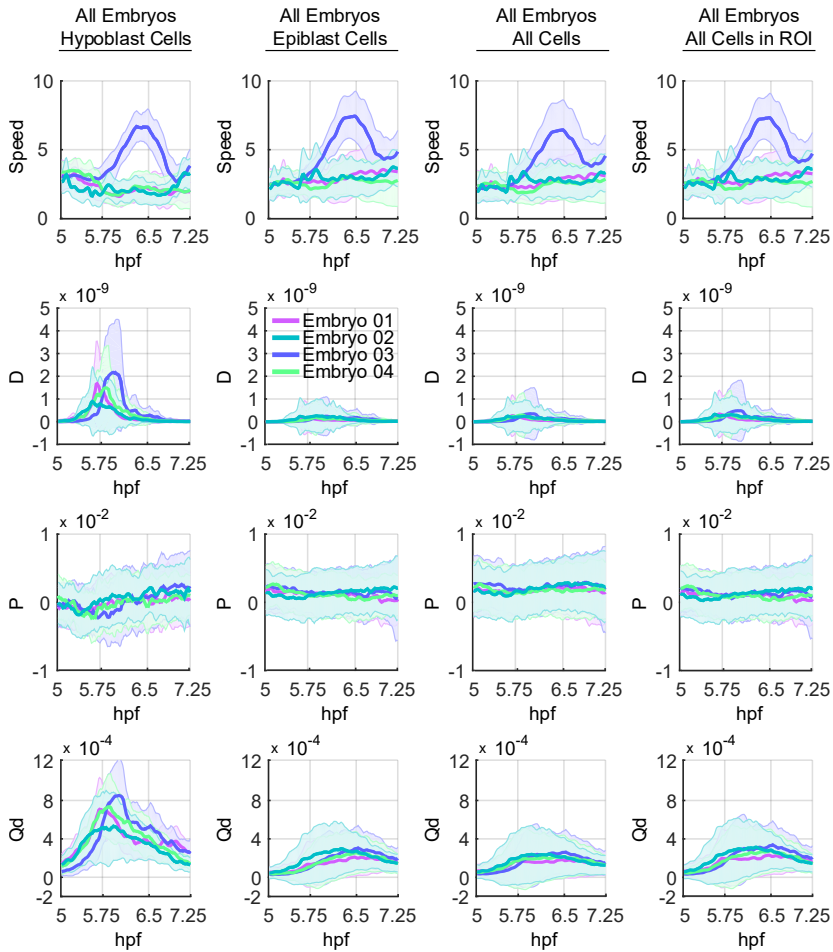


Figure 7.7: Quantitative comparison of selected tissue deformation by [357] measured for all hypoblast cells of all embryos (first column), all epiblast cells (second column), all cells present in selected time window of 5 – 7.25 hpf (third column) and the spatially cropped data selection (fourth column). The extracted features are identical to the ones depicted in Figure 7.6 but all embryos are shown in a single plot for easier comparison. Except for the speed feature of Embryo 3 that was affected by the movement during image acquisition, all other quantitative features were nicely captured and revealed comparable patterns among all analyzed embryos and groups. The group selections in the last two columns show that properly selected regions are crucial to avoid the superposition of different movement behaviors. The figure and the caption are taken from [348].

7.2 Application of the KD-Process to Analyze Neural Crest Cells in Zebrafish Development

The aim of this research project was to extract, visualize and quantify different types of cells groups in the cranial neural crest cell population in zebrafish development. Therefore, light-sheet microscopy imaging with a subsequent tracking approach was used to image and track the cell movement of the neural crest cells in 3D. Existing prior knowledge (Section 3.2) of biologists was used to select two cells groups at predefined time points at a fixed developmental stage (Section 4.1.2). Due to error-prone tracking data, the initial selection of cells has to be curated leading to tracks over the complete time interval. For the curation and selection process, the methods derived in the last chapters were applied to demonstrate the successful interactive selection (Section 4.2), curation (Section 4.4) and multi-view visual analysis (Section 4.1) within a current biological dataset containing fluorescently labeled cell tracking data.

7.2.1 Dataset Description

Zebrafish (*Danio rerio*) embryos with a neural crest reporter line *Tg(-7.2sox10:h2afaa-EosFP)* were used for the analysis described in this section. Further details of the reporter line and the embryo maintenance can be found in [358]. A custom-made light-sheet microscope (DSLIM) that allows multi-channel image recording was used to capture the cellular dynamics during the post-gastrulation stages (11-28 hpf) of neural crest cells [108]. For the image acquisition process using the DSLIM, the embryos were mounted in fluorinated ethylene propylene tubes filled with low melting point agarose (50 mM NaCl, 0.17 mM KCl, 0.33 mM CaCl₂, 0.33 mM MgSO₄, 0.7 mM HEPES pH7.0). The image resolution was 2595 x 2188 pixels with a 20 seconds time period to capture one 3D stack [108]. In each of the acquired 3D images the fluorescently labeled nuclei were detected and subsequently tracked to preserve the movement trajectories. For the tracking, a nearest neighbor method was used to search for the spatially nearest neighbor candidates. Furthermore, the neural crest tracking dataset was oriented in the way that the y-axis and the anteroposterior axis were aligned such that the posterior and anterior sides are located on the positive and negative parts of the y-axis [348]. The left-right axis within the dataset

was extended from the negative to the positive x-axis (Figure 7.8). The oriented dataset was saved as a SciXMiner project and used for subsequent analysis and interactive knowledge discovery as described in Section 7.2.2.

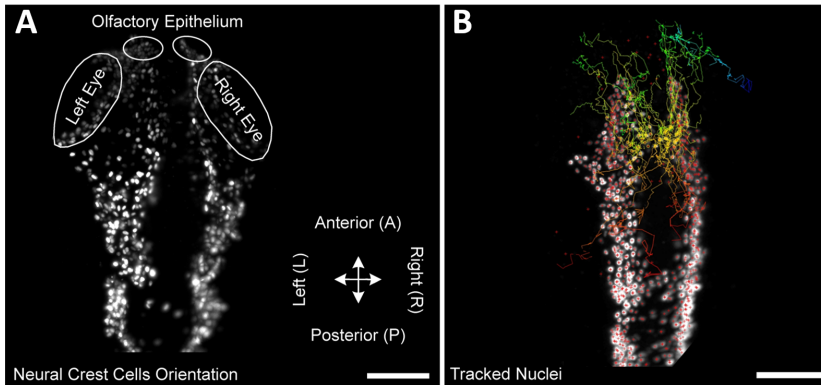


Figure 7.8: (A) Head region of the zebrafish embryo as highlighted in Figure 7.2D with neural crest cells. The circles indicate the spatial positions of the olfactory epithelium and the eyes. (B) Tracking result of 25 arbitrary selected cell nuclei. Scale bar: 100 μm . Figure from [348].

7.2.2 Interactive Trajectory Knowledge Discovery Framework

The interactive trajectory knowledge discovery framework, as described in the previous Chapters 3 and 4, was applied to extract two groups of cells (1°NC and 2°NC) within the cranial neural crest (NC) cells that contribute to the development of the vertebrate eye. A detailed background and the result of the KD-framework described in this section can be found in [358]. The underlying dataset containing the tracking data of the NC cells corresponds to Problem class 18, as described in Section 2.1, indicating a highly complex level of difficulty in the analysis (for more details see Table 2.1). Out of the overall conceptual approaches contained in the KD-framework, Figure 7.9 lists the interactive and methodological approaches that were used for the application in this section. In detail, the biological expert started on the basis of the initial tracking result containing error-prone fragments of trajectories. Here, several time points within the XY, XZ, YZ maximum projections overlay (Section 4.1.2)

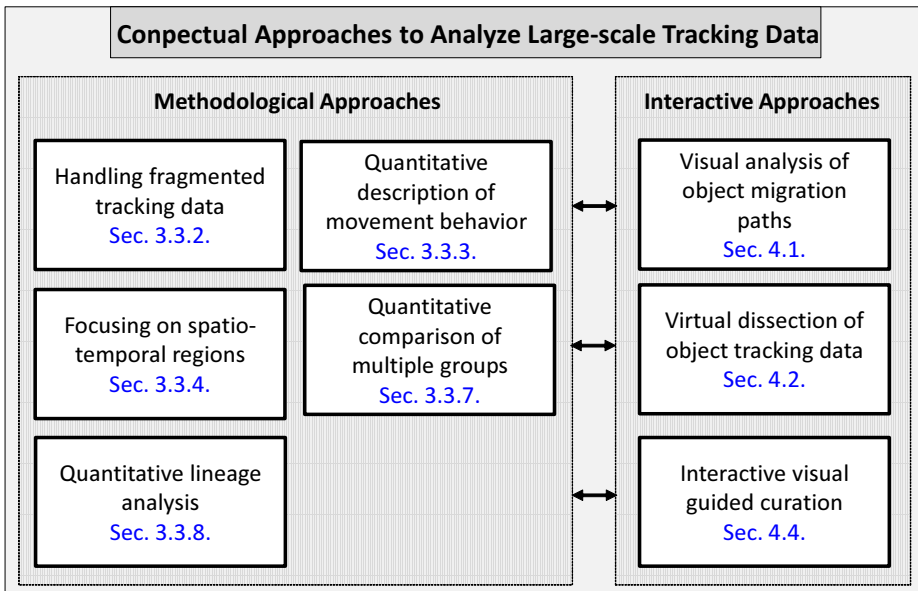


Figure 7.9: Conceptual approaches to extract and analyze cranial neural crest cells in zebrafish eye development. Here, a subset of the overall conceptual approaches to analyze tracking data as described in Section 3.3 is used, pointing out the modularity of the overall framework.

were used to interactively select the two cell groups (1°NC and 2°NC) manually by visual inspection. Using the maximum projection overlays the biological expert was able to use his already gained prior knowledge to extract the two groups adequately. Hereby, the interactive virtual dissection possibilities and the instant propagation of selection results to multiple views as described in Section 4.2 were used to refine the selection result in all three maximum projection overlays and exclude single cells from the selection that do not belong to the groups of interest. Based on this initial selection of the 1°NC and 2°NC cell groups, the manual curation scheme as described in Section 4.4 was used to interactively fuse the trajectory fragments and create cell tracks over the complete time interval. The quality measure was applied for a given number of previous time points allowing to handle temporary misdetections and noise artifacts and therefore skip time points (Section 3.3.2). Arising linking conflicts, such as multiple linkings to one predecessor, are handled automatically. The resulting complete tracks were superimposed on the original maximum projection over-

lays to visually investigate the global movement behavior of each single cell in detail and make qualitative assumptions about observed characteristics.

7.2.3 Results

Analyzing Neural Crest Cells in Developing Zebrafish Embryos

In the manual curation process (Section 4.4) that is applied to the neural crest cell tracking data the correct chosen linking candidate is in one third of the curated tracks among the first three proposals (Figure 7.10A, B). Even, in the cases where the valid linking candidate was not found among the proposed tracks, it took less than 20 seconds on average to identify the correct linking candidate. Therefore, visual inspection methods combined with manual selection strategies of the correct trajectory fragment were used (Figure 7.10C, D). The initially selected groups of 1°NC and 2°NC cells containing highly fragmented trajectories is shown in Figure 7.10E1. The result of a manual curation after 6 hours of curation is depicted in Figure 7.10E2 and the final result after 11 hours is illustrated in Figure 7.10E3. The final corrected trajectories of both cell groups superimposed with the maximum intensity projection overlay in the XY-plane is illustrated in a backward (Figure 7.12B1), forward (Figure 7.12B2) and backward-forward direction (Figure 7.12B3). The trajectories of only the 1°NC cell group are illustrated superimposed with the corresponding maximum intensity projection in the XY-(Figure 7.12C1), XZ-(Figure 7.12C2) and ZY-plane (Figure 7.12C3). Moreover, the corresponding curated tracks for the 2°NC cell group in the XY-(Figure 7.12D1), XZ-(Figure 7.12D2) and ZY-plane (Figure 7.12D3) are shown.

Analyzing Precursor Cells of the Olfactory Epithelium in Zebrafish Embryos

A biological expert selected a set of approximately 100 precursor cells of the olfactory epithelium at 23 hpf within the cranial neural crest dataset. The initial selection result is illustrated in Figure 7.11A (blue selection). The curated tracks, that were fused as described in Section 4.4 are illustrated in Figure 7.11B. Here, the blue colored tracks indicate the curation result, whereas the green

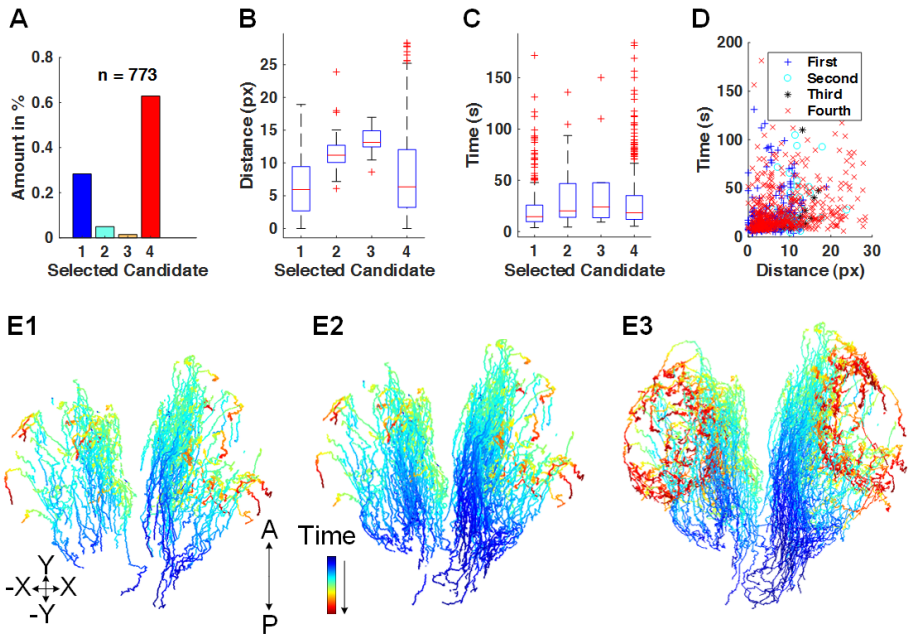


Figure 7.10: Quantitative validation of the manual curation module. (A) Distribution of the selected correct link candidate, where Candidates 1-3 are the automatic suggestions computed by the framework and Candidate 4 indicates that the user selected an alternative link candidate to be correct. (B) Box plots of the distance in pixel of the selected link candidate. (C) Box plot of the time in seconds that was required by the manual annotator to perform the decision. For box plots in (B) and (C), the median is indicated by red bar and the 25% and 75% quantiles are indicated by the lower and upper extents of the box, respectively. (D) Scatter plot of time in seconds versus the distance of the linked candidate. Color and symbols indicate, which candidate has been selected. (E) Selected trajectories of the neural crest dataset before (E1), after 6 hours of curation (E2) and after 11 hours at the end of manual curation (E3). Time of cell movement is indicated by the color-code from blue to red. This figure is extended from [348].

tracks represent the not-curved track fragments. The work of [359] showed that there is a major contribution of the neural crest cells to the microvillous sensory neurons located in the olfactory epithelium. In a current study the contribution of the neural crest cells was presented by photoconversion-based fate mapping in combination with time lapse microscopy imaging. However, the migration path of the precursor cells as well as their origin remained still not analyzed. On the basis of the curation result, as illustrated in Figure 7.11C,

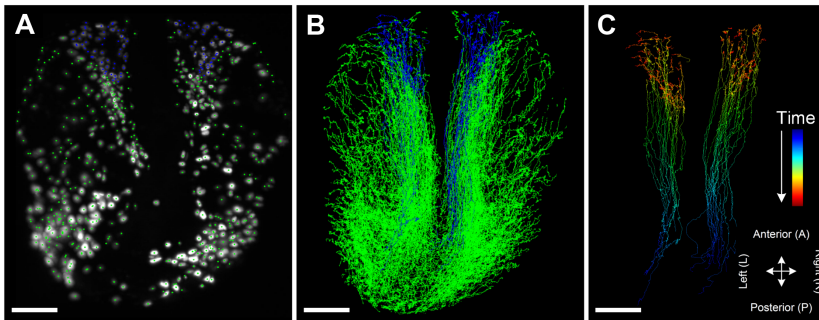


Figure 7.11: Interactive tracking correction of precursor cells of the olfactory epithelium in zebrafish embryo. (A) To analyze the precursors of the olfactory epithelium, a particular subgroup of neural crest cells of a zebrafish embryo, a set of about 100 cells was interactively selected at 23 hpf (blue). The corrected tracks are shown in the global context (B) as well as in isolation using time for coloring (C). Scale bar: 100 μm . The figure and the caption are taken from [348].

the trajectories of the precursor cells of the olfactory epithelium were investigated. Here, it was shown, that the origin of the neural crest cells is located in the olfactory placode. Furthermore, the trajectories revealed a high directive movement toward the anterior direction (Figure 7.11C) that is consistent with the *chase-and-run* movement characteristic of the neural crest cells toward the cells of the olfactory placodal as described in [360].

7.2.4 Discussion

In this section, the knowledge discovery framework was applied to manually curate erroneous tracking data from large-scale light-sheet microscopy image data of neural crest cell populations. The interactive manual curation scheme allowed the biologists to create fulltracks out of the fragmented tracking data within 11 hours of time effort. On the basis of the fulltrack data, the development of the vertebrate eye was studied on a single cell level by the biologists. Furthermore, two underlying groups of neural crest cells were extracted using the virtual dissection possibilities as described in Section 4.2. On the basis of the cell trajectories superimposed with the maximum intensity projection overlay, the spatio-temporal movement behavior of the extracted cells was described, e.g. how the eye primordium becomes established (30 hours post fertilization)

and how the neural crest cells were formed in the neural plate (11-12 hours of post fertilization). Moreover, it was suggested based on the observations that the two neural crest groups have comparable population sizes during the early phase of eye development. The major challenge in the analysis of highly dense and fragmented tracking data is the investigation of a high manual time effort to curate the fragmented trajectories and find the correct predecessor trajectories. Subsequent automated trajectory knowledge discovery methods (Chapter 3) are often only meaningful if applied on the curated fulltrack data. To reduce the manual time effort dramatically, the interactive manual curation framework was presented allowing to fuse fragmented cell tracking data with a mean of 20 seconds per linking. However, in the future automated curation methods are required leading to a robust curation of fragmented tracking data allowing to automatically extract and quantify prior defined characteristics of cell groups.

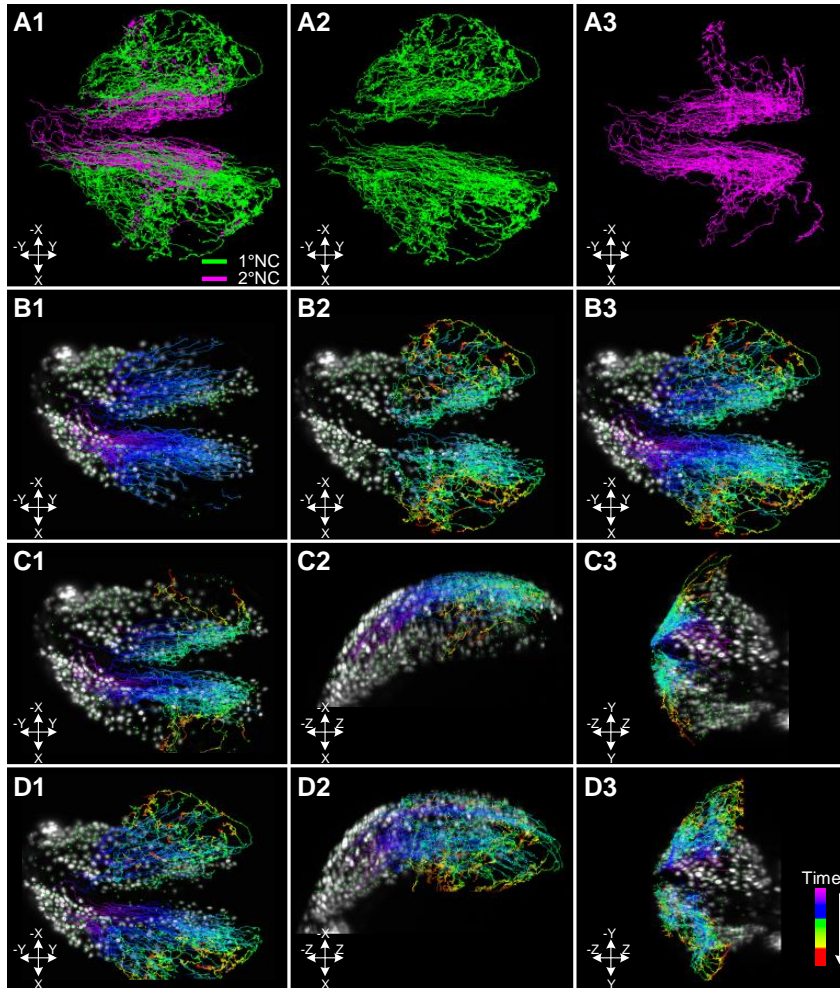


Figure 7.12: Curation result for cranial neural crest cells in zebrafish embryo for two selected groups (1°NC and 2°NC). (A) 3D rendering of the curated neural crest cells with both groups displayed (A1), only the 1°NC group (A2) and the 2°NC group (A3). (B) The 3D trajectories superimposed with the maximum intensity projection overlay (MIP) for both neural crest cell groups in a backward (B1), forward (B2) and combined forward and backward direction (B3). (C) Superimposed 3D trajectories with MIP, only for the extracted 2°NC group in the XY (C1), XZ (C2) and YZ-plane (C3). (D) Superimposed 3D trajectories with MIP for the extracted 1°NC cell group in the XY (D1), XZ (D2) and YZ-plane (D3).

8 Conclusion

As tracking technologies are rapidly evolving and at the same time data storage techniques are enhancing, trajectory data mining has become an essential tool to handle the tremendous amount of generated trajectory data. Due to the enormous size and complexity of the datasets, the extraction of relevant knowledge from the data is a challenging problem. Furthermore, existing prior knowledge can often not be incorporated adequately due to the complex nature of the trajectory databases. A central intention of the work presented in this thesis was to develop new methods and concepts to allow an efficient analysis of huge amounts of trajectory data by using automatic methods. For instance, clustering methods are used in combination with efficient possibilities for incorporating existing prior knowledge in an interactive and easy way. Moreover, new approaches for visual explorative data analysis and trajectory mining were developed.

In Chapter 2, a synthetic benchmark was presented to validate the methods for analyzing trajectory data in a systematic way and to proof the robustness against artifacts such as fragmentation and noise. A new generic concept combining interactive visual explorative data analysis and fully automated trajectory pattern mining approaches allows to efficiently incorporate existing prior knowledge (Chapter 3). To demonstrate how to handle different problem classes of trajectory datasets within the analysis pipelines, Chapter 3 handles several topics leading from the handling of fragmented tracks up to the efficient transfer of complete analysis pipelines to new databases. Furthermore, the efficient incorporation of prior knowledge using the developed visual explorative framework is described in Chapter 4. All concepts and methods were implemented in an open-source software tool (Chapter 5) and were successfully applied to huge trajectory databases in the field of developmental biology (Chapter 7). A user-study is presented in Chapter 6 to evaluate the methodological and interactive approaches described in Chapter 3 and 4. Therefore, the major contributions presented in this thesis are:

1. A new concept to systematically incorporate prior knowledge in the analysis of huge 2D+t and 3D+t tracking databases combining interactive visual explorative and fully automated trajectory analyzing methods.
2. A new generalized framework for the integration of prior knowledge in the extraction and quantification of relevant groups of moving objects in huge 2D+t and 3D+t datasets.
3. An efficient approach to transfer complete analysis pipelines generated by the knowledge discovery process to new datasets. This allows to efficiently extract the same relevant information in multiple trajectory datasets and quantify differences between these datasets.
4. A new approach to handle fragmented tracks within the analysis of 3D+t datasets based on neighborhood-based assignment of fragmented tracks. This allows to include fragmented tracks within the analysis and find transitional regions between existing groups of tracks. The algorithms allocate the fragmented tracks based on the existing group memberships in the neighborhood.
5. An improved curation framework for erroneous and fragmented tracks based on proposed track candidates and track quality measures. Here, the framework allows an efficiently guided manual curation of highly dense and complex 3D+t datasets of moving objects.
6. A new visualization framework for the interactive handling of huge 3D+t tracking datasets. A customized set of visual representations allows to efficiently integrate prior knowledge simultaneously in several levels of detail.
7. The consistently validation of the functionality of the proposed methods on specific validation benchmarks that were newly developed. The robustness of all methods was analyzed using variable levels of track fragmentation and noise.
8. Extension of the open-source MATLAB toolbox SciXMiner to be applicable for processing tracking databases, extracting track-based features, interactive visualizing tracking data, integrating prior knowledge, curating tracking errors extract and quantify relevant groups.

-
9. A new study to evaluate the usability and applicability of the developed methods. Therefore, five use-cases were developed comprising different aspects within trajectory knowledge discovery pipelines. The result of the study indicates that even complex analysis tasks can be solved by inexperienced user in less than 5 minutes.
 10. A new analysis framework for 2D+t and 3D+t light-sheet microscopy image datasets, which comprises the visualization, interactive integration of prior knowledge in the analysis and the statistical quantification of the results. In close collaboration with biologists and physicists, tracking data of 3D+t light sheet microscopy images of zebrafish embryos were successfully analyzed and interpreted. In detail, cranial neural crest cells ending in the eye region were interactively selected and two groups of relevant neural crest cells were extracted. Furthermore, the cell groups of interest were analyzed using features like global cell movement, cell counts, directionality changes and lineage analysis.
 11. The successful application of the newly developed and implemented EmbryoMiner extension for SciXMiner to semi-automatically analyze 3D+t trajectory databases of zebrafish embryos containing 25,000 cells. In detail, hypoblast and epiblast cells were interactively separated at about 5.5 hours post fertilization and subsequently quantified according features such as speed, volume change rate, rotation discriminant and the distortion rate. Furthermore, precursor cells of the olfactory epithelium were reconstructed and the spatio-temporal movement paths were investigated.

In most of the automated methods for the knowledge discovery in trajectory databases that are described in Chapter 3, non-fragmented trajectories were used. In real-world tracking experiments, however, the imaging conditions can differ a lot within a single experiment or between several experiments often leading to segmentation errors and the subsequent lowering of the tracking accuracy. In the future, more robust pipelines including image analysis and segmentation are required to cope with the varying quality of images. Here, deep learning approaches offer a great potential to improve the segmentation quality serving as a basis for the tracking algorithms [361–364]. However, even the best published tracking approaches, which have small error rates, unavoidably

lead to error-prone cell lineages and fragmented trajectory segments requiring an enormous manual correction effort [135]. Therefore, an additional effort is required in the future to improve tracking algorithms that are able to handle events like cell division and missing object detections. In the field of deep learning, several tracking methods are currently developed having the potential to lead to error-free tracks in the future [365–367]. On the basis of such perfect tracks, the knowledge discovery methods for 2D+t and 3D+t developed in this thesis can be applied automatically without requiring the developed curation framework. The registration of different embryo datasets was done manually in this thesis using the interactive methods described in Chapter 4. To facilitate high-throughput experiments of 3D+t datasets in the future, however, automatic methods for the registration are required. Furthermore, there is a drawback in automatically extracting groups of potential interest without the presence of useful prior knowledge. Therefore, new algorithms have to be developed for automatically finding groups of potential interest within trajectory databases. An instant visual feedback of the proposed groups of interest could lead to a fast selection of meaningful groups and the subsequent slight manual refinement. For the interactive choice and refinement of the groups, the methods introduced in Chapter 4 can be used. In this thesis an approach to transfer complete analysis pipelines is introduced. In case of slight inhomogeneities within the datasets, the proposed methods can be still automatically applied. However, in case of databases comprising strong inhomogeneities, manual refinement effort is still required. For future work, it could be enormously helpful to automatically transfer analysis pipelines even for complex and inhomogeneous datasets finding and registering corresponding regions of interest. In case of biological application, this would allow to systematically study effects of chemical components and gene modification in high-throughput 3D+t tracking applications on a detailed single cell level.

A Nomenclature and Symbols

In this section, the notation used throughout the thesis is summarized. The initial enumeration demonstrates the basic concepts used for nomenclature and the subsequent list of symbols explains all used symbols within their context. The nomenclature is partly based on the formulation given in [197].

- Lower case letters are used for scalars, parameters, indexing variables and functions (*e.g.*, a, b, c, d are common names for parameters, $f(\cdot), g(\cdot), h(\cdot)$ are common names for functions or functionals, i, j, k, l are common names for indices and finally x, y, z are common names for scalars or variables).
- Upper case letters are used for number of entities (*e.g.*, N, N_f for the number of data tuples and the number of features, respectively).
- Upper case bold face letters represent matrices (*e.g.*, \mathbf{M}, \mathbf{N}).
- Lower case bold face letters represent vectors (*e.g.*, \mathbf{x}, \mathbf{y} for column vectors and $\mathbf{x}^\top, \mathbf{y}^\top$ for row vectors).
- Upper case script and Greek letters represent sets (*e.g.*, $\Omega, \Theta, \mathcal{X}, \mathcal{L}$).
- Accessing elements of data structures is denoted by square brackets (*e.g.*, $\mathbf{x}[1]$ accesses the first element of vector \mathbf{x} , $\mathbf{M}[1, 2]$ accesses the matrix element in the first row and second column).
- Elements topped with a hat refer to estimates of a certain quantity (*e.g.*, \hat{x} represents an estimate of the scalar x , $\hat{\mathbf{x}}$ represents an estimate of vector \mathbf{x}).
- Elements topped with a tilde refer to a changed quantity (*e.g.*, $\tilde{\mathcal{X}}$ represents a modified subset of \mathcal{X}).

- Optimal solutions are indicated by an asterisk (e.g., x^* would denote the optimal solution to $\min(f(x))$).

Symbol	Description
BB	Bunch deletion based model
CLR	Cross Likelihood Ratio
DBN	Dynamic Bayesian Network
DTW	Dynamic Time Warping
ED	Existing database
FGM	Fulltrack generation module
FGM_{ED}	Fulltrack generation module based on existing databases
FGM_{SY}	Fulltrack generation module based on purely synthetic databases
FGM_{SM}	Fulltrack generation module based on simulated trajectories on the basis of real databases
GFP	Green Fluorescent Protein
GPS	Global Positioning System
HMM	Hidden Markov Model
KD	Knowledge discovery
L_n	The n-th linking within a trajectory
$L_{sel,n}$	List of linkings that are deleted
MM	Markov Model
MB	Markov chain based model

MS	Multi-step approach
\mathcal{N}	Neighborhood
N_L	Number of linkings within a trajectory
N_{LC}	Number of linking candidates
$N_{L,del}$	Number of linkings within a trajectory that are deleted
N_P	Number of points within one trajectory
$N_{P,del}$	Number of points to delete within a trajectory
N_{STS}	Describing the number of different states within S_{state}
N_T	Number of trajectories within a set of trajectories (\mathcal{T}_{set})
N_{TDB}	Number of trajectory databases
N_{TNB}	Number of nearest neighbors within the trajectory neighborhood \mathcal{T}_{NBH}
ODM	Object division module
OS	One-step approach
Pi	Point within a trajectory described through coordinates x_i, y_i, z_i at the time point t_i
$P_{del,BB}$	Percentage of points within a trajectory that is deleted in the Bunch-based model
P_{del}	Percentage of points within a trajectory that is deleted
$P_{del,MB}$	Percentage of points within a trajectory that is deleted in the Markov-based model
$P_{Del,S=1}$	Probability of deleting a point when the state of the Markov Model is equal one

$P_{Del,S=2}$	Probability of deleting a point when the state of the Markov Model is equal two
$P_{t,del}$	Percentage of trajectories that are fragmented
$Q_{A,Noise}$	Simulated noise level for FGM_{SY} modules
$Q_{Alloc,FM}$	Fuzzy membership of a given neighborhood in the allocation process
$Q_{Alloc,NN}$	Number of nearest neighbors for the allocation process
$Q_{Alloc,NNAT}$	Number of not allocated tracks
$Q_{Alloc,NWAT}$	Number of wrongly allocated tracks
$Q_{Alloc,PMC}$	Percentage of most common group membership
$Q_{Alloc,PRT}$	Percentage of representative tracks
$Q_{Alloc,PWAT}$	Percentage of wrongly allocated tracks
QM	Quality measure
QRM	Quality reduction module
QRM_{BB}	Bunch-based quality reduction module
QRM_{MB}	Markov-based quality reduction module
ROI	Region of interest
S_{state}	Sequence of states of a Markov Model
\mathbf{T}	Transition matrix for a Markov model
\mathcal{T}	The notation for a trajectory
\mathcal{T}_{NBH}	Trajectory neighborhood
\mathcal{T}_{NBH}^{SP}	Spatial trajectory neighborhood
\mathcal{T}_{NBH}^{ST}	Spatio-temporal trajectory neighborhood

\mathcal{T}_{set}	A set of trajectories
$\mathcal{T}_{set,QR}$	A set of trajectories after the application of the QRM
TDB	A set of trajectory databases
x_i	X-position of a point within a trajectory
y_i	Y-position of a point within a trajectory
z_i	Z-position of a point within a trajectory
$\Theta[P_{t,del}, P_{del}]$	Specify the percentage of trajectories that are fragmented $P_{t,del}$ and the percentage within a trajectory that is deleted P_{del}
κ_{DA}	Division tracking error
μ_{BD}	Mean bunch length for the bunch deletion based model (BB)
σ_{BD}	Standard deviation of the lengths for the bunch deletion based model (BB)
$\tau_{Seg,min}$	Minimal segment length for the bunch deletion based module (BB)
ψ_{MODR}	Mean object division rate
ψ_{SODR}	Standard deviation of the object division rate

B Benchmark Databases and Parametrization

B Benchmark Databases and Parametrization

Name	NG	NOPG	Time Points	FGM	Description
TDB.S1	1	9597	300	FGM_{SM}	Simulated whole zebrafish embryo using the methodology described in [110]
TDB.S2	1	829	550	FGM_{SM}	Simulated neural crest cell of zebrafish embryos located in the head region using the methodology described in [110]
TDB.S3	2	100	200	$FGM_{SY,DB1}$	Two groups of objects. One group with a straight movement characteristics and the other group with a turn-around movement.
TDB.S4	4	100	200	$FGM_{SY,DB2}$	Two groups of objects. One group with a straight movement characteristics and the other group with turn-around movements to both sides.
TDB.S5	6	100	200	$FGM_{SY,DB3}$	Six groups of objects. The benchmark as described in TDB.S3 with three different time points of the beginning turn-around movement.
TDB.S6	2	100	200	$FGM_{SY,DB4}$	Two groups of objects. One group with a straight movement pattern and the other group containing two periods of short local undirected movement.
TDB.S7	2	100	200	$FGM_{SY,DB5}$	Two groups of objects. One group with a straight movement pattern and the other group containing four periods of short local undirected movement.
TDB.S8	3	100	200	$FGM_{SY,DB6}$	Three groups of objects. One group is moving straight, one group moves straight with an additional short period of undirected movement and the one group with a turn-around movement.

Table B.1: The trajectory benchmark databases used for the validation in Chapter 3, 4 and 6.1. The columns contain the trajectory dataset classifier (Name), the number of groups (NG), the number of objects per group (NOPG), the number of simulated time points (Time Points), the type of fulltrack generation modules used to produce the dataset (FGM) and the description of the dataset.

Name	NG	NOPG	Time Points	FGM	Description
TDB.S9	6	100	200	$FGM_{SY,DB7}$	Six groups of objects. The same characteristics as described in benchmark TDB.S8 but additionally mirrored.
TDB.S10	8	100	200	$FGM_{SY,DB8}$	In total eight groups of objects. Four groups with given characteristics that are mirrored. One of these four groups has a straight movement characteristic. An other group moves also straight with a short period of undirected movement. One group follows a turn-around movement and the fourth group also follows a turn-around movement with an additional short period of undirected movement.
TDB.S11	8	100	200	$FGM_{SY,DB9}$	Eight groups of objects. The same characteristics as described in benchmark TDB.S10. The only difference is that the period of undirected movement happens at an earlier time point. Furthermore, the turn-around movement is not synchronized in both sides.
TDB.S12	4	100	200	$FGM_{SY,DB10}$	Four groups of objects. All four groups start to move straight. Two of the four groups move sideways and back again and the other two groups continue moving straight with a short period of undirected movement.

Table B.2: The trajectory benchmark databases used for the validation in Chapter 3, 4 and 6.1. The columns contain the trajectory dataset classifier (Name), the number of groups (NG), the number of objects per group (NOPG), the number of simulated time points (Time Points), the type of fulltrack generation modules used to produce the dataset (FGM) and the description of the dataset.

Name	NG	NOPG	Time Points	FGM	Description
TDB ₁	2	100	200	$FGM_{SY,DB1}$	Two groups of objects with the same movement characteristics as in TDB.S2. One group with a straight movement characteristics and the other group with a turn-around movement. Here, all trajectories are fragmented, whereas 20% of all points are deleted using a bunch-based deletion module with a mean bunch length of 5.
TDB ₂	2	100	200	$FGM_{SY,DB1}$	Two groups of objects with the same movement characteristics as in TDB.S2. One group with a straight movement characteristics and the other group with a turn-around movement. Here, all trajectories are fragmented, whereas 30% of all points are deleted using a bunch-based deletion module with a mean bunch length of 2.
TDB ₃	2	100	200	$FGM_{SY,DB1}$	Two groups of objects with the same movement characteristics as in TDB.S2. One group with a straight movement characteristics and the other group with a turn-around movement. Here, 80% of all trajectories are fragmented, whereas 40% of all points are deleted using a bunch-based deletion module with a mean bunch length of 1.

Table B.3: Simulated trajectory benchmark databases. The columns contain the trajectory dataset classifier (Name), the number of groups (NG), the number of objects per group (NOPG), the number of simulated time points (Time Points), the type of fulltrack generation modules used to produce the dataset (FGM) and the description of the dataset.

C Knowledge Discovery Process of Large-Scale 2D+t and 3D+t Trajectory Databases

Overview of Conceptual Approaches

Abbreviation	Sec.	Description
MA _{HFTD}	3.3.2	Handling fragmented tracking data
MA _{QDMB}	3.3.3	Quantitative description of movement behavior
MA _{FSTR}	3.3.4	Focusing on spatio-temporal regions
MA _{FBEG}	3.3.5	Feature-based extraction of groups
MA _{TAP}	3.3.6	Transfer of analysis pipelines
MA _{QCMG}	3.3.7	Quantitative comparison of multiple groups
MA _{QLA}	3.3.8	Quantitative lineage analysis
MA _{QNA}	3.3.9	Quantitative neighborhood assumptions
IA _{VAMP}	4.1	Visual analysis of object migration paths
IA _{VDTD}	4.2	Virtual dissection of object tracking data
IA _{VHAP}	4.3	Interactive visual hierarchical analysis process
IA _{IVGC}	4.4	Interactive visual guided curation

Table C.1: Abbreviations for conceptual approaches.

Handling Fragmented Object Tracking Data

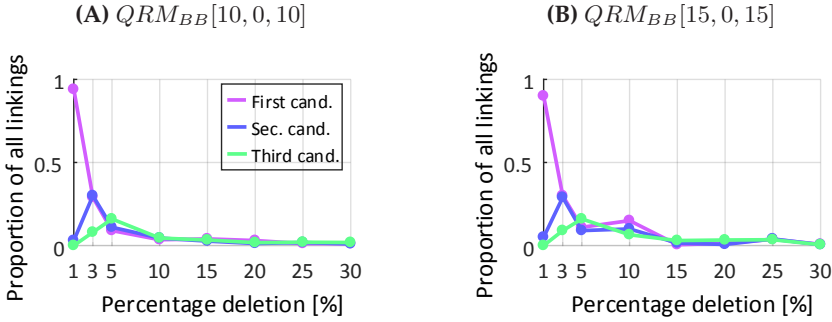


Figure C.1: Impact of fragmentation to the linking quality using the quality measure QM_{EP} . The first (magenta), second (blue) and third (green) candidate that are proposed by the quality measure QM_{EP} are shown. The percentage of overall correct linking decisions is hereby represented by the first candidate. There is no noise added to the trajectory positions. (A) For the fragmentation, a bunch deletion with a mean bunch length of 1 is used leading to short gaps within the trajectories. (B) Here, a mean bunch length of 10 is used resulting in longer gaps.

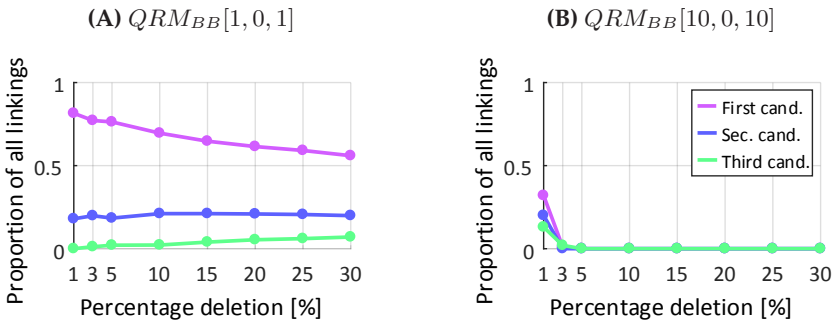


Figure C.2: Impact of fragmentation to the linking quality using the quality measure QM_{AP} . The first (magenta), second (blue) and third (green) candidate that are proposed by the quality measure QM_{AP} are shown. The percentage of overall correct linking decisions is hereby represented by the first candidate. There is no noise added to the trajectory positions. (A) For the fragmentation, a bunch deletion with a mean bunch length of 1 is used leading to short gaps within the trajectories. (B) Here, a mean bunch length of 10 is used resulting in longer gaps.

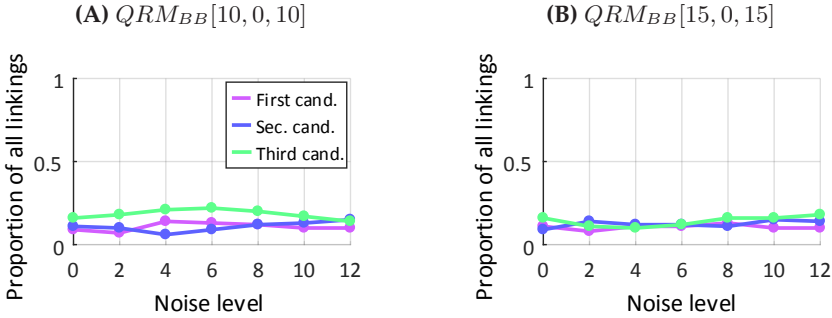


Figure C.3: Impact of noise to the linking quality using the quality measure QM_{EP} . The first (magenta), second (blue) and third (green) candidate that are proposed by the quality measure QM_{EP} are shown. The percentage of overall correct linking decisions is hereby represented by the first candidate. Fragmentation with 5% of deleted points ($\Theta[1, 0.05]$). (A) For the fragmentation, a bunch deletion with a mean bunch length of 10 is used leading to short gaps within the trajectories. (B) Here, a mean bunch length of 15 is used resulting in longer gaps.

$N_{\text{Alloc,NN}}$	$Q_{\text{Alloc,PMC}}^{100}$	$Q_{\text{Alloc,PMC}}^{80}$	$Q_{\text{Alloc,PMC}}^{60}$
5	21/99	28/35	42/0
10	1/261	15/88	34/15
15	0/1466	0/1466	18/46
20	0/1466	0/1466	0/1466

Table C.2: Wrong versus not allocated tracks for the different combinations of the number of nearest neighbors and the percentage of the predominant class. For evaluation benchmark database TDB_S3 with spatially touching groups ($TDB_S3[D_{BG} = 20]$) is used. All trajectories within the benchmark are fragmented using a deletion of 10% ($\Theta[1, 0.1]$). There are no fragmentation artifacts through object division ($\kappa_{DA}[0, ODM[\infty, 0]]$). For fragmentation a bunch deletion module with a mean bunch length of 1 ($QRM_{BB}[1, 0, 1]$) is used, indicating short gaps for the total number of $N_T = 1655$ trajectories.

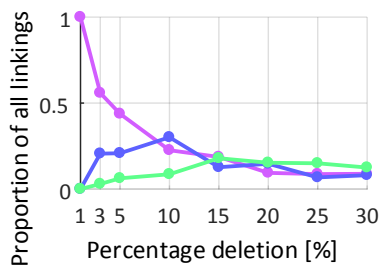


Figure C.4: Impact of fragmentation to the linking quality using the quality measure QM_{EP} . The bunch-based deletion method with the parametrization $QRM_{BB}[5, 2, 5]$ is used. The first (magenta), second (blue) and third (green) candidate that are proposed by the quality measure QM_{EP} are shown. The percentage of overall correct linking decisions is hereby represented by the first candidate. There is no noise added to the trajectory positions. (A) For the fragmentation, a bunch deletion with a mean bunch length of 1 is used leading to short gaps within the trajectories. (B) Here, a mean bunch length of 5 and a standard deviation of the bunch length of 2 is used.

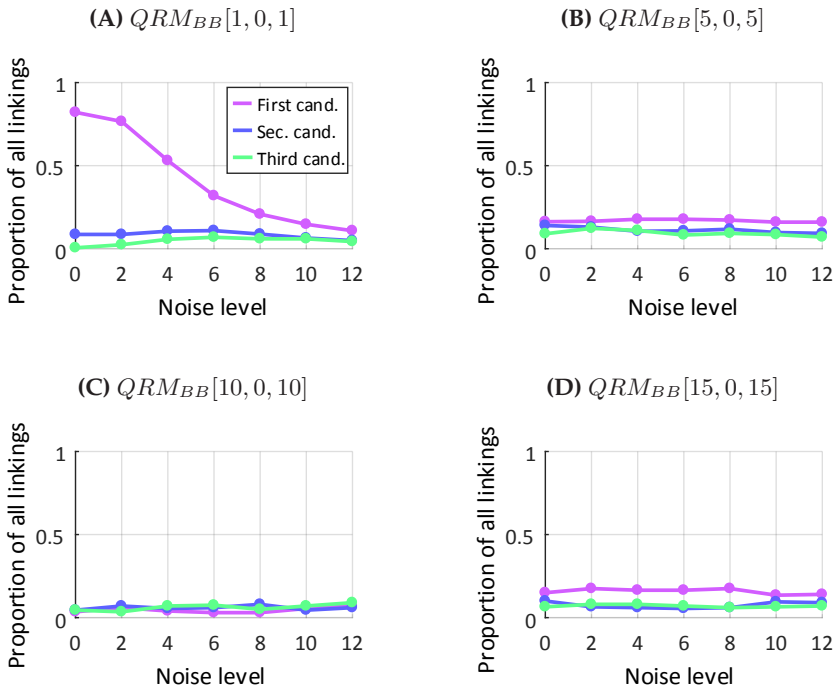
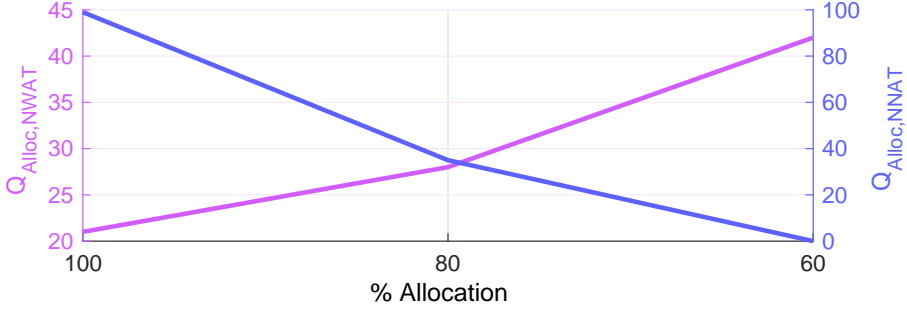
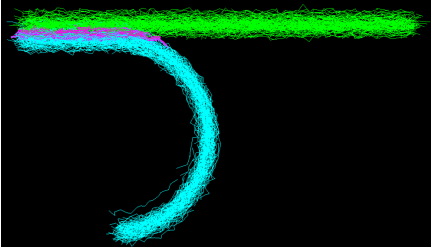


Figure C.5: Impact of noise to the linking quality using the quality measure QM_{EP} . The first (magenta), second (blue) and third (green) candidate that are proposed by the quality measure QM_{EP} are shown. The percentage of overall correct linking decisions is hereby represented by the first candidate. Fragmentation with 10% of deleted points ($\Theta[1, 0.10]$). (A) For the fragmentation, a bunch deletion with a mean bunch length of 1 is used. (B) A mean bunch length of 5 and, (C) a mean bunch length of 10 and (D) a mean bunch length of 15 is used.

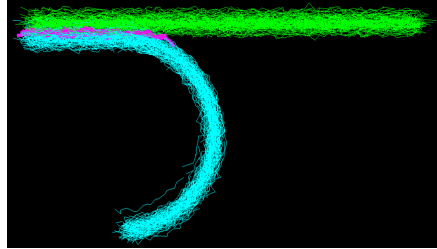
(A) Incorrect and not allocated number of tracks for different percentages of allocation.



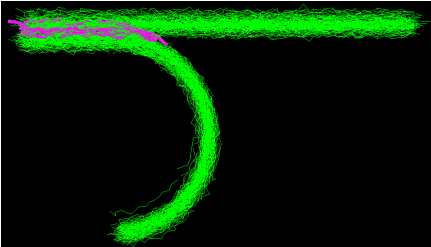
(B) $Q_{Alloc,PMC} = 100\%$ - Not Alloc.



(C) $Q_{Alloc,PMC} = 80\%$ - Not Alloc.



(D) $Q_{Alloc,PMC} = 80\%$ - Wrong Alloc.



(E) $Q_{Alloc,PMC} = 60\%$ - Wrong Alloc.

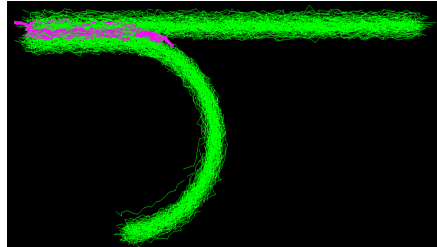


Figure C.6: Evaluation of the percentage of nearest neighbors in the allocation process. (A) For different percentages of allocation the number of wrong allocated trajectories ($Q_{Alloc,NWAT}$), as well as the number of not allocated trajectories ($Q_{Alloc,NNAT}$) are displayed. (B-C) The visualization of not allocated trajectories (magenta) for different percentages of allocation. The two groups of trajectories are colored in cyan and green. (D-E) The visualization of wrong allocated trajectories (magenta trajectories located in the border region) for different percentages of allocation. For the benchmark parametrization the benchmark database `TDB_S3` with touching groups is used. For the fragmentation all trajectories are used with a 10% of deletion ($\Theta[1, 0.1]$, $\kappa_{DA}[0]$, $ODM[\infty, 0]$). Hereby, a bunch deletion module ($QRM_{BB}[1, 0, 1]$) with a mean bunch length of 1, leading to short gaps is used for the total number of $N_T = 1655$ trajectories. For the neighborhood calculation, the 5 nearest neighbors are used ($Q_{Alloc,PRT} = 5$).

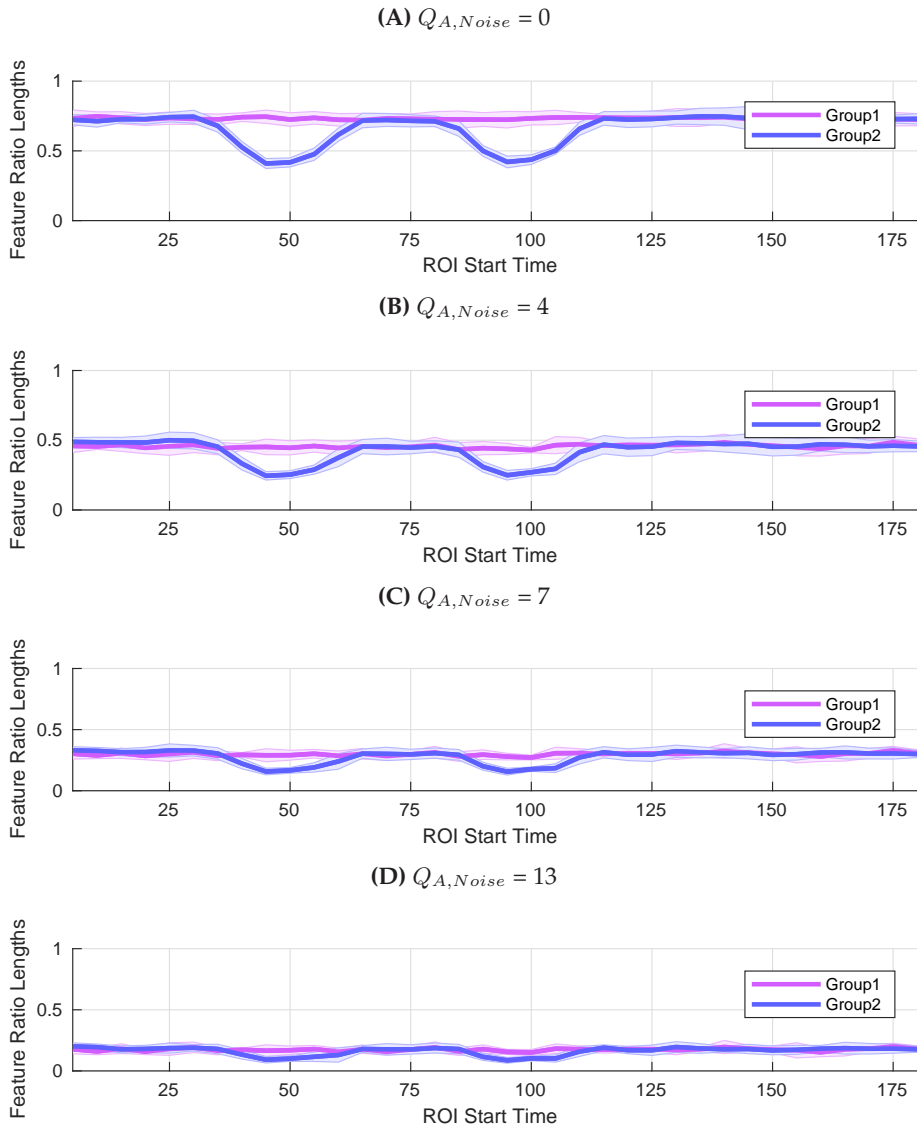


Figure C.7: The impact of the noise level to the extraction of the region of interest. Mean time series for two extracted groups (colored in blue and magenta) for the feature 'ratio lengths' in the benchmark database TDB_S6 . Different noise levels from (A) $Q_{A,Noise} = 0$ up to (D) $Q_{A,Noise} = 13$ are visualized.

Quantitative Description of Object Movement Behavior

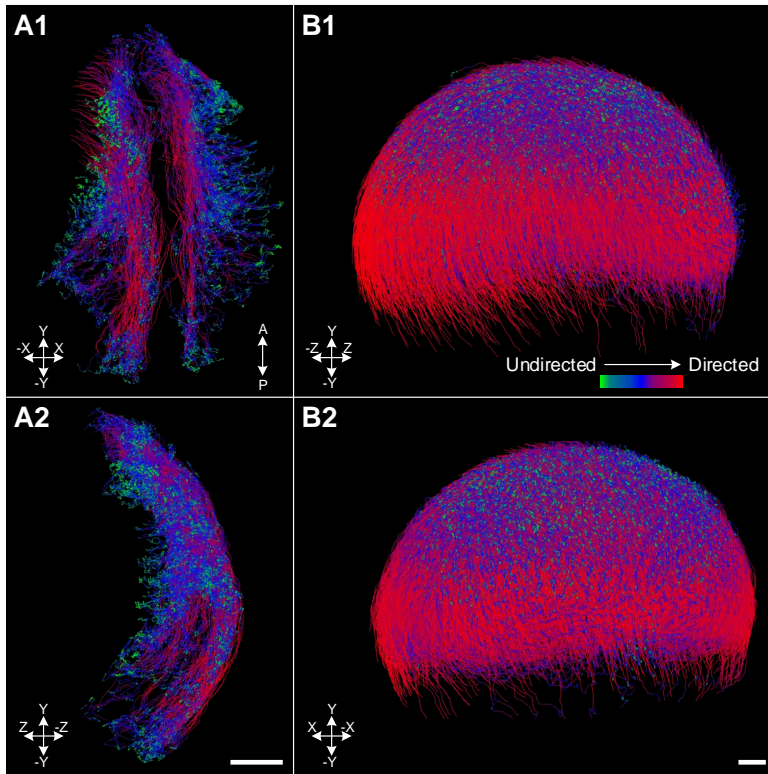


Figure C.8: Visualization of continuous trajectory features using a sliding window approach with window length n_W of 20. For the benchmark database TDB_S1 (A1-A2) and TDB_S2 (B1-B2) the single features describing the ratio of effective and total length are visualized for the sliding window feature calculation from two different perspectives (XY-plane (A1, B2) and YZ-plane (A2, B1)).

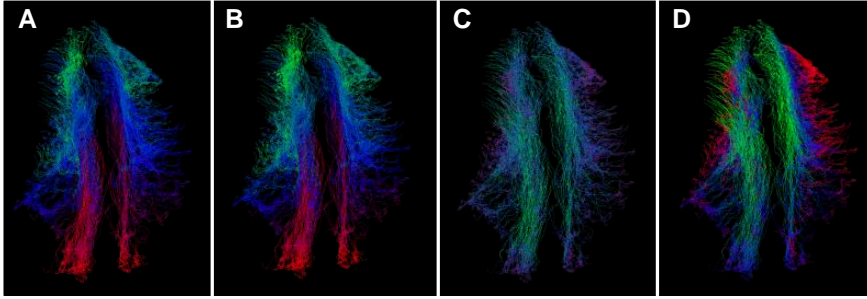


Figure C.9: Qualitative visual analysis of trajectory features using database `TDB_S3`. (A) Single features describing the ratio of effective and total length, (B) Single features describing the mean curvilinear speed, (C) Time series feature describing the directional change, (D) Time series features describing the sum of angles. All features are listed in Table 3.5

Deletion	MeanG1	StdG1	MeanG2	StdG2	p
0	0.762	0.012	0.483	0.009	3.74e-224
0.01	0.765	0.043	0.669	0.151	6.26e-24
0.02	0.771	0.057	0.725	0.117	1.74e-14
0.03	0.776	0.071	0.745	0.099	4.13e-11
0.04	0.775	0.074	0.761	0.094	0.000559
0.05	0.786	0.085	0.771	0.095	0.000122
0.1	0.8	0.102	0.797	0.106	0.36
0.15	0.818	0.116	0.816	0.115	0.538
0.2	0.831	0.125	0.828	0.126	0.281
0.25	0.846	0.128	0.845	0.128	0.707
0.3	0.856	0.13	0.859	0.131	0.234

Table C.3: Evaluation of group distinguishability in relation to the percentage of deletion for each trajectory using benchmark `TDB_S3`. For different percentages of deletion reaching from 0% up to 30% the mean and standard deviation for the two extracted groups (see Figure 3.17) is calculated for the feature 'ratio length'. A two-sided t-test with significance level of 0.0001 is applied as correction against multiple testing. P-values smaller than the significance level are highlighted in bold.

$Q_{A,Noise}$	MeanG1	StdG1	MeanG2	StdG2	p
0	0.99	0.00101	0.668	0.00621	5.62e-311
1	0.411	0.0189	0.352	0.0165	5.69e-59
2	0.218	0.0123	0.196	0.0118	1.76e-28
3	0.147	0.00964	0.133	0.00919	1.21e-19
4	0.11	0.00845	0.101	0.00789	1.21e-14
5	0.0884	0.00784	0.0809	0.00717	2.91e-11
6	0.0738	0.00748	0.0677	0.00675	7.08e-09
7	0.0633	0.00726	0.0582	0.00648	3.7e-07
8	0.0554	0.00711	0.051	0.0063	6.71e-06

Table C.4: Impact of noise artifacts to group distinguishability for each trajectory using benchmark TDB_S3. For different noise levels the mean and standard deviation for the two extracted groups illustrated in Figure 3.18A are calculated for the feature 'ratio length'. A two-sided t-test with significance level of 0.0001 is applied as correction against multiple testing. P-values smaller than the significance level are highlighted in bold.

Noise	MeanG1	StdG1	MeanG2	StdG2	p
0	1.01	0.00972	1.37	0.0101	9.33e-253
1	2.43	0.11	2.61	0.117	2.73e-22
2	4.59	0.229	4.7	0.256	0.000943
3	6.8	0.347	6.9	0.39	0.052
4	9.03	0.464	9.13	0.523	0.143
5	11.3	0.58	11.4	0.655	0.213
6	13.5	0.697	13.6	0.787	0.26
7	15.7	0.813	15.9	0.919	0.291
8	18	0.93	18.1	1.05	0.312

Table C.5: Impact of noise artifacts to group distinguishability for each trajectory using benchmark TDB.S3. For different noise levels the mean and standard deviation for the two extracted groups illustrated in Figure 3.18B are calculated for the feature 'curvilinear speed'. A two-sided t-test with significance level of 0.05 is applied. P-values smaller than the significance level are highlighted in bold.

Focusing on Spatio-Temporal Regions

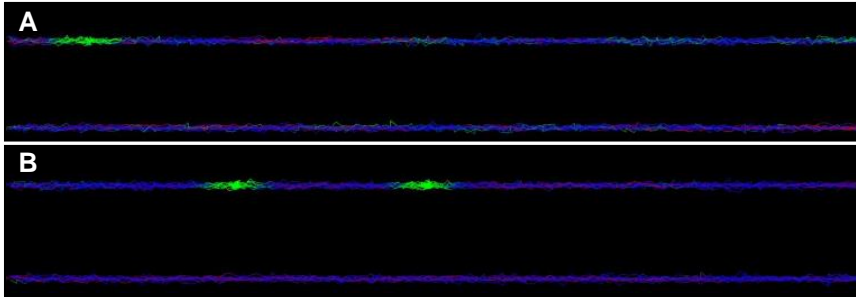


Figure C.10: Visual representation of the benchmark database `TDB_S6_D1` containing periods of undirected movement. (A) One period of undirected movement at time point 10 with a period length of 10 time points. (B) Two periods of undirected movement at time points 10 and 50 with both a period length of 10 time points. The color code indicates the effective lengths of a sliding window approach with window size of 10.

Time Point	MeanG1	StdG1	MeanG2	StdG2	p
1	0.77	0.0505	0.743	0.0458	0.224
6	0.786	0.0276	0.731	0.0415	0.00238
11	0.77	0.0305	0.747	0.0311	0.118
16	0.768	0.0303	0.743	0.0373	0.109
21	0.776	0.0341	0.757	0.0181	0.141
26	0.767	0.0329	0.768	0.0278	0.903
31	0.776	0.0365	0.706	0.0502	0.00211
36	0.794	0.0395	0.548	0.0307	7.35e-12
41	0.789	0.0419	0.433	0.033	3.8e-14
46	0.773	0.0391	0.444	0.0293	3.4e-14
51	0.782	0.0351	0.511	0.0565	1.62e-10
56	0.771	0.038	0.671	0.0478	6.34e-05
61	0.766	0.0388	0.775	0.0509	0.664
66	0.782	0.0288	0.773	0.0441	0.58
71	0.783	0.0297	0.764	0.0368	0.218
76	0.765	0.036	0.752	0.0391	0.457
81	0.777	0.0499	0.693	0.047	0.00115
86	0.778	0.04	0.531	0.038	3.35e-11
91	0.764	0.0534	0.453	0.0356	9.17e-12
96	0.784	0.0508	0.466	0.0312	1.77e-12
101	0.781	0.0485	0.54	0.0252	4.46e-11
106	0.774	0.0387	0.703	0.045	0.00125
111	0.781	0.0313	0.776	0.0433	0.79
116	0.774	0.0449	0.772	0.0499	0.949
121	0.772	0.0539	0.769	0.0552	0.894
126	0.766	0.0595	0.767	0.0587	0.964
131	0.768	0.0618	0.776	0.0584	0.778
136	0.768	0.0542	0.778	0.0417	0.641
141	0.757	0.0444	0.767	0.0588	0.673
146	0.753	0.0171	0.768	0.0696	0.501
151	0.761	0.0292	0.761	0.0608	0.998
156	0.752	0.0447	0.75	0.0353	0.899
161	0.763	0.037	0.75	0.0264	0.359
166	0.758	0.0487	0.754	0.0296	0.853
171	0.753	0.041	0.756	0.0395	0.849
176	0.752	0.0319	0.766	0.0387	0.392

Table C.6: Choice of region of interest with predefined appropriate size using benchmark TDB_S6 . The window size of the ROI is set to be 20 time points. There is no noise added to the benchmark ($Q_{A,Noise} = 0$). The mean and standard deviation values for the two groups shown in Figure 3.19A. For different start points of the ROI a two-sided t-test with significance level of 0.0001 is applied as correction against multiple testing. P-values smaller than the significance level are highlighted in bold.

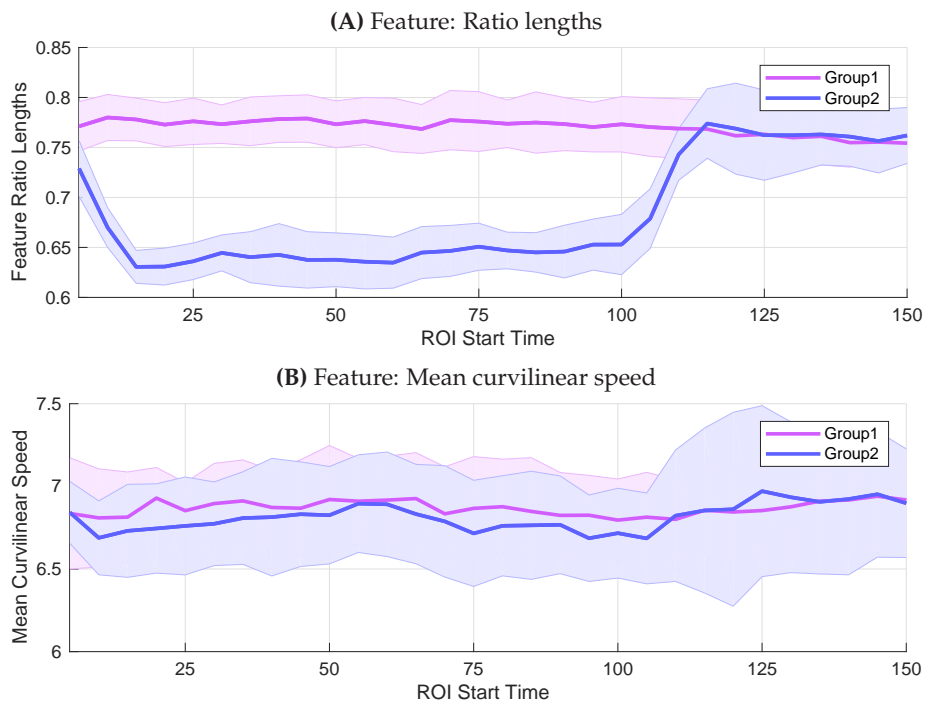


Figure C.11: Choice of region of interest with predefined size using benchmark TDB.S6. The window size of the ROI is set to be 50 time points. There is no noise added to the benchmark ($Q_{A,Noise} = 0$). (A) Mean time series of the two groups within the benchmark (colored in magenta and blue) for the feature 'ratio lengths'. (B) Mean time series of the two groups within the benchmark (colored in magenta and blue) for the features 'mean curvilinear speed'.

Time Point	MeanG1	StdG1	MeanG2	StdG2	p
1	0.729	0.0313	0.707	0.0206	0.0875
6	0.738	0.0293	0.65	0.0258	1.22e-06
11	0.737	0.027	0.608	0.0196	3.34e-10
16	0.726	0.0256	0.604	0.0219	1.06e-09
21	0.73	0.0245	0.606	0.0201	3.16e-10
26	0.727	0.0236	0.61	0.0232	1.48e-09
31	0.73	0.0258	0.604	0.0256	2.17e-09
36	0.73	0.0228	0.603	0.0288	2.39e-09
41	0.729	0.0256	0.597	0.0295	3.27e-09
46	0.727	0.0319	0.598	0.0283	1.89e-08
51	0.729	0.0266	0.597	0.0268	1.67e-09
56	0.722	0.0271	0.592	0.0271	2.81e-09
61	0.72	0.0234	0.601	0.0294	8.71e-09
66	0.73	0.0292	0.607	0.0317	4.28e-08
71	0.73	0.0328	0.611	0.029	8.13e-08
76	0.73	0.0317	0.608	0.0247	1.78e-08
81	0.729	0.0318	0.606	0.0266	2.35e-08
86	0.732	0.0293	0.607	0.0274	1.13e-08
91	0.732	0.0269	0.615	0.0249	8.52e-09
96	0.734	0.0314	0.618	0.0267	5.26e-08
101	0.734	0.0293	0.641	0.0262	6.35e-07
106	0.739	0.0282	0.703	0.038	0.0297
111	0.735	0.0274	0.737	0.0493	0.911
116	0.731	0.0277	0.73	0.0566	0.936
121	0.728	0.0265	0.722	0.0517	0.755
126	0.726	0.0354	0.726	0.0448	0.99
131	0.729	0.0319	0.729	0.0409	0.975
136	0.723	0.0369	0.724	0.0414	0.936
141	0.724	0.0303	0.721	0.0415	0.834
146	0.72	0.031	0.722	0.0326	0.931

Table C.7: Choice of region of interest with predefined wrong size using benchmark TDB_S6. The window size of the ROI is set to be 50 time points. There is no noise added to the benchmark ($Q_{A,Noise} = 0$). The mean and standard deviation values for the two groups shown in Figure 3.19B. For different start points of the ROI a two-sided t-test with significance level of 0.0001 is applied as correction against multiple testing. P-values smaller than the significance level are highlighted in bold.

Size of ROI	MeanG1	StdG1	MeanG2	StdG2	p
5	0.756	0.108	0.675	0.0735	0.0633
10	0.761	0.0552	0.502	0.0773	8.09e-08
15	0.765	0.0437	0.61	0.0731	1.87e-05
20	0.763	0.0362	0.659	0.0481	3.49e-05
25	0.769	0.0337	0.68	0.0377	2.9e-05
30	0.768	0.0311	0.692	0.0458	0.000391
35	0.762	0.0266	0.703	0.0373	0.000717
40	0.764	0.0222	0.71	0.0353	0.000596
45	0.768	0.0235	0.712	0.0296	0.0002
50	0.765	0.0241	0.716	0.0269	0.000388
55	0.768	0.0247	0.719	0.0252	0.000311
60	0.769	0.0293	0.728	0.0221	0.00217
65	0.769	0.0232	0.732	0.0169	0.000721
70	0.768	0.0238	0.733	0.0195	0.00218
75	0.769	0.0204	0.737	0.0174	0.00121
80	0.77	0.0202	0.741	0.0165	0.00227
85	0.768	0.0188	0.741	0.0166	0.00302
90	0.767	0.0148	0.743	0.0184	0.00612
95	0.769	0.0148	0.746	0.0194	0.00881
100	0.767	0.0148	0.747	0.0192	0.0151
105	0.765	0.0151	0.746	0.0213	0.0379
110	0.766	0.0141	0.744	0.0215	0.0181
115	0.766	0.0147	0.746	0.0199	0.0168
120	0.765	0.016	0.745	0.0179	0.0209
125	0.765	0.0162	0.746	0.0183	0.0252
130	0.764	0.0155	0.748	0.0176	0.038
135	0.764	0.0153	0.748	0.0173	0.0477
140	0.764	0.0159	0.749	0.0151	0.041
145	0.764	0.0144	0.748	0.0148	0.024
150	0.765	0.0149	0.75	0.0132	0.0299
155	0.765	0.0136	0.748	0.0122	0.00976
160	0.764	0.0146	0.747	0.0117	0.0124
165	0.766	0.0138	0.747	0.00972	0.00297
170	0.765	0.0137	0.747	0.0102	0.00354
175	0.765	0.014	0.748	0.0101	0.00483
180	0.765	0.0122	0.748	0.01	0.0031

Table C.8: Impact of the correct choice of the size of the region of interest using benchmark TDB_S6. The size of the region of interest varies starting from 5 time points up to 190 time points. For each size of the ROI, the mean and standard deviation values for the features 'ratio length' is calculated. A two-sided t-test with significance level of 0.0001 is applied as correction against multiple testing. P-values smaller than the significance level are highlighted in bold.

Feature-Based Extraction of Groups of Trajectories

Method	TDB_S8	TDB_S9	TDB_S10	TDB_S11
OS_{Autom}	0.333/3.039	0.167/3.072	0.125/3.087	0.125/3.092
OS_{Prior}	1/2.493	0.442/2.547	0.472/2.901	0.125/2.962
MS_{Prior}	1/25	1/39	1/51	1/59
$MS_{No-Prior}$	1/128	1/253	1/521	1/534

Table C.9: The impact of one-step and multi-step analysis approaches combined with the integration of prior knowledge. For four different benchmark databases TDB_S8, TDB_S9, TDB_S10 and TDB_S11 a fully automatic approach (OS_{Autom}), a one-step approach using prior knowledge (OS_{Prior}), a multi-step approach using prior knowledge (MS_{Prior}) and a multi-step approach without the existence of prior knowledge ($MS_{No-Prior}$) are validated using the required time effort and the accuracy of group assignments. The entries of the table indicate the accuracy versus the time effort in seconds.

D Interactivity in Knowledge Discovery of Large-Scale Trajectory Databases

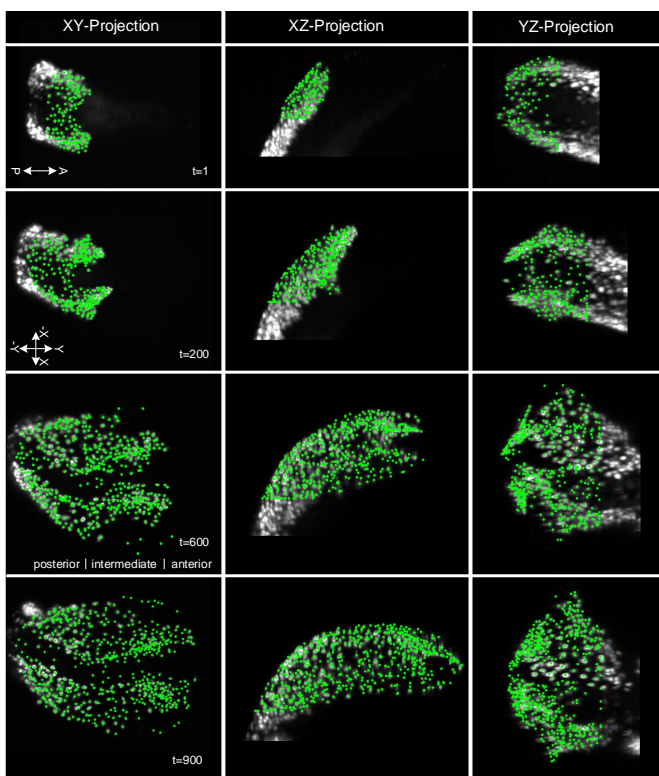


Figure D.1: Temporally scrollable maximum intensity projections for benchmark database `TDB_S2` displaying different planes (XY-, XZ-, YZ-Plane) with superimposed segmentation results from detected objects (green dots).

E Implementation Details

Quickstart Guide for EmbryoMiner

A quickstart guide for EmbryoMiner on the basis of the paper published in <https://doi.org/10.1371/journal.pcbi.1006128> [348]. Sample data is provided that can be used to understand the presented components of the framework including all visualization modules described in Chapter 4, selection possibilities, application of data mining methods (Chapter 3) and data import (Section 5.1.6). As a prerequisite to run EmbryoMiner, make sure MATLAB and SciXMiner are properly installed. The latest release of SciXMiner can be obtained from <https://sourceforge.net/projects/scixminer/> and general information as well as installation instructions for SciXMiner are provided in the following document: <https://arxiv.org/abs/1704.03298>. To install EmbryoMiner and to run the examples, perform the following steps:

1. Download and install the EmbryoMiner toolbox from <https://sourceforge.net/projects/scixminer/files/Extension%20packages/> by extracting the `embryominer.zip` archive to the `application_specials` folder of your SciXMiner installation. This extension package contains all required components of the interactive knowledge discovery framework EmbryoMiner.
2. Open MATLAB and start SciXMiner by typing `scixminer` in the command line window of MATLAB. If you don't see the EmbryoMiner entry in the menu bar, open `Extras` → `Choose application-specific extension packages ...` and activate the tracking toolbox. After restarting SciXMiner, the tracking toolbox should properly load and everything is ready for running the application examples.

3. Download the application examples from <https://sourceforge.net/projects/scixminer/files/InteractiveKnowledgeDiscovery/> and extract them to a folder of your choice.
4. Return to the SciXMiner GUI and select one of the examples from the `InteractiveKDEExamples` folder with the extension `*.batch`. Use the menu entry `File → Apply SciXMiner Batch File` and select the example of your choice.
5. The `*.batch` file automatically loads the demo projects as well as the visualization windows. Batch files can be opened in a standard text editing software, in case you're interested in understanding or changing the code.
6. Further information is provided in the readme files associated with each of the examples. The application examples comprise an overview of the visualization possibilities, interactive selection capabilities, application of data mining methods to cell tracking data, track filtering as well as the import of tracking data generated by other tools.

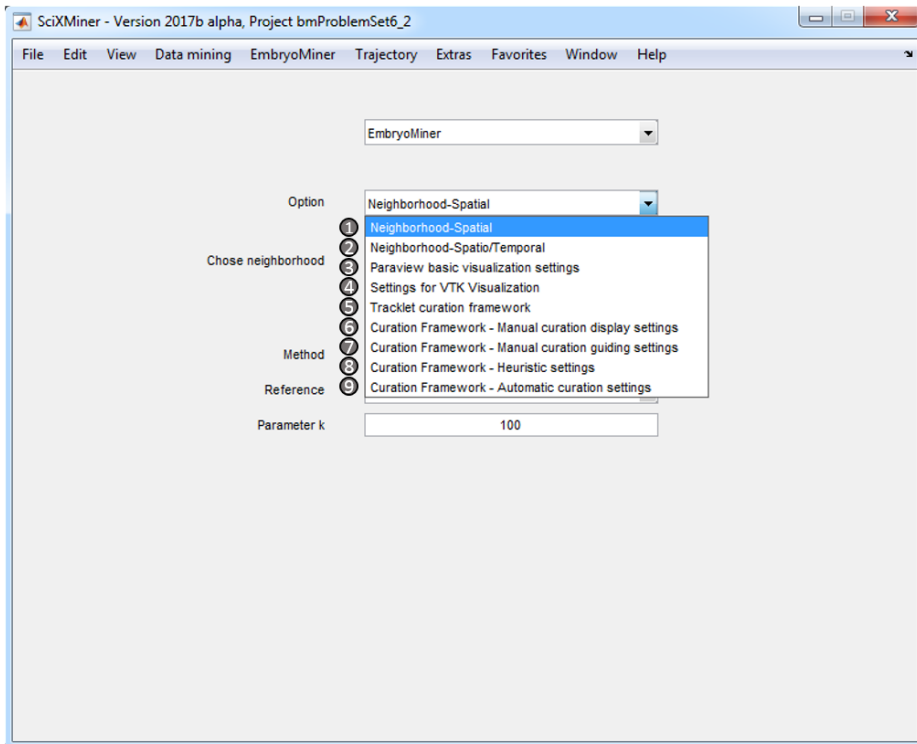


Figure E.1: Control elements in the SciXMiner toolbox EmbryoMiner to interactively adapt parameters in the implemented algorithms. Here, the main control elements can be used to adapt: (1) Spatial neighborhood, (2) spatio-temporal neighborhood, (3) basic visualization settings, (4) settings for the visualization windows, (5) tracklet curation, (6) display settings for curation, (7) guiding settings for curation, (8) heuristic settings and (9) settings for automatic curation.

F Evaluation of the Knowledge Discovery Framework

Survey on the Evaluation of the Interactive Knowledge Discovery Framework for Large-scale 3D+t Trajectory Databases

1. Prior experience with the underlying software framework.

	Not at all 1	2	3	4	Very much 5
1.1 I have prior experience in using the software SciXMiner					
1.2 I have prior knowledge in analyzing tracking data					
1.3 I have prior knowledge in software programming					

2. Usability of the interactive knowledge discovery framework.

	Very low 1	2	3	4	Very high 5
2.1 How high was the training effort for the interactive framework?					
2.2 How high is the effort of transferring already applied approaches to new trajectory data?					
2.3 How big was the knowledge growth in using the interactive framework after applying the proposed methods in a 15-minute session yourself?					

Survey on the Evaluation of the Interactive Knowledge Discovery Framework for Large-scale 3D+t Trajectory Databases, state 01.06.2016, B. Schott, contact: benjamin.schott@kit.edu

Figure F.1: Questionnaire for the evaluation of the interactive knowledge discovery framework - page 1.

3. Which of the proposed methods tailored for large-scale trajectory databases offer a great potential for the analysis of tracking data in general?

	Not at all 1	2	3	4	Very much 5	I don't know
3.1 Feature-based description of trajectory data						
3.2 Focusing on spatio-temporal regions within trajectory databases						
3.3 Interactive curation of fragmented tracking data						
3.4 Feature-based extraction of groups using clustering and filtering approaches						
3.5 Allocation process to handle fragmented trajectory data within analysis						
3.6 Quantitative comparison of multiple groups						
3.7 Quantitative neighborhood calculation						
3.8 Transfer of analysis pipelines to new trajectory databases						

4. Which of the proposed interactive visual representations of trajectory data offer a great potential for the analysis of tracking data in general?

	Not at all 1	2	3	4	Very much 5	I don't know
4.1 3D rendering of large-scale object trajectory data						
4.2 Maximum intensity projection overlay with raw data						
4.3 3D+t visualization of migrating objects over time						
4.4 Scatter plot for track-based feature selections						
4.5 Lineage visualization of dividing object characteristics						
4.6 Interactive visual data exploration using the hierarchical analysis process						
4.7 Visual guided interactive curation of tracking error						

Survey on the Evaluation of the Interactive Knowledge Discovery Framework for Large-scale 3D+t Trajectory Databases, state 01.06.2018, B. Schott, contact: benjamin.schott@kit.edu

Figure F2: Questionnaire for the evaluation of the interactive knowledge discovery framework - page 2.

5. Impact of the newly developed knowledge discovery framework in biological cell tracking applications

	Not at all 1	2	3	4	Very much 5
5.1 Are software approaches already used to analyze existing cell tracking data?					
5.2 Is there a need for software approaches handling cell tracking data					

6. Applicability in the context of biological cell tracking application

	Percent
For how many percentage of your cell tracking applications is the interactive knowledge discovery framework useful?	

7. What are the new emerging possibilities in biological applications, especially cell tracking, using the interactive knowledge discovery framework?

8. Using these new possibilities in cell tracking application which new conclusions can be made on the newly gained quantitative results?

Survey on the Evaluation of the Interactive Knowledge Discovery Framework for Large-scale 3D+t Trajectory Databases, state 01.06.2016, B. Schott, contact: benjamin.schott@kit.edu

Figure F.3: Questionnaire for the evaluation of the interactive knowledge discovery framework - page 3.

9. Using the possibilities of the knowledge discovery framework, what may possibly change for future biological experiments compared to the current state?

10. Are there any suggestions for improvement? If yes, which suggestions?

Survey on the Evaluation of the Interactive Knowledge Discovery Framework for Large-scale 3D+t Trajectory Databases, state 01.06.2018, B. Schott, contact: benjamin.schott@kit.edu

Figure F.4: Questionnaire for the evaluation of the interactive knowledge discovery framework - page 4.

Use-Case 2: Multilayer 3D Trajectory Selection - High complexity**Problem definition**

Separate spatially located groups of trajectories in 3D that can not be distinguished using a one-step approach for further ongoing analysis of selected groups in detail.

Task description

The user has to select three different groups in total. Two groups are moving straight, out of which one have a short period of undirected movement behavior. The third groups performs an internalization movement (Benchmark TDB_S08 depicted in Section 2.4.2).

Required tools

To successfully accomplish the presented task the following functionality developed in this thesis can be applied:

- 3D rendering of large-scale trajectory data (Section 4.1.1)
- Scatter plot for trajectory-based feature handling (Section 4.1.4)
- Interactive trajectory selection possibilities (Section 4.2.1)
- Online propagation of intermediate selection results (Section 4.2.2)
- Interactive visual tree view (Section 4.3)

Target output

The requested output are three selected groups color-coded in different colors in Figure F.5.

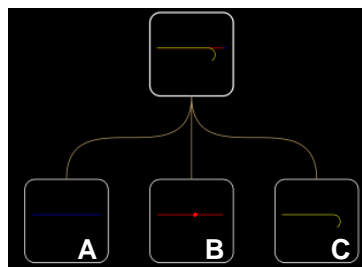


Figure F.5: Target output to validate the interactive framework: Hierarchical selection process. Here, three different groups are contained in the benchmark database.

Use-Case 3: Multilayer 3D Trajectory Selection - Enormous Complexity

Problem definition

Separate spatially located groups of trajectories in 3D that can not be distinguished using a one-step approach for further analysis of selected groups in detail.

Task description

The user has to select eight different groups in total. Four groups are moving straight, out of which two have a short period of clustering. The other four groups perform an internalization movement at the same time point with two out of these four groups additionally clustering. The user has to select the eight different groups in benchmark database TDB_S10 (Section 2.4.2) in the 3D visual representation of the trajectories.

Required tools

To successfully accomplish the presented task the following functionality developed in this thesis can be applied:

- 3D rendering of large-scale trajectory data (Section 4.1.1)
- Scatter plot for trajectory-based feature handling (Section 4.1.4)
- Interactive trajectory selection possibilities (Section 4.2.1)
- Online propagation of intermediate selection results (Section 4.2.2)
- Interactive visual tree view (Section 4.3)

Target output

The requested output are eight selected groups color-coded in different colors in Figure F.6.

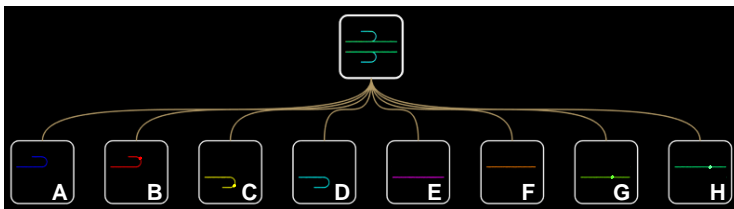


Figure F.6: Target output to validate the interactive framework: Hierarchical selection process.

Use-Case 4: Maximum Intensity Projection - Spatio-temporal Tailored Selection**Problem definition**

Allocate highly specific prior knowledge, in terms of spatial occurrence within a temporal region of possible interest, directly on the single object level.

Task description

The user has to select two different groups of objects that subsist at a specific time interval ($t=600$) within the data. The groups are characterized by complex spatial boundaries within highly dense regions of interest (Benchmark TDB_S02 as depicted in Section 2.4.2 is used for validation purpose).

Required tools

To successfully accomplish the presented task the following functionality developed in this thesis can be applied:

- Maximum intensity projection overlay (Section 4.1.2)
- 3D+t visualization of migrating objects (Section 4.1.3)
- Interactive trajectory selection possibilities (Section 4.2.1)
- Online propagation of intermediate selection results (Section 4.2.2)

Target output

The requested output are the two groups characterized by the complex spatial boundaries as illustrated in Figure F.7.

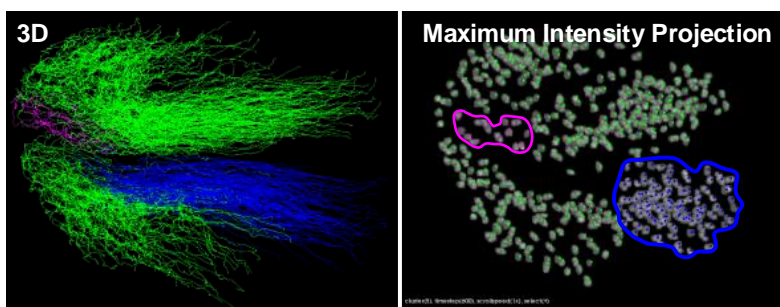


Figure F.7: Target output to validate the interactive framework: Object-specific allocation of prior knowledge using the maximum intensity projection overlay.

Use-Case 5: Feature-based Selection Strategies

Problem definition

Extract 3D trajectories exhibiting a predefined spatio-temporal movement pattern in a predefined range allowing to efficiently focus and validate specific groups characterized by underlying patterns.

Task description

The user has to select trajectories exhibiting a ratio of their effective length and total length (A detailed description of the features can be found in Section 3.3.3) in the range between 0 and 0.4 (Benchmark `TDB_S02` as depicted in Section 2.4.2) is used for validation purpose).

Required tools

To successfully accomplish the presented task the following functionality developed in this thesis can be applied:

- 3D rendering of large-scale trajectory data (Section 4.1.1)
- Scatter plot for trajectory-based feature handling (Section 4.1.4)
- Interactive trajectory selection possibilities (Section 4.2.1)
- Online propagation of intermediate selection results (Section 4.2.2)

Target output

The requested output is one group exhibiting the defined characteristic within the predefined feature range as depicted in Figure F.8.

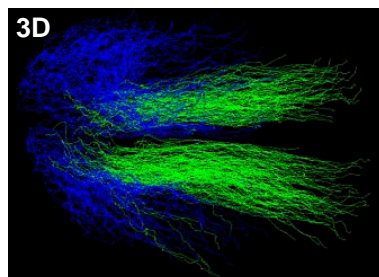


Figure F.8: Target output to validate the interactive framework: Feature-based selection strategies

Study coordinator:
Karlsruhe Institute of Technology
Institute for Automation and Applied Informatics
Group for Automated Image and Data Analysis
Hermann-von-Helmholtz-Platz 1
76344 Eggenstein-Leopoldshafen
M.Sc. Benjamin Schott
Tel: +49 721 608-26672
Fax: +49 721 608-22602
Mail: benjamin.schott@kit.edu

Information for the test users

Survey on the evaluation of the interactive knowledge discovery framework for large-scale 3D+t trajectory databases

Dear test users,

the working group for Automated Image and Data Analysis of the Karlsruhe Institute of Technology develops software frameworks in the field of image and data analysis.

In this study the usability and interactivity of a developed interactive knowledge discovery framework for large-scale 3D+t trajectory databases is investigated. Here, a wide variety of conceptual approaches for the analysis of 3D tracking data sets can be used to investigate moving object characteristics and the corresponding pattern of single tracks or groups of tracks.

Before the users starts to perform the use-cases developed for the study, an introduction of around 20 minutes is given to user to introduce the functionality of the knowledge discovery software framework and to show how to operate on the software framework *EmbryoMiner*. In the next step, the users are asked to use the *EmbryoMiner* framework to fulfill five uses cases one after the other. After each use-case approximately 60 seconds is planned for a short rest period. Furthermore, the time required for each use case is measured, as well as the quality in terms of the percentage of correct assigned tracks. During the complete test period the study coordinator is present. After the application to all five use-cases, the user is asked to complete a questionnaire. For the session of completing the questionnaire 5 minutes are proposed.

All use-cases for one test user are done in a stretch resulting in a complete time interval for the study of around 90 minutes in total. The appointments for the study is arranged in advance. There is no insurance for the application of this study.

Figure F.9: Proband information

Study coordinator:
Karlsruhe Institute of Technology
Institute for Automation and Applied Informatics
Group for Automated Image and Data Analysis
Hermann-von-Helmholtz-Platz 1
76344 Eggenstein-Leopoldshafen
M.Sc. Benjamin Schott
Tel: +49 721 608-26672
Fax: +49 721 608-22602
Mail: benjamin.schott@kit.edu

Information about privacy protection

Dear test users,

During the complete study no contact information is collected. The measured result for the study are instantaneously anonymized.

The participation in the study is voluntarily. There are no disadvantages that arise for the user during the study. Also in the case of an already signed declaration of consent, the study can be aborted without mentioning any reasons. In case of abortion, all collected data are deleted.

The depersonalized research information are deleted 10 years after the last published publication.

The research result are exclusively used in an aggregated form.

In case of further questions about the procedure of the study or if contents remain unclear, please contact the study coordinator. If there are any open questions after the study you can also contact the study coordinator.

(Study coordinator M.Sc. Benjamin Schott)

Figure F.10: Information about the privacy protection

Study coordinator:
Karlsruhe Institute of Technology
Institute for Automation and Applied Informatics
Group for Automated Image and Data Analysis
Hermann-von-Helmholtz-Platz 1
76344 Eggenstein-Leopoldshafen
M.Sc. Benjamin Schott
Tel: +49 721 608-26672
Fax: +49 721 608-22602
Mail: benjamin.schott@kit.edu

Declaration of consent

Survey on the evaluation of the interactive knowledge discovery framework for large-scale
3D+t trajectory databases

(Name of the user)

Mr. Benjamin Schott explained me today in a detailed explanatory meeting the content and the scope of this user study. Here, the aim of the study as well as the duration of the study were discussed. I received and read the proband information and a copy of the declaration of consent. Existing questions in this context were discussed and answered. I had enough time to decide whether to participate in the study or not. I was informed that there is no special insurance during the complete study.

I agree to participate in this study as a test user.
I'm aware of the fact that the primarily aim of this study is the knowledge enhancement and that there might be no personal advantages.

I was informed that my attendance to the study is completely voluntary. Furthermore, I was informed that I can revoke my declaration of consent at any time without giving reasons and without any personal disadvantage.

I have read the proband information and especially the information sheet about the **privacy protection** and all of my questions were fully answered.

Figure F.11: Declaration of consent - page 1.

I'm aware that personal data is collected, saved and evaluated in this study. The use of the personal data follows legal regulations and required the following voluntarily signed declaration of consent. Without the signed declaration of consent I cannot participate in this study.

I agree that personal data that is described in "Information about privacy protection" is collected, saved and evaluated in the context of this study in an anonymized way with the purpose described in "Proband information".

(Name of the user)

Place, Date, Signature

(Name of study coordinator)

Place, Date, Signature

Figure F.12: Declaration of consent - page 2.

F.1 Biological-tailored Evaluation

Answers: Biological-tailored topics in the questionnaire

Question 5.1:

Are software approaches already used to analyze existing cell tracking data?

Answer: Not at all

Note: The data is often too big for existing commercial or Java software platform in combination with a lack of efficient integration of prior knowledge.

Question 5.2:

Is there a need for software approaches handling cell tracking data?

Answer: Very much

Note: Cell tracking analysis software is the key technology for developmental biology.

Question 6:

For how many percentage of your cell tracking applications is the interactive knowledge discovery framework useful?

Answer: 100 %

Note: The knowledge discovery framework developed in this thesis is a versatile tool for any kinds of tracking data

Question 7:

What are the new emerging possibilities in biological applications, especially cell tracking, using the interactive knowledge discovery framework?

Answers:

- Analysis of cytoplasmic or membrane labelling to elucidate the relationship between the cell shape change, which reflects the force either received or generated from a cell of interest and the direction of cell migration
- Long term and comprehensive lineage analysis to understand how organs (eye, spinal cord, brain, etc.) is formed. This may lead to the discovery of adult stem cells. Once the lineage is established with one embryo, it can be compared with other embryos to see individual variations. Next step will be the comparison with genetic mutant or combination with cell ablation study
- Combination of nucleus labelling and tracking with other markers
- Signaling reporter transgenic lines (Wnt, BMP, Shh, Notch, etc.)

- High resolution images with sub-micrometer per voxel scale will also get interesting
- In addition to nucleus or membrane labelling, other sub-cellular markers, such as mitochondria, centromere and cytoskeleton (actin, myosin) will be probably used
- The 3D+t registration techniques of different individual embryos. This allows to compare 2 groups of data sets, for example wild type and genetic mutant embryos in a quantitative fashion

List of Figures

1.1	Direct measurement techniques to create trajectory databases	4
1.2	Indirect measurement techniques to create trajectory databases	5
1.3	Pipeline to process light sheet microscopy images	6
1.4	Pipeline to process light sheet microscopy images for tracking	7
1.5	Knowledge discovery in trajectory databases	12
1.6	Scheme for visual explorative data analysis	20
2.1	Systematization of reasons for fragmented tracking data	32
2.2	Synthetic creation of quality dependent benchmark databases	33
2.3	Markov model to simulate fragmentation in trajectory databases	36
2.4	Parameter examples of Markov-bases quality reduction module	37
2.5	Parameter examples of Bunch-bases quality reduction module	38
2.6	General bunch-deletion process	39
2.7	Object division simulation module	40
2.8	Fragmentation artifacts through object division	41
2.9	Fragmentation artifacts through inserted deletion	42
2.10	Systematic generation of validation benchmarks	43
2.11	Generated trajectory benchmarks for validation purpose	46
3.1	Overview of the knowledge discovery process	48

3.2	Scheme for the knowledge discovery framework	51
3.3	Conceptual approaches to analyze large-scale trajectory databases . .	56
3.4	Impact of fragmentation to the linking quality of quality measures .	60
3.5	Impact of noise to the linking quality of quality measures	61
3.6	Neighborhood calculation for selected trajectories	63
3.7	Distinction of neighborhood cases	64
3.8	The effect of the size of neighborhood	66
3.9	The effect of the choice of representative trajectories	67
3.10	Allocation process in trajectory databases	68
3.11	The impact of spatial separability in the allocation process	70
3.12	Fuzzy membership for trajectory datasets	71
3.13	Correlation of representative and wrong allocated tracks	73
3.14	Impact of strategies to representative tracks	74
3.15	Evaluation of nearest neighbors in the allocation process	76
3.16	The effect of nearest neighbors in the allocation process	78
3.17	Evaluation of group distinguishability using the ratio of lengths . . .	82
3.18	Impact of noise artifacts to group distinguishability	83
3.19	Choice of region of interest with prior defined appropriate size . . .	84
3.20	Impact of the correct size of the region of interest	85
3.21	Feature-based group extraction methods	87
3.22	Time series clustering on trajectory data	87
3.23	Impact of one-step and multi-step analysis approaches	89
3.24	General scheme for the application of a KD-process to new databases	90
3.25	Transfer of analysis pipelines	91

3.26	Comparison of multiple groups within trajectory databases	92
3.27	Quantitative lineage analysis of dividing objects.	94
3.28	Homogeneity assumptions for trajectory neighborhoods	95
3.29	Neighborhood-based trajectory feature description	96
4.1	Highly responsive 3D rendering of trajectory data	101
4.2	Maximum intensity projection overlay	103
4.3	Superimposition of segmentation results in 3D	105
4.4	Scatter plot for trajectory features	106
4.5	Visual representation of dividing object characteristics	107
4.6	Interactive selection strategies in large-scale trajectory databases . . .	109
4.7	Propagation process of selection in trajectory databases	111
4.8	Hierarchical selection tree structure	114
4.9	Interactive transfer of analysis pipelines	115
4.10	Interactive manual curation process	117
4.11	Curation process of erroneous tracks	119
5.1	Graphical user interface of the software toolbox EmbryoMiner	124
5.2	Functionality of the visualization framework	126
5.3	Overview of implemented trajectory features	127
5.4	Functionality of lineage analysis	128
5.5	Import tracking data from existing platforms	132
6.1	Design of the validation study	134
6.2	Use-Case: 3D selection strategies	136

6.3	Evaluation of the time effort for the user study	138
6.4	Evaluation of the questionnaire for the user study	141
7.1	Analysis framework to analyze large-scale biological tracking data	144
7.3	Conceptual approaches to analyze cells in zebrafish development	147
7.4	Cell movement characterization using trajectory features	148
7.5	Transfer of analysis pipelines in wild-type zebrafish embryos	150
7.6	Quantitative comparison of tissue deformation within embryos	151
7.7	Quantitative comparison of tissue deformation between embryos	154
7.8	Tracking of neural crest cells	156
7.9	Conceptual approaches to analyze cranial neural crest cells	157
7.10	Quantitative validation of manual curation module	159
7.11	Tracking correction of precursor cells	160
7.12	Curation result for cranial neural crest cells	162
C.1	Impact nearest endpoint heuristic on fusion result	178
C.2	Impact nearest neighbor heuristic on fusion result	178
C.3	Impact of noise to the linking quality	179
C.4	Impact nearest endpoint heuristic on fusion result	180
C.5	Impact of noise to the linking quality	181
C.6	Influence of nearest neighbors in the allocation process	182
C.7	Impact of noise to region of interest extraction	183
C.8	Visualization of continuous trajectory features	184
C.9	Qualitative visual analysis of trajectory features	185
C.10	Visual representation of undirected movement characteristics	188

C.11	Choice of region of interest with wrong initial size	190
D.1	Maximum intensity projection overlay for three planes	195
E.1	Control elements for parameter adjustment	199
F.1	Questionnaire for the evaluation of the interactive knowledge discovery framework - page 1.	202
F.2	Questionnaire for the evaluation of the interactive knowledge discovery framework - page 2.	203
F.3	Questionnaire for the evaluation of the interactive knowledge discovery framework - page 3.	204
F.4	Questionnaire for the evaluation of the interactive knowledge discovery framework - page 4.	205
F.5	Use-Case: Hierarchical selection process	206
F.6	Use-Case: Extract hierarchical ordered groups	207
F.7	Use-Case: Object-specific allocation	208
F.8	Use-Case: Feature-based selection strategies	209
F.9	Proband information	210
F.10	Privacy declaration	211
F.11	Declaration of consent - page 1	212
F.12	Declaration of consent - page 2	213

List of Tables

1.1	Similarity measures for trajectory data	13
1.2	Generic trajectory patterns	14
1.3	Group trajectory patterns	15
1.4	Clustering techniques for trajectory data	16
2.1	Systematization of problem classes	30
2.2	Parameters for the quality reduction modules	44
3.1	Identification of suitable prior knowledge	54
3.2	Applicability of conceptual approaches to trajectory databases	58
3.3	Impact of manual and random selection strategies	75
3.4	Evaluation of quality of allocation	77
3.5	Quantitative movement description of trajectories	80
4.1	Impact of modifications on the transfer of analysis pipelines	116
4.2	Applicability of visual representation of 3D+t trajectory data	121
5.2	Comparison of trajectory visualization and analysis software tools	130
6.1	Questionnaire for the user study	137
B.1	Developed trajectory benchmarks for validation	174

- B.2 Developed trajectory benchmarks for validation 175
- B.3 Simulated trajectory benchmark databases 176

- C.2 Influence of number nearest neighbors to allocation result 179
- C.3 Evaluation of group distinguishability in relation to fragmentation artifacts using the ratio of lengths 185
- C.4 Impact of noise to group distinguishability 186
- C.5 Impact of noise to group distinguishability 187
- C.6 Choice of region of interest with predefined size 189
- C.7 Choice of region of interest with wrong chosen size 191
- C.8 Impact of the correct size of the region of interest 192
- C.9 Accuracy and time effort for different approaches to incorporate prior knowledge 193

Bibliography

- [1] LI, X.; HU, W.; HU, W.: A Coarse-to-Fine Strategy for Vehicle Motion Trajectory Clustering. In: *18th International Conference on Pattern Recognition*, (August 20-24, Hong Kong, China), IEEE, Piscataway, NJ, USA, 2006, p. 591–594, DOI: 10.1109/ICPR.2006.45.
- [2] LIN, M.; HSU, W.-J.: Mining GPS Data for Mobility Patterns: A Survey. *Pervasive and Mobile Computing* 12 (2014) 1, p. 1–16, DOI: 10.1016/j.pmcj.2013.06.005.
- [3] ANDRIENKO, G.; ANDRIENKO, N.; FUCHS, G.; GARCIA, J. M. C.: Clustering Trajectories by Relevant Parts for Air Traffic Analysis. *IEEE Transactions on Visualization and Computer Graphics* 24 (2017) 1, p. 34–44, DOI: 10.1109/TVCG.2017.2744322.
- [4] MEIJERING, E.; DZYUBACHYK, O.; SMAL, I.; ET AL.: Methods for Cell and Particle Tracking. *Methods Enzymol* 504 (2012) 9, p. 183–200, DOI: 10.1016/B978-0-12-391857-4.00009-4.
- [5] ZHENG, Y.: Trajectory Data Mining: An Overview. *ACM Transactions on Intelligent Systems and Technology* 6 (2015) 3, p. 29, DOI: 10.1145/2743025.
- [6] ZHENG, Y.; ZHOU, X.: *Computing with Spatial Trajectories*. Springer, London, UK, DOI: 10.1007/978-1-4614-1629-6, 2011.
- [7] JIANG, F.; WU, Y.; KATSAGGELOS, A. K.: A Dynamic Hierarchical Clustering Method for Trajectory-based Unusual Video Event Detection. *IEEE Transactions on Image Processing: A publication of the IEEE Signal Processing Society* 18 (2009) 4, p. 907–913, DOI: 10.1109/TIP.2008.2012070.
- [8] ZHOU, H.; TAO, D.; YUAN, Y.; LI, X.: Object Trajectory Clustering via Tensor Analysis. In: *16th IEEE International Conference on Image Processing*, (November 7-10, Cairo, Egypt), IEEE, Piscataway, NJ, USA, 2009, p. 1945–1948, DOI: 10.1109/ICIP.2009.5414536.

- [9] IVANOV, Y. A.; BOBICK, A. F.: Recognition of visual activities and interactions by stochastic parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (2000) 8, p. 852–872, DOI: 10.1109/34.868686.
- [10] PORIKLI, F.; HAGA, T.: Event Detection by Eigenvector Decomposition using Object and Frame Features. In: *Conference on Computer Vision and Pattern Recognition*, (June 27 - July 2, Washington, DC, USA), IEEE, Piscataway, NJ, USA, 2004, p. 114–114, DOI: 10.1109/CVPR.2004.335.
- [11] XIANG, T.; GONG, S.: Video Behaviour Profiling and Abnormality Detection without Manual Labelling. In: *Proceedings of the 10th IEEE International Conference on Computer Vision*, (October 17-21, Beijing, China), IEEE, Piscataway, NJ, USA, 2005, p. 1238–1245, DOI: 10.1109/ICCV.2005.248.
- [12] ZHANG, D.; GATICA-PEREZ, D.; BENGIO, S.; MCCOWAN, I.: Semi-Supervised Adapted HMMS for Unusual Event Detection. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (June 20-25, San Diego, USA), IEEE, Piscataway, NJ, USA, 2005, p. 611–618, DOI: 10.1109/CVPR.2005.316.
- [13] BASHIR, F. I.; KHOKHAR, A. A.; SCHONFELD, D.: Real-Time Motion Trajectory-Based Indexing and Retrieval of Video Sequences. *IEEE Transactions on Multimedia* 9 (2007) 1, p. 58–65, DOI: 10.1109/TMM.2006.886346.
- [14] DUONG, T. V.; BUI, H. H.; PHUNG, D. Q.; VENKATESH, S.: Activity Recognition and Abnormality Detection with the Switching Hidden Semi-Markov Model. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (June 20-25, San Diego, USA), IEEE, Piscataway, NJ, USA, 2005, p. 838–845, DOI: 10.1109/CVPR.2005.61.
- [15] CUNTOOR, N. P.; CHELLAPPA, R.: Epitomic Representation of Human Activities. In: *IEEE Conference on Computer Vision and Pattern Recognition*, (June 17-22, Minneapolis, USA), IEEE, Piscataway, NJ, USA, 2007, p. 1–8, DOI: 10.1109/CVPR.2007.383135.
- [16] JUANG, B.-H.; RABINER, L. R.: A Probabilistic Distance Measure for Hidden Markov Models. *AT&T Technical Journal* 64 (1985) 2, p. 391–408, DOI: 10.1002/j.1538-7305.1985.tb00439.x.

-
- [17] ANJUM, N.; CAVALLARO, A.: Multifeature Object Trajectory Clustering for Video Analysis. *IEEE Transactions on Circuits and Systems for Video Technology* 18 (2008) 11, p. 1555–1564, DOI: 10.1109/TCSVT.2008.2005603.
- [18] CHADIL, N.; RUSSAMEESAWANG, A.; KEERATIWINTAKORN, P.: Real-time Tracking Management System using GPS, GPRS and Google Earth. In: *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology Conference*, (May 14-17, Krabi, Thailand), IEEE, Piscataway, NJ, USA, 2008, p. 393–396, DOI: 10.1109/ECTICON.2008.4600454.
- [19] KHOURY, H. M.; KAMAT, V. R.: Evaluation of Position Tracking Technologies for User Localization in Indoor Construction Environments. *Automation in Construction* 18 (2009) 4, p. 444–457, DOI: 10.1016/j.autcon.2008.10.011.
- [20] ANDRIENKO, G.; ANDRIENKO, N.; HURTER, C.; RINZIVILLO, S.; WROBEL, S.: Scalable Analysis of Movement Data for Extracting and Exploring Significant Places. *IEEE Transactions on Visualization and Computer Graphics* 19 (2013) 7, p. 1078–1094, DOI: 10.1109/TVCG.2012.311.
- [21] SUN, Q. C.; XIA, J. C.; FOSTER, J.; FALKMER, T.; LEE, H.: Pursuing Precise Vehicle Movement Trajectory in Urban Residential Area Using Multi-GNSS RTK Tracking. *Transportation Research Procedia* 25 (2017) 1, p. 2361–2376, DOI: 10.1016/j.trpro.2017.05.255.
- [22] YUAN, J.; ZHENG, Y.; ZHANG, C.; XIE, W.; XIE, X.; SUN, G.; HUANG, Y.: T-drive: Driving Directions Based on Taxi Trajectories. In: *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, (November 3-5, San Jose, USA), ACM, New York, USA, 2010, p. 99–108, DOI: 10.1145/1869790.1869807.
- [23] GARIEL, M.; SRIVASTAVA, A. N.; FERON, E.: Trajectory Clustering and an Application to Airspace Monitoring. *IEEE Transactions on Intelligent Transportation Systems* 12 (2011) 4, p. 1511–1524, DOI: 10.1016/j.imlet.2014.04.003.
- [24] BUCHMÜLLER, J.; JANETZKO, H.; ANDRIENKO, G.; ANDRIENKO, N.; FUCHS, G.; KEIM, D. A.: Visual Analytics for Exploring Local Impact

- of Air Traffic. In: *Proceedings of the Eurographics Conference on Visualization*, (May 25-29, Cagliari, Italy), Wiley Online Library, 2015, p. 181–190, DOI: 10.1111/cgf.12630.
- [25] RODRÍGUEZ, J. P.; FERNÁNDEZ-GRACIA, J.; THUMS, M.; HINDELL, M. A.; SEQUEIRA, A. M.; MEEKAN, M. G.; COSTA, D. P.; GUINET, C.; HARCOURT, R. G.; MCMAHON, C. R.; ET AL.: Big Data Analyses Reveal Patterns and Drivers of the Movements of Southern Elephant Seals. *Scientific Reports* 7 (2017) 1, p. 1–10, DOI: 10.1038/s41598-017-00165-0.
- [26] CHENG, X. E.; DU, S. S.; LI, H. Y.; HU, J. F.; CHEN, M. L.: Obtaining Three-dimensional Trajectory of Multiple Fish in Water Tank Via Video Tracking. *Multimedia Tools and Applications* (2018), p. 1–21, DOI: 10.1007/s11042-018-5755-5.
- [27] VON HÜNERBEIN, K.; HAMANN, H.-J.; RÜTER, E.; WILTSCHKO, W.: A GPS-Based System for Recording the Flight Paths of Birds. *Naturwissenschaften* 87 (2000) 6, p. 278–279, DOI: 10.1007/s001140050721.
- [28] WEIMERSKIRCH, H.; BONADONNA, F.; BAILLEUL, F.; MABILLE, G.; DELL’OMO, G.; LIPP, H.-P.: GPS Tracking of Foraging Albatrosses. *Science* 295 (2002) 5558, p. 1259–1259, DOI: 10.1126/science.1068034.
- [29] HOU, S.; LANG, X.; WELSHER, K.: Robust Real-time 3D Single-particle Tracking using a Dynamically Moving Laser Spot. *Optics Letters* 42 (2017) 12, p. 2390–2393, DOI: 10.1364/OL.42.002390.
- [30] KALAGA, D. V.; PANT, H.; DALVI, S. V.; JOSHI, J. B.; ROY, S.: Investigation of Hydrodynamics in Bubble Column with Internals using Radioactive Particle Tracking (RPT). *AIChE Journal* 63 (2017) 11, p. 4881–4894, DOI: 10.1002/aic.15829.
- [31] WEI, X.: Accurate and Fast 3D Tracking of Dense Particle Swarms. In: *Proceedings of the 4th International Conference on Internet Multimedia Computing and Service*, (September 9-11, Wuhan, China), ACM, ACM, New York, USA, 2012, p. 180–183, DOI: 10.1109/ICPR.2004.1333863.
- [32] BUARIA, D.; YEUNG, P.: A Highly Scalable Particle Tracking Algorithm using Partitioned Global Address Space (PGAS) Programming for

- Extreme-scale Turbulence Simulations. *Computer Physics Communications* 221 (2017) 1, p. 256–258, DOI: 10.1016/j.cpc.2017.08.022.
- [33] KIM, W.; MOON, S.-W.; LEE, J.; NAM, D.-W.; JUNG, C.: Multiple Player Tracking in Soccer Videos: An Adaptive Multiscale Sampling Approach. *Multimedia Systems* (2018), p. 1–13, DOI: 10.1007/s00530-018-0586-9.
- [34] WISBEY, B.; MONTGOMERY, P. G.; PYNE, D. B.; RATTRAY, B.: Quantifying Movement Demands of AFL Football Using GPS Tracking. *Journal of Science and Medicine in Sport* 13 (2010) 5, p. 531–536, DOI: 10.1016/j.jsams.2009.09.002.
- [35] AUGHEY, R. J.: Applications of GPS Technologies to Field Sports. *International Journal of Sports Physiology and Performance* 6 (2011) 3, p. 295–310, DOI: 10.1123/ijsp.6.3.295.
- [36] STEIN, M.; JANETZKO, H.; LAMPRECHT, A.; BREITKREUTZ, T.; ZIMMERMANN, P.; GOLDLÜCKE, B.; SCHRECK, T.; ANDRIENKO, G.; GROSSNIKLAUS, M.; KEIM, D. A.: Bring it to the Pitch: Combining Video and Movement Data to Enhance Team Sport Analysis. *IEEE Transactions on Visualization and Computer Graphics* 24 (2017) 1, DOI: 10.1109/TVCG.2017.2745181.
- [37] KAMBLE, P. R.; KESKAR, A. G.; BHURCHANDI, K. M.: Ball Tracking in Sports: A Survey. *Artificial Intelligence Review* (2017), p. 1–51, DOI: 10.1007/s10462-017-9582-2.
- [38] ZHENG, Y.; LI, Q.; CHEN, Y.; XIE, X.; MA, W.-Y.: Understanding Mobility Based on GPS Data. In: *Proceedings of the 10th International Conference on Ubiquitous Computing*, (September 21-24, Seoul, Korea), ACM, New York, USA, 2008, p. 312–321, DOI: 10.1145/1409635.1409677.
- [39] DAHLBOM, A.; NIKLASSON, L.: Trajectory Clustering for Coastal Surveillance. In: *10th International Conference on Information Fusion*, (July 9-12, Quebec, Canada), IEEE, Piscataway, NJ, USA, 2007, p. 1–8, DOI: 10.1109/ICIF.2007.4408114.
- [40] VAN DER SPEK, S.; VAN SCHAICK, J.; DE BOIS, P.; DE HAAN, R.: Sensing Human Activity: GPS Tracking. *Sensors* 9 (2009) 4, p. 3033–3055, DOI: 10.3390/s90403033.

- [41] ZHENG, Y.; ZHANG, L.; XIE, X.; MA, W.-Y.: Mining Interesting Locations and Travel Sequences from GPS Trajectories. In: *Proceedings of the 18th International Conference on World Wide Web*, (April 20-24, Madrid, Spain), ACM, New York, USA, 2009, p. 791–800, DOI: 10.1145/1526709.1526816.
- [42] FOXLIN, E.: Pedestrian Tracking with Shoe-mounted Inertial Sensors. *IEEE Computer Graphics and Applications* 25 (2005) 6, p. 38–46, DOI: 10.1109/MCG.2005.140.
- [43] BUCHHEIT, M.; ALLEN, A.; POON, T. K.; MODONUTTI, M.; GREGSON, W.; DI SALVO, V.: Integrating Different Tracking Systems in Football: Multiple Camera Semi-automatic System, Local Position Measurement and GPS Technologies. *Journal of Sports Sciences* 32 (2014) 20, p. 1844–1857, DOI: 10.1080/02640414.2014.942687.
- [44] SOBKOVÁ, L.; ACHTEN, H.: Smartphone Application for Long-term Urban Lifestyle and Mobility Monitoring. In: *Architectural Research Addressing Societal Challenges*, p. 77–80, CRC Press, Boca Raton, USA, 2017.
- [45] LANE, N. D.; MILUZZO, E.; LU, H.; PEEBLES, D.; CHOUDHURY, T.; CAMPBELL, A. T.: A Survey of Mobile Phone Sensing. *IEEE Communications Magazine* 48 (2010) 9, p. 140–150, DOI: 10.1109/mcom.2010.5560598.
- [46] ZHU, X.; LI, Q.; CHEN, G.: APT: Accurate Outdoor Pedestrian Tracking with Smartphones. In: *Proceedings of the IEEE INFOCOM*, (April 14-19, Turin, Italy), IEEE, Piscataway, NJ, USA, 2013, p. 2508–2516, DOI: 10.1109/INFCOM.2013.6567057.
- [47] AHMED, M. S.; HOQUE, M. A.; KHATTAK, A. J.: Demo: Real-time Vehicle Movement Tracking on Android Devices Through Bluetooth Communication with DSRC Devices. In: *IEEE Vehicular Networking Conference*, (December 8-10, Columbus, USA), IEEE, Piscataway, NJ, USA, 2016, p. 1–2, DOI: 10.1109/VNC.2016.7835956.
- [48] OOSTERLINCK, D.; BENOIT, D. F.; BAECKE, P.; VAN DE WEGHE, N.: Bluetooth Tracking of Humans in an Indoor Environment: An Application to Shopping Mall Visits. *Applied Geography* 78 (2017) 1, p. 55–65, DOI: 10.1016/j.apgeog.2016.11.005.

- [49] ALESSANDRINI, A.; ALVAREZ, M.; GREIDANUS, H.; GAMMIERI, V.; ARGUEDAS, V. F.; MAZZARELLA, F.; SANTAMARIA, C.; STASOLLA, M.; TARCHI, D.; VESPE, M.: Mining Vessel Tracking Data for Maritime Domain Applications. In: *16th IEEE International Conference on Data Mining Workshops*, (December 12-15, Barcelona, Spain), IEEE, Piscataway, NJ, USA, 2016, p. 361–367, DOI: 10.1109/ICDMW.2016.0058.
- [50] MAZZARELLA, F.; ARGUEDAS, V. F.; VESPE, M.: Knowledge-based Vessel Position Prediction Using Historical AIS Data. In: *Sensor Data Fusion: Trends, Solutions, Applications*, (October 6-8, Bonn, Germany), IEEE, Piscataway, NJ, USA, 2015, p. 1–6, DOI: 10.1109/SDF.2015.7347707.
- [51] TETREAU, B. J.: Use of the Automatic Identification System (AIS) for Maritime Domain Awareness (MDA). In: *Proceedings of the IEEE OCEANS MTS*, (September 17-23, Washington, USA), IEEE, Piscataway, NJ, USA, 2005, p. 1590–1594, DOI: 10.1109/OCEANS.2005.1639983.
- [52] WEI, B.; NENER, B.; LIU, W.; MA, L.: Global Tracking of Space Debris via CPHD and Consensus. *Advances in Space Research* 59 (2017) 1, p. 2548–2562, DOI: 10.1016/j.asr.2017.03.002.
- [53] HIGGINBOTHAM, H.; YOKOTA, Y.; ANTON, E. S.: Strategies for Analyzing Neuronal Progenitor Development and Neuronal Migration in the Developing Cerebral Cortex. *Cerebral Cortex* 21 (2010) 7, p. 1465–1474, DOI: 10.1093/cercor/bhq197.
- [54] YUAN, Y.; LU, Y.; WANG, Q.: Tracking as a Whole: Multi-Target Tracking by Modeling Group Behavior With Sequential Detection. *IEEE Transactions on Intelligent Transportation Systems* 18 (2017) 2, p. 3339–3349, DOI: 10.1109/TITS.2017.2686871.
- [55] BLOCK, S.; ZHDANOV, V. P.; HÖÖK, F.: Quantification of Multivalent Interactions by Tracking Single Biological Nanoparticle Mobility on a Lipid Membrane. *Nano Letters* 16 (2016) 7, p. 4382–4390, DOI: 10.1021/acs.nanolett.6b01511.
- [56] CACACE, T.; PATURZO, M.; MEMMOLO, P.; VASSALLI, M.; FERRARO, P.; FRALDI, M.; MENSITIERI, G.: Digital Holography as 3D Tracking Tool

- for Assessing Acoustophoretic Particle Manipulation. *Optics Express* 25 (2017) 15, p. 17746–17752, DOI: 10.1364/OE.25.017746.
- [57] LEAL-TAIXÉ, L.; MILAN, A.; SCHINDLER, K.; CREMERS, D.; REID, I.; ROTH, S.: Tracking the Trackers: An Analysis of the State of the Art in Multiple Object Tracking. *arXiv:1704.02781* (2017).
- [58] LI, A.; ZHANG, Y.; CHEN, Z.: Scanpath Mining of Eye Movement Trajectories for Visual Attention Analysis. In: *IEEE International Conference on Multimedia and Expo*, (July 10-14, Hong Kong, China), IEEE, Piscataway, NJ, USA, 2017, p. 535–540, DOI: 10.1109/ICME.2017.8019507.
- [59] WANG, M.; ONG, L.-L. S.; DAUWELS, J.; ASADA, H. H.: Automated Tracking and Quantification of Angiogenic Vessel Formation in 3D Microfluidic Devices. *PLOS ONE* 12 (2017) 11, p. e0186465, DOI: 10.1371/journal.pone.0186465.
- [60] MEIR, R.; BETZER, O.; BARNOY, E.; MOTIEL, M.; POPOVTZER, R.: Gold Nanoparticles for Non-Invasive Cell Tracking with CT Imaging. In: *Nanoscale Imaging, Sensing, and Actuation for Biomedical Applications XV*, International Society for Optics and Photonics, 2018, p. 1050617, DOI: 10.1117/12.2287077.
- [61] WANG, X.; GAO, B.; MASNOU, S.; CHEN, L.; THEURKAUFF, I.; COTTIN-BIZONNE, C.; ZHAO, Y.; SHIH, F.: Active Colloids Segmentation and Tracking. *Pattern Recognition* 60 (2016) 1, p. 177–188, DOI: 10.1016/j.patcog.2016.04.022.
- [62] LEE, G.; MALLIPEDDI, R.; LEE, M.: Trajectory-based Vehicle Tracking at Low Frame Rates. *Expert Systems with Applications* 80 (2017) 1, p. 46–57, DOI: 10.1016/j.eswa.2017.03.023.
- [63] GOYA, Y.; CHATEAU, T.; MALATERRE, L.; TRASSOUDAIN, L.: Vehicle Trajectories Evaluation by Static Video Sensors. In: *Intelligent Transportation Systems Conference*, (September 17-20, Toronto, Canada), IEEE, Piscataway, NJ, USA, 2006, p. 864–869, DOI: 10.1109/ITSC.2006.1706852.
- [64] LIU, Y.-L.; PERILLO, E. P.; LIU, C.; YU, P.; CHOU, C.-K.; HUNG, M.-C.; DUNN, A. K.; YEH, H.-C.: Segmentation of 3D Trajectories Acquired by

- TSUNAMI Microscope: An Application to EGFR Trafficking. *Biophysical Journal* 111 (2016) 10, p. 2214–2227, DOI: 10.1016/j.bpj.2016.09.041.
- [65] FIGUEIREDO, T. C.; VIVAS, J.; PEÑA, N.; MIRANDA, J. G.: Fractal Measures of Video-recorded Trajectories can Classify Motor Subtypes in Parkinsons Disease. *Physica A: Statistical Mechanics and its Applications* 462 (2016) 1, p. 12–20, DOI: 10.1016/j.physa.2016.05.050.
- [66] ZHOU, J.; KWAN, C.: Tracking of Multiple Pixel Targets using Multiple Cameras. In: *International Symposium on Neural Networks*, Springer International AIP Publishing, 2018, p. 484–493, DOI: 10.1007/978-3-319-92537-0_56.
- [67] SPATARO, D.; ARCURI, P.; DE RANGO, A.; SPATARO, W.; D’AMBROSIO, D.; MARI, A.: A Tracking Algorithm for Particle-Like Moving Objects. In: *25th Euromicro International Conference on Parallel, Distributed and Network-based Processing*, (March 6-8, St. Petersburg, Russia), IEEE, Piscataway, NJ, USA, 2017, p. 491–496, DOI: 10.1109/PDP.2017.53.
- [68] PARKER, D.: Positron Emission Particle Tracking and its Application to Granular Media. *Review of Scientific Instruments* 88 (2017) 5, p. 051803, DOI: 10.1063/1.4983046.
- [69] CHENOUEARD, N.; SMAL, I.; DE CHAUMONT, F.; MAŠKA, M.; SBALZARINI, I. F.; GONG, Y.; CARDINALE, J.; CARTHEL, C.; CORALUPPI, S.; WINTER, M.; ET AL.: Objective Comparison of Particle Tracking Methods. *Nature Methods* 11 (2014) 3, p. 281–289, DOI: 10.1038/nmeth.2808.
- [70] GYONGY, I.; DAVIES, A.; CRESPO, A. M.; GREEN, A.; DUTTON, N. A.; DUNCAN, R. R.; RICKMAN, C.; HENDERSON, R. K.; DALGARNO, P. A.: High-Speed Particle Tracking in Microscopy using SPAD Image Sensors. In: *Proceedings of the High-Speed Biomedical Imaging and Spectroscopy III: Toward Big Data Instrumentation and Management*, (San Francisco), International Society for Optics and Photonics, 2018, p. 105050A, DOI: 10.1117/12.2290199.
- [71] CHEN, K.; GU, Y.; SUN, W.; BIN, D.; WANG, G.; FAN, X.; XIA, T.; FANG, N.: Characteristic Rotational Behaviors of Rod-shaped Cargo Revealed

- by Automated Five-dimensional Single Particle Tracking. *Nature Communications* 8 (2017) 1, p. 887, DOI: 10.1038/s41467-017-01001-9.
- [72] TUBAISHAT, M.; MADRIA, S.: Sensor Networks: An Overview. *IEEE Potentials* 22 (2003) 2, p. 20–23, DOI: 10.1109/MP.2003.1197877.
- [73] YI, T.-H.; LI, H.-N.; GU, M.: Recent Research and Applications of GPS-based Monitoring Technology for High-rise Structures. *Structural Control and Health Monitoring* 20 (2013) 5, p. 649–670, DOI: 10.1002/stc.1501.
- [74] AHMED, S. A.; DOGRA, D. P.; KAR, S.; ROY, P. P.: Extraction of Long-Duration Moving Object Trajectories from Curtailed Tracks. In: *Proceedings of 2nd International Conference on Computer Vision & Image Processing*, Springer, Singapore, 2017, p. 315–326, DOI: 10.1007/978-981-10-7898-9_26.
- [75] CHERIYADAT, A. M.; RADKE, R. J.: Detecting Dominant Motions in Dense Crowds. *IEEE Journal of Selected Topics in Signal Processing* 2 (2008) 4, p. 568–581, DOI: 10.1109/JSTSP.2008.2001306.
- [76] CHONGJING, W.; XU, Z.; YI, Z.; YUNCAI, L.: Analyzing Motion Patterns in Crowded Scenes via Automatic Tracklets Clustering. *China Communications* 10 (2013) 4, p. 144–154, DOI: 10.1109/CC.2013.6506940.
- [77] SUGIMURA, D.; KITANI, K. M.; OKABE, T.; SATO, Y.; SUGIMOTO, A.: Using Individuality to Track Individuals: Clustering Individual Trajectories in Crowds using Local Appearance and Frequency Trait. In: *12th International Conference on Computer Vision*, (September 29 - October 2, Kyoto, Japan), IEEE, Piscataway, NJ, USA, 2009, p. 1467–1474, DOI: 10.1109/ICCV.2009.5459286.
- [78] ZHOU, B.; WANG, X.; TANG, X.: Random Field Topic Model for Semantic Region Analysis in Crowded Scenes from Tracklets. In: *IEEE Conference on Computer Vision and Pattern Recognition*, (June 20-25, Providence, Rhode Island), IEEE, Piscataway, NJ, USA, 2011, p. 3441–3448, DOI: 10.1109/CVPR.2011.5995459.
- [79] LI, T.; CHANG, H.; WANG, M.; NI, B.; HONG, R.; YAN, S.: Crowded Scene Analysis: A Survey. *IEEE Transactions on Cir-*

-
- uits and Systems for Video Technology* 25 (2015) 3, p. 367–386, DOI: 10.1109/TCSVT.2014.2358029.
- [80] AHMED, S. A.; DOGRA, D. P.; KAR, S.; ROY, P. P.: Unsupervised Classification of Erroneous Video Object Trajectories. *Soft Computing* 22 (2017) 1, p. 4703–4721, DOI: 10.1007/s00500-017-2656-x.
- [81] LEWANDOWSKI, M.; SIMONNET, D.; MAKRIS, D.; VELASTIN, S. A.; ORWELL, J.: Tracklet Reidentification in Crowded Scenes Using Bag of Spatio-temporal Histograms of Oriented Gradients. In: *Pattern Recognition*, p. 94–103, Springer, Berlin, Heidelberg, Germany, 2013.
- [82] METZLER, J.; MONARI, E.; KUNTZSCH, C.: Application-driven Merging and Analysis of Person Trajectories for Distributed Smart Camera Networks. In: *IS&T/SPIE Electronic Imaging*, (San Francisco), International Society for Optics and Photonics, 2014, p. 90260I, DOI: 10.1117/12.2038964.
- [83] WANG, B.; WANG, G.; CHAN, K. L.; WANG, L.: Tracklet Association with Online Target-specific Metric Learning. In: *IEEE Conference on Computer Vision and Pattern Recognition*, (June 23–28, Columbus, USA), IEEE, Piscataway, NJ, USA, 2014, p. 1234–1241, DOI: 10.1109/CVPR.2014.161.
- [84] STEGMAIER, J.: *New Methods to Improve Large-Scale Microscopy Image Analysis with Prior Knowledge and Uncertainty*. PhD Thesis, Karlsruhe Institute of Technology, 2016.
- [85] HUISKEN, J.; STAINIER, D.: Selective Plane Illumination Microscopy Techniques in Developmental Biology. *Development* 136 (2009) 12, p. 1963–1975, DOI: 10.1242/dev.022426.
- [86] ARGANDA-CARRERAS, I.; ANDREY, P.: Designing Image Analysis Pipelines in Light Microscopy: A Rational Approach. *Light Microscopy: Methods and Protocols* 1563 (2017) 1, p. 185–207, DOI: 10.1007/978-1-4939-6810-7_13.
- [87] KOBAYASHI, S.; IWAMOTO, M.; HARAGUCHI, T.: Live Correlative Light-electron Microscopy to Observe Molecular Dynamics in High Resolution. *Microscopy* 65 (2016) 4, p. 296–308, DOI: 10.1093/jmicro/dfw024.

- [88] MANTON, J. D.; REES, E. J.: triSPIM: Light Sheet Microscopy with Isotropic Super-Resolution. *Optics Letters* 41 (2016) 18, p. 4170–4173, DOI: 10.1364/OL.41.004170.
- [89] LAVAGNINO, Z.; SANCATALDO, G.; DAMORA, M.; FOLLERT, P.; TONELLI, D. D. P.; DIASPRO, A.; ZANACCHI, F. C.: 4D (xyzt) Imaging of Thick Biological Samples by Means of Two-Photon Inverted Selective Plane Illumination Microscopy (2PE-iSPIM). *Scientific Reports* 6 (2016) 1, p. 23923, DOI: 10.1038/srep23923.
- [90] YANG, Y.; YAO, B.; LEI, M.; DAN, D.; LI, R.; VAN HORN, M.; CHEN, X.; LI, Y.; YE, T.: Two-Photon Laser Scanning Stereomicroscopy for Fast Volumetric Imaging. *PLOS ONE* 11 (2016) 12, p. e0168885, DOI: 10.1371/journal.pone.0168885.
- [91] ELISA, Z.; TOON, B.; DE SMEDT, S. C.; KATRIEN, R.; KRISTIAAN, N.; KEVIN, B.: Technical Implementations of Light Sheet Microscopy. *Microscopy Research and Technique* (2018), p. 1–18, DOI: 10.1002/jemt.22981.
- [92] KELLER, P. J.; STELZER, E. H.: Quantitative in Vivo Imaging of Entire Embryos with Digital Scanned Laser Light Sheet Fluorescence Microscopy. *Current Opinion in Neurobiology* 18 (2008) 6, p. 624–632, DOI: 10.1016/j.conb.2009.03.008.
- [93] FU, Q.; MARTIN, B. L.; MATUS, D. Q.; GAO, L.: Imaging Multicellular Specimens with Real-time Optimized Tiling Light-sheet Selective Plane Illumination Microscopy. *Nature Communications* 7 (2016) 1, p. 11088, DOI: 10.1038/ncomms11088.
- [94] KROMM, D.; THUMBERGER, T.; WITTBRODT, J.: An Eye on Light-sheet Microscopy. *Methods in Cell Biology* 133 (2016) 1, p. 105–123, DOI: 10.1016/bs.mcb.2016.01.001.
- [95] WOLFF, C.; TINEVEZ, J.-Y.; PIETZSCH, T.; STAMATAKI, E.; HARICH, B.; PREIBISCH, S.; SHORTE, S.; KELLER, P. J.; TOMANCAK, P.; PAVLOPOULOS, A.: Reconstruction of Cell Lineages and Behaviors Underlying Arthropod Limb Outgrowth with Multi-view Light-sheet Imaging and Tracking. *eLife* (2017), p. e34410, DOI: 10.7554/eLife.34410.

- [96] STEGMAIER, J.; OTTE, J. C.; KOBITSKI, A.; BARTSCHAT, A.; GARCIA, A.; NIENHAUS, G. U.; STRÄHLE, U.; MIKUT, R.: Fast Segmentation of Stained Nuclei in Terabyte-Scale, Time Resolved 3D Microscopy Image Stacks. *PLOS ONE* 9 (2014) 2, p. e90036, DOI: 10.1371/journal.pone.0090036.
- [97] REYNAUD, E. G.; PEYCHL, J.; HUISKEN, J.; TOMANCAK, P.: Guide to Light-sheet Microscopy for Adventurous Biologists. *Nature Methods* 12 (2015) 1, p. 30–34, DOI: 10.1038/nmeth.3222.
- [98] KELLER, P. J.; SCHMIDT, A. D.; WITTBRODT, J.; STELZER, E. H.: Reconstruction of Zebrafish Early Embryonic Development by Scanned Light Sheet Microscopy. *Science* 322 (2008) 5904, p. 1065–1069, DOI: 10.1126/science.1162493.
- [99] BAO, Z.; MURRAY, J. I.; BOYLE, T.; OOI, S. L.; SANDEL, M. J.; WATERSTON, R. H.: Automated Cell Lineage Tracing in *Caenorhabditis Elegans*. *Proceedings of the National Academy of Sciences of the United States of America* 103 (2006) 8, p. 2707–2712, DOI: 10.1111/j.1753-4887.1981.tb06752.x.
- [100] HUISKEN, J.; SWOGER, J.; DEL BENE, F.; WITTBRODT, J.; STELZER, E. H.: Optical Sectioning Deep Inside Live Embryos by Selective Plane Illumination Microscopy. *Science* 305 (2004) 5686, p. 1007–1009, DOI: 10.1126/science.1100035.
- [101] SCHMID, B.; SHAH, G.; SCHERF, N.; WEBER, M.; THIERBACH, K.; CAMPOS, C. P.; ROEDER, I.; AANSTAD, P.; HUISKEN, J.: High-speed Panoramic Light-sheet Microscopy Reveals Global Endodermal Cell Dynamics. *Nature Communications* 4 (2013) 1, p. 2207, DOI: 10.1038/ncomms3207.
- [102] ZHANG, J.; CAMPBELL, R. E.; TING, A. Y.; TSIEN, R. Y.: Creating new Fluorescent Probes for Cell Biology. *Nature Reviews Molecular Cell Biology* 3 (2002) 12, p. 906–918, DOI: 10.1038/nrm976.
- [103] KHMELINSKII, A.; KELLER, P. J.; BARTOSIK, A.; MEURER, M.; BARRY, J. D.; MARDIN, B. R.; KAUFMANN, A.; TRAUTMANN, S.; WACHSMUTH, M.; PEREIRA, G.; ET AL.: Tandem Fluorescent Protein Timers for in Vivo

- Analysis of Protein Dynamics. *Nature Biotechnology* 30 (2012) 7, p. 708–714, DOI: 10.1038/nbt.2281.
- [104] REISCHL, M.; BARTSCHAT, A.; LIEBEL, U.; GEHRIG, J.; MLLER, F.; MIKUT, R.: ZebrafishMiner: An Open Source Software for Interactive Evaluation of Domain-specific Fluorescence in Zebrafish. *Current Directions in Biomedical Engineering* 3 (2017) 2, p. 199–202, DOI: 10.1515/cdbme-2017-0042.
- [105] KOKEL, D.; DUNN, T.; AHRENS, M.; ALSHUT, R.; CHEUNG, C. Y.; SAINT-AMANT, L.; BRUNI, G.; MATEUS, R.; VAN HAM, T.; SHIRAKI, T.; FUKADA, Y.; KOJIMA, D.; YEH, J.-R.; MIKUT, R.; VON LINTIG, J.; ENGERT, F.; PETERSON, R.: Identification of Non-visual Photomotor Response Cells in the Vertebrate Hindbrain. *The Journal of Neuroscience* 33 (2013) 9, p. 3834–3843, DOI: 10.1523/JNEUROSCI.3689-12.201.
- [106] BATES, M.; DEMPSEY, G. T.; CHEN, K. H.; ZHUANG, X.: Multicolor Super-Resolution Fluorescence Imaging via Multi-Parameter Fluorophore Detection. *ChemPhysChem* 13 (2012) 1, p. 99–107, DOI: 10.1002/cphc.201100735.
- [107] EISSING, N.; HEGER, L.; BARANSKA, A.; CESNJEVAR, R.; BÜTTNER-HEROLD, M.; SÖDER, S.; HARTMANN, A.; HEIDKAMP, G. F.; DUDZIAK, D.: Easy Performance of 6-Color Confocal Immunofluorescence with 4-Laser Line Microscopes. *Immunology Letters* 161 (2014) 1, p. 1–5, DOI: 10.1016/j.imlet.2014.04.003.
- [108] KOBITSKI, A.; OTTE, J. C.; TAKAMIYA, M.; SCHÄFER, B.; MERTES, J.; STEGMAIER, J.; RASTEGAR, S.; RINDONE, F.; HARTMANN, V.; STOTZKA, R.; GARCÍA, A.; WEZEL, J. v.; MIKUT, R.; STRÄHLE, U.; NIENHAUS, G. U.: An Ensemble-Averaged, Cell Density-based Digital Model of Zebrafish Embryo Development Derived from Light-Sheet Microscopy Data with Single-Cell Resolution. *Scientific Reports* 5 (2015) 8601, p. 1–10, DOI: 10.1038/srep08601.
- [109] BETZIG, E.; PATTERSON, G. H.; SOUGRAT, R.; LINDWASSER, O. W.; OLENYCH, S.; BONIFACINO, J. S.; DAVIDSON, M. W.; LIPPINCOTT-SCHWARTZ, J.; HESS, H. F.: Imaging Intracellular Fluorescent Proteins

- at Nanometer Resolution. *Science* 313 (2006) 5793, p. 1642–1645, DOI: 10.1126/science.1127344.
- [110] STEGMAIER, J.; ARZ, J.; SCHOTT, B.; OTTE, J. C.; KOBITSKI, A.; NIENHAUS, G. U.; STRÄHLE, U.; SANDERS, P.; MIKUT, R.: Generating Semi-Synthetic Validation Benchmarks for Embryomics. In: *Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, (April 13-16, Prague, Czech Republic), IEEE, Piscataway, NJ, USA, 2016, p. 684–688, DOI: 10.1109/ISBI.2016.7493359.
- [111] STEGMAIER, J.; SCHOTT, B.; HÜBNER, E.; TRAUB, M.; SHAHID, M.; TAKAMIYA, M.; KOBITSKI, A.; HARTMANN, V.; STOTZKA, R.; VAN WEZEL, J.; STREIT, A.; NIENHAUS, G. U.; STRÄHLE, U.; REISCHL, M.; MIKUT, R.: Automation Strategies for Large-Scale 3D Image Analysis. *at-Automatisierungstechnik* 64 (2016) 7, p. 555–566, DOI: 10.1515/auto-2016-0019.
- [112] NACHTEGAEL, M.; VAN DER WEKEN, D.; VAN DE VILLE, D.; KERRE, E. E.: *Fuzzy Filters for Image Processing*, Vol. 122. Springer, Berlin, Heidelberg, Germany, DOI: 10.1007/978-3-540-36420-7, 2013.
- [113] SONKA, M.; HLAVAC, V.; BOYLE, R.: *Image Processing, Analysis, and Machine Vision*. Cengage Learning, Boston, UK, 2014.
- [114] ALSHUT, R.: *Konzept für Bildanalysen in Hochdurchsatz-Systemen am Beispiel des Zebrafärbings*. PhD Thesis, Karlsruher Institut für Technologie (KIT), 2016.
- [115] KHAN, A. U. M.; MIKUT, R.; REISCHL, M.: A Benchmark Data Set to Evaluate the Illumination Robustness of Image Processing Algorithms for Object Segmentation and Classification. *PLOS ONE* 10 (2015) 7, p. e0131098, DOI: 10.1371/journal.pone.0131098.
- [116] SOILLE, P.: *Morphological Image Analysis: Principles and Applications*. Springer, New York, USA, 2003.
- [117] LUENGO-OROZ, M.; LEDESMA-CARBAYO, M.; PEYRIÉRAS, N.; SANTOS, A.: Image Analysis for Understanding Embryo Development: a Bridge from Microscopy to Biological Insights. *Current Opinion in Genetics & Development* 21 (2011) 5, p. 630–637, DOI: 10.1016/j.jgde.2011.08.001.

- [118] CHEN, S.; ZHU, Y.; XIA, W.; XIA, S.; XU, X.: Automated Analysis of Zebrafish Images for Phenotypic Changes in Drug Discovery. *Journal of Neuroscience Methods* 200 (2011) 2, p. 229–236, DOI: 10.1016/j.jneumeth.2011.06.015.
- [119] MIKUT, R.; DICKMEIS, T.; DRIEVER, W.; GEURTS, P.; HAMPRECHT, F.; KAUSLER, B. X.; LEDESMA-CARBAYO, M. J.; MARÉE, R.; MIKULA, K.; PANTAZIS, P.; RONNEBERGER, O.; SANTOS, A.; STOTZKA, R.; STRÄHLE, U.; PEYRIÉRAS, N.: Automated Processing of Zebrafish Imaging Data - A Survey. *Zebrafish* 10 (2013) 3, p. 401–421, DOI: 10.1089/zeb.2013.0886.
- [120] PREIBISCH, S.; ROHLFING, T.; HASAK, M. P.; TOMANCAK, P.: Mosaicing of Single Plane Illumination Microscopy Images using Groupwise Registration and Fast Content-based Image Fusion. *Medical Imaging 2008: Image Processing* 6914 (2008) 1, p. 69140E, DOI: 10.1117/12.770893.
- [121] BROWN, M.; LOWE, D. G.: Automatic Panoramic Image Stitching using Invariant Features. *International Journal for Computer Vision* 74 (2007) 1, p. 59–73, DOI: 10.1007/s11263-006-0002-3.
- [122] PREIBISCH, S.; SAALFELD, S.; SCHINDELIN, J.; TOMANCAK, P.: Software for Bead-Based Registration of Selective Plane Illumination Microscopy Data. *Nature Methods* 7 (2010) 6, p. 418–419, DOI: 10.1038/nmeth0610-418.
- [123] UMEYAMA, S.: Least-squares Estimation of Transformation Parameters Between Two Point Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13 (1991) 4, p. 376–380, DOI: 10.1109/34.88573.
- [124] STEGMAIER, J.; AMAT, F.; LEMON, W. B.; MCDOLE, K.; WAN, Y.; TEODORO, G.; MIKUT, R.; KELLER, P. J.: Real-Time Three-Dimensional Cell Segmentation in Large-Scale Microscopy Data of Developing Embryos. *Developmental Cell* 36 (2016) 2, p. 225–240, DOI: 10.1016/j.devcel.2015.12.028.
- [125] STEGMAIER, J.; PETER, N.; PORTL, J.; MANG, I. V.; SCHRÖDER, R.; LEITTE, H.; MIKUT, R.; REISCHL, M.: A Framework for Feedback-based Segmentation of 3D Image Stacks. *Current Directions in Biomedical Engineering* 2 (2016) 1, p. 437–441, DOI: 10.1515/cdbme-2016-0097.

- [126] MIKULA, K.; REMEŠÍKOVÁ, M.; STAŠOVÁ, O.; PEYRIÉRAS, N.: Segmentation and Analysis of 3D Zebrafish Cell Image Data. In: *3rd International Congress on Image and Signal Processing*, (October 16-18, Yantai, China), IEEE, Piscataway, NJ, USA, 2010, p. 1444–1448, DOI: 10.1109/CISP.2010.5648035.
- [127] MARR, D.; HILDRETH, E.: Theory of Edge Detection. *Proceedings of the Royal Society of London. Series B. Biological Sciences* 207 (1980) 1167, p. 187–217, DOI: 10.1098/rspb.1980.0020.
- [128] LOWE, D. G.: Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60 (2004) 2, p. 91–110, DOI: 10.1023/B:VISI.0000029664.99615.94.
- [129] FENT, K.; WEISBROD, C. J.; WIRTH-HELLER, A.; PIELES, U.: Assessment of Uptake and Toxicity of Fluorescent Silica Nanoparticles in Zebrafish (*Danio rerio*) Early Life Stages. *Aquatic Toxicology* 100 (2010) 2, p. 218–228, DOI: 10.1016/j.aquatox.2010.02.019.
- [130] MOSALIGANTI, K. R.; NOCHE, R. R.; XIONG, F.; SWINBURNE, I. A.; MEGASON, S. G.: ACME: Automated Cell Morphology Extractor for Comprehensive Reconstruction of Cell Membranes. *PLoS Computational Biology* 8 (2012) 12, p. e1002780, DOI: 10.1371/journal.pcbi.1002780.
- [131] KHAN, A. U. M.; MIKUT, R.; REISCHL, M.: A New Feedback-Based Method for Parameter Adaptation in Image Processing Routines. *PloS one* 11 (2016) 10, p. e0165180, DOI: 10.1371/journal.pone.0165180.
- [132] GONZALEZ, R. C.; WOODS, R. E.: *Digital Image Processing*. Publishing House of Electronics Industry, Beijing, China, 2007.
- [133] SHARMA, H.; ALEKSEYCHUK, A.; LESKOVSKY, P.; HELLWICH, O.; ANAND, R.; ZERBE, N.; HUFNAGL, P.; ET AL.: Determining Similarity in Histological Images using Graph-Theoretic Description and Matching Methods for Content-based Image Retrieval in Medical Diagnostics. *Diagnostic Pathology* 7 (2012) 1, p. 134, DOI: 10.1186/1746-1596-7-134.
- [134] LOU, X.; KASTER, F. O.; LINDNER, M. S.; KAUSLER, B. X.; KOETHE, U.; HÖCKENDORF, B.; WITTBRODT, J.; JÄNICKE, H.; HAMPRECHT, F. A.:

- Deltr: Digital Embryo Lineage Tree Reconstructor. In: *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, (March 30 - April 2, Chicago, USA), IEEE, Piscataway, NJ, USA, 2011, p. 1557–1560, DOI: 10.1109/ISBI.2011.5872698.
- [135] ULMAN, V.; MAŠKA, M.; MAGNUSSON, K. E.; RONNEBERGER, O.; HAUBOLD, C.; HARDER, N.; MATULA, P.; MATULA, P.; SVOBODA, D.; RADOJEVIC, M.; ET AL.: An Objective Comparison of Cell-Tracking Algorithms. *Nature Methods* 14 (2017) 12, p. 1141–1152, DOI: 10.1038/nmeth.4473.
- [136] MCMAHON, A.; REEVES, G. T.; SUPATTO, W.; STATHOPOULOS, A.: Mesoderm Migration in *Drosophila* is a Multi-Step Process Requiring FGF Signaling and Integrin Activity. *Development* 137 (2010) 13, p. 2167–2175, DOI: 10.1242/dev.051573.
- [137] TOMER, R.; KHAIRY, K.; AMAT, F.; KELLER, P. J.: Quantitative High-speed Imaging of Entire Developing Embryos with Simultaneous Multi-view Light-sheet Microscopy. *Nature Methods* 9 (2012) 7, p. 755–763, DOI: 10.1038/nmeth.2062.
- [138] MEIJERING, E.; DZYUBACHYK, O.; SMAL, I.; VAN CAPPELLEN, W.: Tracking in Cell and Developmental Biology. In: *Seminars in Cell & Developmental Biology*, Elsevier, Elsevier, 2009, p. 894–902, DOI: 10.1016/j.semcdb.2009.07.004.
- [139] SCHIEGG, M.; HANSLOVSKY, P.; HAUBOLD, C.; KOETHE, U.; HUFNAGEL, L.; HAMPRECHT, F. A.: Graphical Model for Joint Segmentation and Tracking of Multiple Dividing Cells. *Bioinformatics* 31 (2014) 6, p. 948–956, DOI: 10.1093/bioinformatics/btu764.
- [140] AMAT, F.; KELLER, P. J.: Towards Comprehensive Cell Lineage Reconstructions in Complex Organisms using Light-sheet Microscopy. *Development, Growth & Differentiation* 55 (2013) 4, p. 563–578, DOI: 10.1111/dgd.12063.
- [141] AMAT, F.; LEMON, W.; MOSSING, D. P.; MCDOLE, K.; WAN, Y.; BRANSON, K.; MYERS, E. W.; KELLER, P. J.: Fast, Accurate Reconstruction

- of Cell Lineages from Large-scale Fluorescence Microscopy Data. *Nature Methods* 11 (2014) 1, p. 951–958, DOI: 10.1038/nmeth.3036.
- [142] MOUSAVI, A.: *Ontology Based Semantic Knowledge Discovery for Movement Behaviors*. PhD Thesis, University of Calgary, 2016.
- [143] RENSO, C.; BAGLIONI, M.; DE MACEDO, J. A. F.; TRASARTI, R.; WACHOWICZ, M.: How You Move Reveals Who You Are: Understanding Human Behavior by Analyzing Trajectory Data. *Knowledge and Information Systems* 37 (2013) 2, p. 331–362, DOI: 10.1007/s10115-012-0511-z.
- [144] HOU, L.; WAN, W.; HWANG, J.-N.; MUHAMMAD, R.; YANG, M.; HAN, K.: Human Tracking over Camera Networks: A Review. *EURASIP Journal on Advances in Signal Processing* 2017 (2017) 1, p. 1–20, DOI: 10.1186/s13634-017-0482-z.
- [145] JEUNG, H.; YIU, M. L.; JENSEN, C. S.: Trajectory Pattern Mining. In: *Computing with Spatial Trajectories*, p. 143–177, Springer, New York, USA, DOI: 10.1007/978-1-4614-1629-6_5, 2011.
- [146] KALAYEH, M. M.; MUSSMANN, S.; PETRAKOVA, A.; LOBO, N. D. V.; SHAH, M.: Understanding Trajectory Behavior: A Motion Pattern Approach. *arXiv* (2015).
- [147] GUDMUNDSSON, J.; LAUBE, P.; WOLLE, T.: Movement Patterns in Spatio-temporal Data. In: *Encyclopedia of GIS*, p. 726–732, Springer, Boston, USA, DOI: 10.1007/978-0-387-35973-1_823, 2008.
- [148] BRAGA, R. B.; TAHIR, A.; BERTOLOTTO, M.; MARTIN, H.: Clustering User Trajectories to Find Patterns for Social Interaction Applications. In: *Web and Wireless Geographical Information Systems*, p. 82–97, Springer, Berlin, Heidelberg, Germany, 2012.
- [149] BASIRI, A.; AMIRIAN, P.; WINSTANLEY, A.; MOORE, T.: Making Tourist Guidance Systems more Intelligent, Adaptive and Personalised using Crowd Sourced Movement Data. *Journal of Ambient Intelligence and Humanized Computing* 9 (2018), p. 1–15, DOI: 10.1007/s12652-017-0550-0.
- [150] KIHLSSTRÖM, P.: Literature Study and Assessment of Trajectory Data Mining Tools, Bachelor Thesis, KTH Royal Institute of Technology, Stockholm, 2015.

- [151] KANG, J.-Y.; YONG, H.-S.: Mining Spatio-temporal Patterns in Trajectory Data. *Journal of Information Processing Systems* 6 (2010) 4, p. 521–536, DOI: 10.3745/JIPS.2010.6.4.521.
- [152] TOOR, Z. G.; KHAN, T. I.; KHAN, O.; MASUD, J.: Comparative Analysis of Weapon Release Trajectory Using Numerical and Experimental Techniques. In: *AIAA Aerospace Sciences Meeting*, 2018, p. 1522, DOI: 10.2514/6.2018-1522.
- [153] FANGERAU, J.: *Interactive Similarity Analysis for 3D+ t Cell Trajectory Data*. PhD Thesis, Ruprecht-Karls-Universität Heidelberg, DOI: 10.11588/heidok.00018095, 2015.
- [154] DODGE, S.; WEIBEL, R.; LAUBE, P.: Exploring Movement-Similarity Analysis of Moving Objects. *SIGSPATIAL Special* 1 (2009) 3, p. 11–16, DOI: 10.1145/1645424.1645427.
- [155] ZHANG, Y.; EICK, C. F.: Novel Clustering and Analysis Techniques for Mining Spatio-Temporal Data. In: *Proceedings of the 1st ACM SIGSPATIAL PhD Workshop*, (November 4-7, Dallas, USA), ACM, New York, USA, 2014, p. 2, DOI: 10.1145/2694859.2694865.
- [156] ATEV, S.; MILLER, G.; PAPANIKOLOPOULOS, N. P.: Clustering of Vehicle Trajectories. *IEEE Transactions on Intelligent Transportation Systems* 11 (2010) 3, p. 647–657, DOI: 10.1109/TITS.2010.2048101.
- [157] GUO, L.; ZHANG, D.; CONG, G.; WU, W.; TAN, K.-L.: Influence Maximization in Trajectory Databases. *IEEE Transactions on Knowledge and Data Engineering* 29 (2017) 3, p. 627–641, DOI: 10.1109/TKDE.2016.2621038.
- [158] JEUNG, H.; LU, H.; SATHE, S.; YIU, M. L.: Managing Evolving Uncertainty in Trajectory Databases. *IEEE Transactions on Knowledge and Data Engineering* 26 (2014) 7, p. 1692–1705, DOI: 10.1109/TKDE.2013.141.
- [159] FENG, Z.; ZHU, Y.: A Survey on Trajectory Data Mining: Techniques and Applications. *IEEE Access* 4 (2016) 1, p. 2056–2067, DOI: 10.1109/ACCESS.2016.2553681.
- [160] TANUJA, V.; GOVINDARAJULU, P.: A Survey on Trajectory Data Mining. *International Journal of Computer Science and Security* 10 (2016) 5, p. 195–214.

-
- [161] MENG, F.; YUAN, G.; LV, S.; WANG, Z.; XIA, S.: An Overview on Trajectory Outlier Detection. *Artificial Intelligence Review* (2018), p. 1–20, DOI: 10.1007/s10462-018-9619-1.
- [162] SABARISH, B.; KARTHI, R.; GIREESHKUMAR, T.: Clustering of Trajectory Data Using Hierarchical Approaches. In: *Computational Vision and Bio Inspired Computing*, p. 215–226, Springer, Basel, Switzerland, 2018.
- [163] CHEN, L.; ÖZSU, M. T.; ORIA, V.: Robust and Fast Similarity Search for Moving Object Trajectories. In: *Proceedings of the ACM SIGMOD international conference on Management of data*, (June 14-16, Baltimore, Maryland, USA), ACM, New York, USA, 2005, p. 491–502.
- [164] ZHANG, Z.; HUANG, K.; TAN, T.: Comparison of Similarity Measures for Trajectory Clustering in Outdoor Surveillance Scenes. In: *18th International Conference on Pattern Recognition*, (August 20-24, Hong Kong, China), IEEE, Piscataway, NJ, USA, 2006, p. 1135–1138, DOI: 10.1109/ICPR.2006.392.
- [165] IZAKIAN, Z.; MESGARI, M. S.; ABRAHAM, A.: Automated Clustering of Trajectory Data Using a Particle Swarm Optimization. *Computers, Environment and Urban Systems* 55 (2016) 1, p. 55–65, DOI: 10.1016/j.compenvurbsys.2015.10.009.
- [166] YANAGISAWA, Y.; SATOH, T.: Clustering Multidimensional Trajectories Based on Shape and Velocity. In: *Proceedings of the 22nd International Conference on Data Engineering Workshops*, (April 3-7, Atlanta, USA), IEEE, Piscataway, NJ, USA, 2006, p. 12–12, DOI: 10.1109/ICDEW.2006.39.
- [167] VLACHOS, M.; GUNOPULOS, D.; DAS, G.: Rotation Invariant Distance Measures for Trajectories. In: *Proceedings of the tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (August 22-25, Seattle, USA), ACM, New York, USA, 2004, p. 707–712, DOI: 10.1145/1014052.1014144.
- [168] YANAGISAWA, Y.; AKAHANI, J.-I.; SATOH, T.: Shape-Based Similarity Query for Trajectory of Mobile Objects. In: *Mobile Data Management*, Springer, Berlin, Heidelberg, Germany, 2003, p. 63–77, DOI: 10.1007/3-540-36389-0_5.

- [169] ANTONINI, G.; THIRAN, J. P.: Counting Pedestrians in Video Sequences Using Trajectory Clustering. *IEEE Transactions on Circuits and Systems for Video Technology* 16 (2006) 8, p. 1008–1020, DOI: 10.1109/TCSVT.2006.879118.
- [170] FASHANDI, H.; MOGHADDAM, A. E.: A New Rotation Invariant Similarity Measure for Trajectories. In: *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, (June 27-30, Espoo, Finland), IEEE, Piscataway, NJ, USA, 2005, p. 631–634, DOI: 10.1109/CIRA.2005.1554347.
- [171] MORRIS, B.; TRIVEDI, M.: Learning Trajectory Patterns by Clustering: Experimental Studies and Comparative Evaluation. In: *IEEE Conference on Computer Vision and Pattern Recognition*, (June 20-25, Miami, USA), IEEE, Piscataway, NJ, USA, 2009, p. 312–319, DOI: 10.1109/CVPR.2009.5206559.
- [172] MAKRIS, D.; ELLIS, T.: Path Detection in Video Surveillance. *Image and Vision Computing* 20 (2002) 12, p. 895–903, DOI: 10.1016/S0262-8856(02)00098-7.
- [173] CHEN, L.; NG, R.: On the Marriage of Edit Distance and Lp Norms. In: *Proc. of 30th. International Conference on Very Large Data Bases*, (August 31 - September 3, Toronto, Ontario, Canada), 2004.
- [174] KEOGH, E.; CHAKRABARTI, K.; PAZZANI, M.; MEHROTRA, S.: Locally Adaptive Dimensionality Reduction for Indexing Large Time Series Databases. *ACM SIGMOD Record* 30 (2001) 2, p. 151–162, DOI: 10.1145/375663.375680.
- [175] KEOGH, E. J.; PAZZANI, M. J.: Scaling up Dynamic Time Warping for Datamining Applications. In: *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (August 20-23, Boston, USA), ACM, New York, USA, 2000, p. 285–289, DOI: 10.1145/347090.347153.
- [176] LITTLE, J. J.; GU, Z.: Video Retrieval by Spatial and Temporal Structure of Trajectories. In: *Photonics West -Electronic Imaging*, International Society for Optics and Photonics, 2001, p. 545–552, DOI: 10.1117/12.410966.

-
- [177] DUCHÊNE, F.; GARBAY, C.; RIALLE, V.: Similarity Measure for Heterogeneous Multivariate Time-Series. In: *12th European Conference on Signal Processing*, (September 6-10, Vienna, Austria), IEEE, Piscataway, NJ, USA, 2004, p. 1605–1608.
- [178] WEN, J.; LI, C.; XIONG, Z.: Behavior Pattern Extraction by Trajectory Analysis. *Frontiers of Computer Science in China* 5 (2011) 1, p. 37–44, DOI: 10.1007/s11704-010-0074-7.
- [179] CAI, Y.; NG, R.: Indexing Spatio-Temporal Trajectories with Chebyshev Polynomials. In: *Proceedings of the ACM SIGMOD International Conference on Management of Data*, (June 13-18, Paris, France), ACM, New York, USA, 2004, p. 599–610, DOI: 10.1145/1007568.1007636.
- [180] LEE, S.-L.; CHUN, S.-J.; KIM, D.-H.; LEE, J.-H.; CHUNG, C.-W.: Similarity Search for Multidimensional Data Sequences. In: *Proceedings of the 16th International Conference on Data Engineering*, (February 29 - March 3, San Diego, USA), IEEE, Piscataway, NJ, USA, 2000, p. 599–608, DOI: 10.1109/ICDE.2000.839473.
- [181] KEOGH, E.; CHAKRABARTI, K.; PAZZANI, M.; MEHROTRA, S.: Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases. *Knowledge and Information Systems* 3 (2001) 3, p. 263–286, DOI: 10.1007/PL00011669.
- [182] KEOGH, E.; RATANAMAHATANA, C. A.: Exact Indexing of Dynamic Time Warping. *Knowledge and Information Systems* 7 (2005) 3, p. 358–386, DOI: 10.1016/B978-155860869-6/50043-3.
- [183] PARK, S.; CHU, W. W.; YOON, J.; HSU, C.: Efficient Searches for Similar Subsequences of Different Lengths in Sequence Databases. In: *Proceedings of the 16th International Conference on Data Engineering*, (February 29, March 3, San Diego, USA), IEEE, Piscataway, NJ, USA, 2000, p. 23–32, DOI: 10.1109/ICDE.2000.839384.
- [184] VLACHOS, M.; KOLLIOS, G.; GUNOPULOS, D.: Discovering Similar Multidimensional Trajectories. In: *18th International Conference on Data Engineering*, (February 26 - March 1, San Jose, USA), IEEE, Piscataway, NJ, USA, 2002, p. 673–684, DOI: 10.1109/ICDE.2002.994784.

- [185] DODGE, S.; WEIBEL, R.; LAUTENSCHÜTZ, A.-K.: Towards a Taxonomy of Movement Patterns. *Information Visualization* 7 (2008) 3-4, p. 240–252, DOI: 10.1057/palgrave.ivs.9500182.
- [186] ZAKI, M. H.; SAYED, T.: Automated Analysis of Pedestrian Group Behavior in Urban Settings. *IEEE Transactions on Intelligent Transportation Systems* 19 (2017) 6, p. 1880–1889, DOI: 10.1109/TITS.2017.2747516.
- [187] LI, Y.; BAILEY, J.; KULIK, L.: Efficient Mining of Platoon Patterns in Trajectory Databases. *Data & Knowledge Engineering* 100 (2015) 1, p. 167–187, DOI: 10.1016/j.datak.2015.02.001.
- [188] WANG, Y.; LIM, E.-P.; HWANG, S.-Y.: Efficient Mining of Group Patterns from User Movement Data. *Data & Knowledge Engineering* 57 (2006) 3, p. 240–282, DOI: 10.1016/j.datak.2005.04.006.
- [189] GUDMUNDSSON, J.; VAN KREVELD, M.: Computing Longest Duration Flocks in Trajectory Data. In: *Proceedings of the 14th Annual ACM International Symposium on Advances in Geographic Information Systems*, (November 10-11, Arlington, USA), ACM, New York, USA, 2006, p. 35–42, DOI: 10.1145/1183471.1183479.
- [190] JEUNG, H.; YIU, M. L.; ZHOU, X.; JENSEN, C. S.; SHEN, H. T.: Discovery of Convoys in Trajectory Databases. *Proceedings of the VLDB Endowment* 1 (2008) 1, p. 1068–1080, DOI: 10.14778/1453856.1453971.
- [191] KALNIS, P.; MAMOULIS, N.; BAKIRAS, S.: On Discovering Moving Clusters in Spatio-temporal Data. In: *Advances in Spatial and Temporal Databases*, p. 364–381, Springer, Berlin, Heidelberg, Germany, DOI: 10.1007/11535331_2, 2005.
- [192] ANDERSSON, M.; GUDMUNDSSON, J.; LAUBE, P.; WOLLE, T.: Reporting Leaders and Followers among Trajectories of Moving Point Objects. *GeoInformatica* 12 (2008) 4, p. 497–528, DOI: 10.1007/s10707-007-0037-9.
- [193] WU, F.; LEI, T. K. H.; LI, Z.; HAN, J.: MoveMine 2.0: Mining Object Relationships from Movement Data. *Proceedings of the VLDB Endowment* 7 (2014) 13, p. 1613–1616, DOI: 10.14778/2733004.2733043.

-
- [194] LI, Z.; JI, M.; LEE, J.-G.; TANG, L.-A.; YU, Y.; HAN, J.; KAYS, R.: MoveMine: Mining Moving Object Databases. In: *Proceedings of the ACM SIGMOD International Conference on Management of Data*, (June 6-10, Indianapolis, USA), ACM, New York, USA, 2010, p. 1203–1206, DOI: 10.1145/1807167.1807319.
- [195] WACHOWICZ, M.; ONG, R.; RENSO, C.; NANNI, M.: Finding Moving Flock Patterns among Pedestrians through Collective Coherence. *International Journal of Geographical Information Science* 25 (2011) 11, p. 1849–1864, DOI: 10.1080/13658816.2011.561209.
- [196] LI, Z.; DING, B.; WU, F.; LEI, T. K. H.; KAYS, R.; CROFOOT, M. C.: Attraction and Avoidance Detection from Movements. *Proceedings of the VLDB Endowment* 7 (2013) 3, p. 157–168, DOI: 10.14778/2732232.2732235.
- [197] MIKUT, R.: *Data Mining in der Medizin und Medizintechnik*. Universitätsverlag Karlsruhe, Karlsruhe, Germany, 2008.
- [198] BUZAN, D.; SCLAROFF, S.; KOLLIOS, G.: Extraction and Clustering of Motion Trajectories in Video. In: *Proceedings of the 17th International Conference on Pattern Recognition*, (August 26, Cambridge, UK), IEEE, Piscataway, NJ, USA, 2004, p. 521–524, DOI: 10.1109/ICPR.2004.1334287.
- [199] ANJUM, N.; CAVALLARO, A.: Unsupervised Fuzzy Clustering for Trajectory Analysis. In: *IEEE International Conference on Image Processing*, IEEE, Piscataway, NJ, USA, 2007, p. III–213, DOI: 10.1109/ICIP.2007.4379284.
- [200] MORRIS, B. T.; TRIVEDI, M. M.: Learning, Modeling, and Classification of Vehicle Track Patterns from Live Video. *IEEE Transactions on Intelligent Transportation Systems* 9 (2008) 3, p. 425–437, DOI: 10.1109/TITS.2008.922970.
- [201] BILIOTTI, D.; ANTONINI, G.; THIRAN, J. P.: Multi-Layer Hierarchical Clustering of Pedestrian Trajectories for Automatic Counting of People in Video Sequences. In: *Proceedings of the IEEE Workshop on Motion and Video Computing*, (January 5-7, Breckenridge, USA), IEEE, Piscataway, NJ, USA, 2005, p. 50–57, DOI: 10.1109/ACVMOT.2005.77.
- [202] LIAO, T. W.: Clustering of Time Series Data: A Survey. *Pattern Recognition* 38 (2005) 11, p. 1857–1874, DOI: 10.1016/j.patcog.2005.01.025.

- [203] DUDA, R. O.; HART, P. E.; STORK, D. G.: *Pattern Classification*. John Wiley & Sons, New York, USA, 2012.
- [204] YI, B.-K.; FALOUTSOS, C.: Fast Time Sequence Indexing for Arbitrary Lp Norms. In: *26th International Conference on Very Large Data Bases, VLDB*, Morgan Kaufmann, San Francisco, USA, 2000, p. 385–39.
- [205] MOEHRMANN, J.; HEIDEMANN, G.: Automatic Trajectory Clustering for Generating Ground Truth Data Sets. In: *Proceedings of Image Processing: Machine Vision Applications*, (January 28, San Jose, USA), International Society for Optics and Photonics, 2010, p. 753808, DOI: 10.1117/12.838954.
- [206] PELEKIS, N.; KOPANAKIS, I.; MARKETOS, G.; NTOUTSI, I.; ANDRIENKO, G.; THEODORIDIS, Y.: Similarity Search in Trajectory Databases. In: *14th International Symposium on Temporal Representation and Reasoning*, (June 28-30, Alicante, Spain), IEEE, Piscataway, NJ, USA, 2007, p. 129–140, DOI: 10.1109/TIME.2007.59.
- [207] KISILEVICH, S.; MANSMANN, F.; NANNI, M.; RINZIVILLO, S.: *Spatio-Temporal Clustering*. Springer, New York, USA, DOI: 10.1007/978-0-387-09823-4_44, 2010.
- [208] CHEN, S. S.; GOPALAKRISHNAN, P. S.: Clustering via the Bayesian Information Criterion with Applications in Speech Recognition. In: *IEEE International Conference on Acoustics, Speech and Signal Processing*, (May 15, Seattle, USA), IEEE, Piscataway, NJ, USA, 1998, p. 645–648, DOI: 10.1109/ICASSP.1998.675347.
- [209] FU, Z.; HU, W.; TAN, T.: Similarity Based Vehicle Trajectory Clustering and Anomaly Detection. In: *IEEE International Conference on Image Processing*, (September 14, Genova, Italy), IEEE, Piscataway, NJ, USA, 2005, p. II–602, DOI: 10.1109/ICIP.2005.1530127.
- [210] LEE, J.-G.; HAN, J.; LI, X.; GONZALEZ, H.: TraClass: Trajectory Classification using Hierarchical Region-based and Trajectory-based Clustering. *Proceedings of the VLDB Endowment* 1 (2008) 1, p. 1081–1094, DOI: 10.14778/1453856.1453972.
- [211] ZELNIK-MANOR, L.; IRANI, M.: Event-Based Analysis of Video. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and*

-
- Pattern Recognition*, (December 8-14, Kauai, USA), IEEE, Piscataway, NJ, USA, 2001, p. 123–130, DOI: 10.1109/CVPR.2001.990935.
- [212] FRENTZOS, E.; GRATSIAS, K.; THEODORIDIS, Y.: Index-Based Most Similar Trajectory Search. In: *IEEE 23rd International Conference on Data Engineering*, (April 15-20, Istanbul, Turkey), IEEE, Piscataway, NJ, USA, 2007, p. 816–825, DOI: 10.1109/ICDE.2007.367927.
- [213] LEE, J.-G.; HAN, J.; WHANG, K.-Y.: Trajectory Clustering: A Partition-and-group Framework. In: *Proceedings of the ACM SIGMOD International Conference on Management of Data*, (June 11-14, Beijing, China), ACM, New York, USA, 2007, p. 593–604, DOI: 10.1145/1247480.1247546.
- [214] HWANG, S.-Y.; LIU, Y.-H.; CHIU, J.-K.; LIM, E.-P.: Mining Mobile Group Patterns: A Trajectory-Based Approach. In: *Advances in Knowledge Discovery and Data Mining*, p. 713–718, Springer, Berlin, Heidelberg, Germany, DOI: 10.1007/11430919_82, 2005.
- [215] LI, Y.; HAN, J.; YANG, J.: Clustering Moving Objects. In: *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (August 22-25, Seattle, WA, USA), ACM, New York, USA, 2004, p. 617–622, DOI: 10.1145/1014052.1014129.
- [216] RINZIVILLO, S.; PEDRESCHI, D.; NANNI, M.; GIANNOTTI, F.; ANDRIENKO, N.; ANDRIENKO, G.: Visually Driven Analysis of Movement Data by Progressive Clustering. *Information Visualization* 7 (2008) 3-4, p. 225–239, DOI: 10.1057/palgrave.ivs.9500183.
- [217] ANDRIENKO, G.; ANDRIENKO, N.; RINZIVILLO, S.; NANNI, M.; PEDRESCHI, D.; GIANNOTTI, F.: Interactive Visual Clustering of Large Collections of Trajectories. In: *IEEE Symposium on Visual Analytics Science and Technology*, (October 12-13, Atlantic City, USA), IEEE, Piscataway, NJ, USA, 2009, p. 3–10, DOI: 10.1109/VAST.2009.5332584.
- [218] PELEKIS, N.; KOPANAKIS, I.; KOTSIFAKOS, E. E.; FRENTZOS, E.; THEODORIDIS, Y.: Clustering Trajectories of Moving Objects in an Uncertain World. In: *IEEE International Conference on Data Mining*, (December 6-9, Miami, USA), IEEE, Piscataway, NJ, USA, 2009, p. 417–427, DOI: 10.1109/ICDM.2009.57.

- [219] ZHOU, H.; KIMBER, D.: Unusual Event Detection via Multi-camera Video Mining. In: *18th International Conference on Pattern Recognition*, (August 20-24, Hong Kong, China), IEEE, Piscataway, NJ, USA, 2006, p. 1161–1166, DOI: 10.1109/ICPR.2006.1149.
- [220] GRIMSON, W. E. L.; STAUFFER, C.; ROMANO, R.; LEE, L.: Using Adaptive Tracking to Classify and Monitor Activities in a Site. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (June 25, Santa Barbara, USA), IEEE, Piscataway, NJ, USA, 1998, p. 22–29, DOI: 10.1109/CVPR.1998.698583.
- [221] SYEDA-MAHMOOD, T.; WANG, F.: Unsupervised Clustering using Multi-Resolution Perceptual Grouping. In: *IEEE Conference on Computer Vision and Pattern Recognition*, (June 17-22, Minneapolis, USA), IEEE, Piscataway, NJ, USA, 2007, p. 1–8, DOI: 10.1109/CVPR.2007.382986.
- [222] ANKERST, M.; BREUNIG, M. M.; KRIEGEL, H.-P.; SANDER, J.: OPTICS: Ordering Points to Identify the Clustering Structure. In: *Proceedings of the ACM SIGMOD International Conference on Management and Data*, (May 31 - June 3, New York, USA), ACM, New York, USA, 1999, p. 49–60, DOI: 10.1145/304181.304187.
- [223] NANNI, M.; PEDRESCHI, D.: Time-Focused Clustering of Trajectories of Moving Objects. *Journal of Intelligent Information Systems* 27 (2006) 3, p. 267–289, DOI: 10.1007/s10844-006-9953-7.
- [224] ZHONG, H.; SHI, J.; VISONTAI, M.: Detecting Unusual Activity in Video. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (June 27 - July 2, Washington, USA), IEEE, Piscataway, NJ, USA, 2004, p. II-819, DOI: 10.1109/CVPR.2004.1315249.
- [225] JUNEJO, I. N.; JAVED, O.; SHAH, M.: Multi Feature Path Modeling for Video Surveillance. In: *Proceedings of the 17th International Conference on Pattern Recognition*, (August 26, Cambridge, UK), IEEE, Piscataway, NJ, USA, 2004, p. 716–719, DOI: 10.1109/ICPR.2004.1334359.
- [226] GAO, X.; YU, F.: Trajectory Clustering using a New Distance Based on Minimum Convex Hull. In: *17th World Congress of International Fuzzy Systems Association*, (June 27-30, Otsu, Japan), IEEE, Piscataway, NJ, USA, 2017, p. 1–6, DOI: 10.1109/IFSA-SCIS.2017.8023255.

-
- [227] KUMAR, D.; BEZDEK, J. C.; RAJASEGARAR, S.; LECKIE, C.; PALANISWAMI, M.: A Visual-numeric Approach to Clustering and Anomaly Detection for Trajectory Data. *The Visual Computer* 33 (2015) 3, p. 265–281, DOI: 10.1007/s00371-015-1192-x.
- [228] LI, W.; MAHADEVAN, V.; VASCONCELOS, N.: Anomaly Detection and Localization in Crowded Scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36 (2014) 1, p. 18–32, DOI: 10.1109/TPAMI.2013.111.
- [229] LU, W.; WEI, X.; XING, W.; LIU, W.: Trajectory-based Motion Pattern Analysis of Crowds. *Neurocomputing* 247 (2017) 1, p. 231–223, DOI: 10.1016/j.neucom.2017.03.074.
- [230] MEHRAN, R.; OYAMA, A.; SHAH, M.: Abnormal Crowd Behavior Detection using Social Force Model. In: *IEEE Conference on Computer Vision and Pattern Recognition*, (June 20-25, Miami, USA), IEEE, Piscataway, NJ, USA, 2009, p. 935–942, DOI: 10.1109/CVPR.2009.5206641.
- [231] PICIARELLI, C.; MICHELONI, C.; FORESTI, G.: Anomalous Trajectory Patterns Detection. In: *19th International Conference on Pattern Recognition*, (December 8-11, Tampa, USA), IEEE, Piscataway, NJ, USA, 2008, p. 1–4, DOI: 10.1109/ICPR.2008.4761422.
- [232] KNORR, E. M.; NG, R. T.; TUCAKOV, V.: Distance-based Outliers: Algorithms and Applications. *The VLDB Journal - The International Journal on Very Large Data Bases* 8 (2000) 3-4, p. 237–253, DOI: 10.1007/s007780050006.
- [233] LEE, J.-G.; HAN, J.; LI, X.: Trajectory Outlier Detection: A Partition-and-detect Framework. In: *24th IEEE International Conference on Data Engineering*, (April 7-12, Cancun, Mexico), IEEE, Piscataway, NJ, USA, 2008, p. 140–149, DOI: 10.1109/ICDE.2008.4497422.
- [234] SCHABENBERGER, O.; GOTWAY, C. A.: *Statistical Methods for Spatial Data Analysis*. CRC Press, London, UK, 2017.
- [235] PAPADIMITRIOU, S.; KITAGAWA, H.; GIBBONS, P. B.; FALOUTSOS, C.: Loci: Fast Outlier Detection using the Local Correlation Integral. In: *19th International Conference on Data Engineering*, (March 5-8,

- Bangalore, India), IEEE, Piscataway, NJ, USA, 2003, p. 315–326, DOI: 10.1109/ICDE.2003.1260802.
- [236] BREUNIG, M. M.; KRIEGEL, H.-P.; NG, R. T.; SANDER, J.: LOF: Identifying Density-based Local Outliers. In: *Proceedings of the ACM SIGMOD International Conference on Management of Data*, (May 15-18, Dallas, USA), ACM, New York, USA, 2000, p. 93–104, DOI: 10.1145/342009.335388.
- [237] AGGARWAL, C. C.; YU, P. S.: Outlier Detection for High Dimensional Data. In: *Proceedings of the ACM SIGMOD International Conference on Management of Data*, (May 21-24, Santa Barbara, USA), ACM, New York, USA, 2001, p. 37–46, DOI: 10.1145/376284.375668.
- [238] BEYAN, C.: Fish Behavior Analysis. In: *Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data*, p. 161–179, Springer, Basel, Switzerland, DOI: 10.1007/978-3-319-30208-9_12, 2016.
- [239] YU, Y.; CAO, L.; RUNDENSTEINER, E. A.; WANG, Q.: Detecting Moving Object Outliers in Massive-scale Trajectory Streams. In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (August 24-27, New York, USA), ACM, New York, USA, 2014, p. 422–431, DOI: 10.1145/2623330.2623735.
- [240] DA SILVA, T. L. C.; ZEITOUNI, K.; DE MACÊDO, J. A.; CASANOVA, M. A.: A Framework for Online Mobility Pattern Discovery from Trajectory Data Streams. In: *17th IEEE International Conference on Mobile Data Management*, (June 13-16, Porto, Portugal), IEEE, Piscataway, NJ, USA, 2016, p. 365–368, DOI: 10.1109/MDM.2016.65.
- [241] XU, H.; ZHOU, Y.; LIN, W.; ZHA, H.: Unsupervised Trajectory Clustering via Adaptive Multi-Kernel-Based Shrinkage. In: *Proceedings of the IEEE International Conference on Computer Vision*, (December 7-13, Washington, USA), IEEE, Piscataway, NJ, USA, 2015, p. 4328–4336, DOI: 10.1109/ICCV.2015.492.
- [242] PAN, X.; HE, Y.; WANG, H.; XIONG, W.; PENG, X.: Mining Regular Behaviors Based on Multidimensional Trajectories. *Expert Systems with Applications* 66 (2016) 1, p. 106–113, DOI: 10.1016/j.eswa.2016.09.015.

- [243] ALVARES, L. O.; BOGORNY, V.; KUIJPERS, B.; DE MACEDO, J. A. F.; MOELANS, B.; VAISMAN, A.: A Model for Enriching Trajectories with Semantic Geographical Information. In: *Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information Systems*, (November 7-9, Seattle, USA), ACM, New York, USA, 2007, p. 22, DOI: 10.1145/1341012.1341041.
- [244] MAKRIS, D.; ELLIS, T.: Learning Semantic Scene Models from Observing Activity in Visual Surveillance. *IEEE Transactions on Systems, Man, and Cybernetics* 35 (2005) 3, p. 397–408, DOI: 10.1109/TSMCB.2005.846652.
- [245] WANG, X.; TIEU, K.; GRIMSON, E.: Learning Semantic Scene Models by Trajectory Analysis. In: *Computer Vision–ECCV 2006*, p. 110–123, Springer, Berlin, Heidelberg, Germany, DOI: 10.1007/11744078_9, 2006.
- [246] MAGNUSSON, K. E.; JALDÉN, J.; GILBERT, P. M.; BLAU, H. M.: Global Linking of Cell Tracks using the Viterbi Algorithm. *IEEE Transactions on Medical Imaging* 34 (2015) 4, p. 911–929, DOI: 10.1109/TMI.2014.2370951.
- [247] VIJVERBERG, J.; KOELEMAN, C. J.; DE WITH, P.; ET AL.: Towards Real-time and Low-latency Video Object Tracking by Linking Tracklets of Incomplete Detections. In: *10th IEEE International Conference on Advanced Video and Signal Based Surveillance*, (August 27-30, Krakow, Poland), IEEE, Piscataway, NJ, USA, 2013, p. 300–305, DOI: 10.1109/AVSS.2013.6636656.
- [248] WU, Z.; KUNZ, T. H.; BETKE, M.: Efficient Track Linking Methods for Track Graphs using Network-flow and Set-cover Techniques. In: *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Piscataway, NJ, USA, 2011, p. 1185–1192, DOI: 10.1109/CVPR.2011.5995515.
- [249] YAO, D.; ZHANG, C.; ZHU, Z.; HU, Q.; WANG, Z.; HUANG, J.; BI, J.: Learning Deep Representation for Trajectory Clustering. *Expert Systems* 35 (2018) 2, p. e12252, DOI: 10.1111/exsy.12252.
- [250] MACIAG, P. S.: A Survey on Data Mining Methods for Clustering Complex Spatiotemporal Data. In: *International Conference: Beyond Databases, Architectures and Structures*, (May 30 - June 2, Ustron, Poland), Springer, Berlin, Heidelberg, Germany, 2017, p. 115–126, DOI: 10.1007/978-3-319-58274-0_10.

- [251] YUAN, G.; SUN, P.; ZHAO, J.; LI, D.; WANG, C.: A Review of Moving Object Trajectory Clustering Algorithms. *Artificial Intelligence Review* 47 (2017) 1, p. 123–144, DOI: 10.1007/s10462-016-9477-7.
- [252] MOHAMMED, N.; FUNG, B.; DEBBABI, M.: Walking in the Crowd: Anonymizing Trajectory Data for Pattern Analysis. In: *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, (November 2-6, Hong Kong, China), ACM, New York, USA, 2009, p. 1441–1444, DOI: 10.1145/1645953.1646140.
- [253] AKASAPU, A. K.; RAO, P. S.; SHARMA, L.; SATPATHY, S.: Density Based K-Nearest Neighbors Clustering Algorithm for Trajectory Data. *International Journal of Advanced Science and Technology* 31 (2011) 1, p. 47–58.
- [254] ALELYANI, S.; TANG, J.; LIU, H.: Feature Selection for Clustering: A Review. *Data Clustering: Algorithms and Applications* 29 (2013) 1, p. 29–60.
- [255] BERKHIN, P.: A Survey of Clustering Data Mining Techniques. In: *Grouping Multidimensional Data*, p. 25–71, Springer, Berlin, Heidelberg, Germany, DOI: 10.1007/3-540-28349-8_2, 2006.
- [256] SUZUKI, N.; HIRASAWA, K.; TANAKA, K.; KOBAYASHI, Y.; SATO, Y.; FUJINO, Y.: Learning Motion Patterns and Anomaly Detection by Human Trajectory Analysis. In: *IEEE International Conference on Systems, Man and Cybernetics*, (October 7-10, Montreal, Canada), IEEE, Piscataway, NJ, USA, 2007, p. 498–503, DOI: 10.1109/ICSMC.2007.4413596.
- [257] DOBRKOVIC, A.; IACOB, M.-E.; VAN HILLEGERSBERG, J.: Maritime Pattern Extraction and Route Reconstruction From Incomplete AIS Data. *International Journal of Data Science and Analytics* 5 (2018) 2, p. 111–136, DOI: 10.1007/s41060-017-0092-8.
- [258] DELL, A. I.; BENDER, J. A.; BRANSON, K.; COUZIN, I. D.; DE POLAVIEJA, G. G.; NOLDUS, L. P.; PÉREZ-ESCUADERO, A.; PERONA, P.; STRAW, A. D.; WIKELSKI, M.; ET AL.: Automated Image-based Tracking and its Application in Ecology. *Trends in Ecology & Evolution* 29 (2014) 7, p. 417–428, DOI: 10.1016/j.tree.2014.05.004.

-
- [259] FRADET, M.; ROBERT, P.; PÉREZ, P.: Clustering Point Trajectories with Various Life-Spans. In: *Conference for Visual Media Production*, (November 12-13, London, UK), IEEE, Piscataway, NJ, USA, 2009, p. 7–14, DOI: 10.1109/CVMP.2009.24.
- [260] KRIEGEL, H.-P.; KRÖGER, P.; ZIMEK, A.: Clustering High-dimensional Data: A Survey on Subspace Clustering, Pattern-based Clustering, and Correlation Clustering. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 3 (2009) 1, p. 1, DOI: 10.1145/1497577.1497578.
- [261] TRIPATHI, P. K.; ET AL.: *Techniques for Spatio-temporal Analysis of Trajectory Data*. PhD Thesis, University of Texas Arlington, 2016.
- [262] HUANG, Q.; LI, Z.; LI, J.; CHANG, C.: Mining Frequent Trajectory Patterns from Online Footprints. In: *Proceedings of the 7th ACM SIGSPATIAL International Workshop on GeoStreaming*, (October 31, Burlingame, USA), ACM, New York, USA, 2016, p. 10, DOI: 10.1145/3003421.3003431.
- [263] BROSTOW, G. J.; CIPOLLA, R.: Unsupervised Bayesian Detection of Independent Motion in Crowds. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (June 17-22, New York, USA), IEEE, Piscataway, NJ, USA, 2006, p. 594–601, DOI: 10.1109/CVPR.2006.320.
- [264] KEIM, D.; KOHLHAMMER, J.; ELLIS, G.; MANSMANN, F.: *Mastering the Information Age Solving Problems with Visual Analytics*. Eurographics Association, Goslar, Germany, 2010.
- [265] KASEMSAP, K.: Knowledge Discovery and Data Visualization: Theories and Perspectives. *International Journal of Organizational and Collective Intelligence* 7 (2017) 3, p. 56–69, DOI: 10.4018/IJOCI.2017070105.
- [266] NANDHINI, K.; SHANTHI, I. E.: Analysis of Mining, Visual Analytics Tools and Techniques in Space and Time. In: *Proceedings of the Second International Conference on Computer and Communication Technologies*, (July 24-26, Hyderabad, India), Springer, Berlin, Heidelberg, Germany, 2016, p. 547–556, DOI: 10.1007/978-81-322-2523-2_53.
- [267] CHEN, C.: *Information Visualization: Beyond the Horizon*. Springer, London, UK, DOI: 10.1007/1-84628-579-8, 2006.

- [268] CENEDA, D.; GSCHWANDTNER, T.; MAY, T.; MIKSCH, S.; SCHULZ, H.-J.; STREIT, M.; TOMINSKI, C.: Characterizing Guidance in Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics* 23 (2017) 1, p. 111–120, DOI: 10.1109/TVCG.2016.2598468.
- [269] BARRETT, J. C.; FRY, B.; MALLER, J.; DALY, M. J.: Haploview: Analysis and Visualization of LD and Haplotype Maps. *Bioinformatics* 21 (2004) 2, p. 263–265, DOI: 10.1093/bioinformatics/bth457.
- [270] ANDRIENKO, G.; ANDRIENKO, N.: Interactive Cluster Analysis of Diverse Types of Spatiotemporal Data. *ACM SIGKDD Explorations Newsletter* 11 (2010) 2, p. 19–28, DOI: 10.1145/1809400.1809405.
- [271] ENGUEHARD, R. A.; HOEBER, O.; DEVILLERS, R.: Interactive Exploration of Movement Data: A Case Study of Geovisual Analytics for Fishing Vessel Analysis. *Information Visualization* 12 (2012) 1, p. 65–84, DOI: 10.1177/1473871612456121.
- [272] DEMŠAR, U.; BUCHIN, K.; CAGNACCI, F.; SAFI, K.; SPECKMANN, B.; VAN DE WEGHE, N.; WEISKOPF, D.; WEIBEL, R.: Analysis and Visualisation of Movement: An Interdisciplinary Review. *Movement Ecology* 3 (2015) 1, p. 5, DOI:10.1186/s40462-015-0032-y.
- [273] GUO, H.; WANG, Z.; YU, B.; ZHAO, H.; YUAN, X.: Tripvista: Triple Perspective Visual Trajectory Analytics and its Application on Microscopic Traffic Data at a Road Intersection. In: *IEEE Pacific Visualization Symposium*, (March 1-4, Hong Kong, China), IEEE, Piscataway, NJ, USA, 2011, p. 163–170, DOI: 10.1109/PACIFICVIS.2011.5742386.
- [274] ANDRIENKO, N.; ANDRIENKO, G.: Designing Visual Analytics Methods for Massive Collections of Movement Data. *Cartographica: The International Journal for Geographic Information and Geovisualization* 42 (2007) 2, p. 117–138, DOI: 10.3138/carto.42.2.117.
- [275] CHEN, C.; YU, Y.: Empirical Studies of Information Visualization: A Meta-analysis. *International Journal of Human-Computer Studies* 53 (2000) 5, p. 851–866, DOI: 10.1006/ijhc.2000.0422.

- [276] DOS SANTOS, S.; BRODLIE, K.: Gaining Understanding of Multivariate and Multidimensional Data Through Visualization. *Computers & Graphics* 28 (2004) 3, p. 311–325, DOI: 10.1016/j.cag.2004.03.013.
- [277] ANDRIENKO, G.; ANDRIENKO, N.; FUCHS, G.; WOOD, J.: Revealing Patterns and Trends of Mass Mobility Through Spatial and Temporal Abstraction of Origin-destination Movement Data. *IEEE Transactions on Visualization and Computer Graphics* 23 (2017) 9, p. 2120–2136, DOI: 10.1109/TVCG.2016.2616404.
- [278] BAK, P.; OMER, I.; SCHRECK, T.: Visual Analytics of Urban Environments using High-resolution Geographic Data. *Geospatial Thinking* 1 (2010), p. 25–42, DOI: 10.1007/978-3-642-12326-9_2.
- [279] WANG, Z.; YE, T.; LU, M.; YUAN, X.; QU, H.; YUAN, J.; WU, Q.: Visual Exploration of Sparse Traffic Trajectory Data. *IEEE Transactions on Visualization and Computer Graphics* 20 (2014) 12, p. 1813–1822, DOI: 10.1109/TVCG.2014.2346746.
- [280] PALLOTTA, G.; VESPE, M.; BRYAN, K.: Vessel Pattern Knowledge Discovery from AIS Data: A Framework for Anomaly Detection and Route Prediction. *Entropy* 15 (2013) 6, p. 2218–2245, DOI: 10.3390/e15062218.
- [281] GUIDOTTI, R.; MONREALE, A.; RINZIVILLO, S.; PEDRESCHI, D.; GIANNOTTI, F.: Retrieving Points of Interest from Human Systematic Movements. In: *Software Engineering and Formal Methods*, p. 294–308, Springer, Basel, Switzerland, DOI: /10.1007/978-3-319-15201-1_19, 2014.
- [282] ANDRIENKO, G.; ANDRIENKO, N.; WROBEL, S.: Visual Analytics Tools for Analysis of Movement Data. *ACM SIGKDD Explorations Newsletter* 9 (2007) 2, p. 38–46, DOI: 10.1145/1345448.1345455.
- [283] FAURE, E.; SAVY, T.; RIZZI, B.; MELANI, C.; STAŠOVÁ, O.; FABRÈGES, D.; ŠPIR, R.; HAMMONS, M.; ČÚNDERLÍK, R.; RECHER, G.: A Workflow to Process 3D+ Time Microscopy Images of Developing Organisms and Reconstruct their Cell Lineage. *Nature Communications* 7 (2016), p. 8674, DOI: 10.1038/ncomms9674.
- [284] CHEN, X.; KORDY, P.; LU, R.; PANG, J.: MinUS: Mining User Similarity with Trajectory Patterns. In: *Joint European Conference on Machine Learning*

- and Knowledge Discovery in Databases*, Springer, Berlin, Heidelberg, Germany, 2014, p. 436–439, DOI: 10.1007/978-3-662-44845-8_29.
- [285] HAI, P. N.; PONCELET, P.; TEISSEIRE, M.: Get.move: An Efficient and Unifying Spatio-temporal Pattern Mining Algorithm for Moving Objects. In: *International Symposium on Intelligent Data Analysis*, Springer, Berlin, Heidelberg, Germany, 2012, p. 276–288, DOI: 10.1007/978-3-642-34156-4_26.
- [286] GIANNOTTI, F.; NANNI, M.; PEDRESCHI, D.; PINELLI, F.; RENSO, C.; RINZIVILLO, S.; TRASARTI, R.: Unveiling the Complexity of Human Mobility by Querying and Mining Massive Trajectory Data. *The International Journal on Very Large Data Bases* 20 (2011) 5, p. 695–719, DOI: 10.1007/s00778-011-0244-8.
- [287] WANG, G.; MALIK, A.; YAU, C.; SURAKITBANHARN, C.; EBERT, D. S.: TraSeer: A Visual Analytics Tool for Vessel Movements in the Coastal Areas. In: *IEEE International Symposium on Technologies for Homeland Security*, (April 25-26, Waltham, USA), IEEE, Piscataway, NJ, USA, 2017, p. 1–6, DOI: 10.1109/THS.2017.7943473.
- [288] MIKUT, R.; BARTSCHAT, A.; DONEIT, W.; GONZÁLEZ ORDIANO, J. Á.; SCHOTT, B.; STEGMAIER, J.; WACZOWICZ, S.; REISCHL, M.: The MATLAB Toolbox SciXMiner: User’s Manual and Programmer’s Guide. Tech. Rep., arXiv:1704.03298, 2017.
- [289] MIERSWA, I.; WURST, M.; KLINKENBERG, R.; SCHOLZ, M.; EULER, T.: YALE: Rapid Prototyping for Complex Data Mining Tasks. In: *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (UNGAR, L.; CRAVEN, M.; GUNOPULOS, D.; ELIASRAD, T., Ed.), (August 20-23, Philadelphia, USA), ACM, New York, USA, 2006, p. 935–940, DOI: 10.1145/1150402.1150531.
- [290] BERTHOLD, M. R.; CEBRON, N.; DILL, F.; GABRIEL, T. R.; KÖTTER, T.; MEINL, T.; OHL, P.; SIEB, C.; THIEL, K.; WISWEDEL, B.: KNIME: The Konstanz Information Miner. In: *Data Analysis, Machine Learning and Applications*, p. 319–326, Springer, Berlin, Heidelberg, Germany, 2008.

- [291] FRANK, E.; HALL, M.; HOLMES, G.; KIRKBY, R.; PFAHRINGER, B.; WITTEN, I.: Weka: A Machine Learning Workbench for Data Mining. In: *Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers*, p. 1305–1314, Springer, New York, USA, DOI: 10.1007/978-0-387-09823-4_66, 2005.
- [292] TINEVEZ, J.-Y.; PERRY, N.; SCHINDELIN, J.; HOOPES, G. M.; REYNOLDS, G. D.; LAPLANTINE, E.; BEDNAREK, S. Y.; SHORTE, S. L.; ELICEIRI, K. W.: TrackMate: An Open and Extensible Platform for Single-Particle Tracking. *Methods* 115 (2016) 1, p. 80–90, DOI: 10.1016/j.ymeth.2016.09.016.
- [293] PICCININI, F.; KISS, A.; HORVATH, P.: CellTracker (not only) for Dummies. *Bioinformatics* 32 (2015) 6, p. 955–957, DOI: 10.1093/bioinformatics/btv686.
- [294] SCHINDELIN, J.; ARGANDA-CARRERAS, I.; FRISE, E.; KAYNIG, V.; LONGAIR, M.; PIETZSCH, T.; PREIBISCH, S.; RUEDEN, C.; SAALFELD, S.; SCHMID, B.; TINEVEZ, J.-Y.; WHITE, D. J.; HARTENSTEIN, V.; ELICEIRI, K.; TOMANCAK, P.; CARDONA, A.: Fiji: An Open-Source Platform for Biological-Image Analysis. *Nature Methods* 9 (2012) 7, p. 676–682, DOI: 10.1038/nmeth.2019.
- [295] BARTSCHAT, A.; HÜBNER, E.; REISCHL, M.; MIKUT, R.; STEGMAIER, J.: XPIWIT - An XML Pipeline Wrapper for the Insight Toolkit. *Bioinformatics* 32 (2016) 2, p. 315–317, DOI: 10.1093/bioinformatics/btv559.
- [296] STEGMAIER, J.; ALSHUT, R.; REISCHL, M.; MIKUT, R.: Information Fusion of Image Analysis, Video Object Tracking, and Data Mining of Biological Images using the Open Source MATLAB Toolbox Gait-CAD. *Biomedizinische Technik (Biomedical Engineering)* 57 (2012) SI-1, p. 458–461, DOI: 10.1515/bmt-2012-4073.
- [297] WOLFF, C.; TINEVEZ, J.-Y.; PIETZSCH, T.; STAMATAKI, E.; HARICH, B.; GUIGNARD, L.; PREIBISCH, S.; SHORTE, S.; KELLER, P. J.; TOMANCAK, P.; ET AL.: Multi-view Light-sheet Imaging and Tracking with the MaMuT Software Reveals the Cell Lineage of a Direct Developing Arthropod Limb. *eLife* 7 (2018) 1, p. e34410, DOI: 10.7554/eLife.34410.

- [298] PIETZSCH, T.; SAALFELD, S.; PREIBISCH, S.; TOMANČAK, P.: Big-DataViewer: Visualization and Processing for Large Image Data Sets. *Nature Methods* 12 (2015) 6, p. 481–483, DOI: 10.1038/nmeth.3392.
- [299] CEDILNIK, A.; GEVECI, B.; MORELAND, K.; AHRENS, J.; FAVRE, J.: Remote Large Data Visualization in the ParaView Framework. In: *Proceedings of the 6th Eurographics conference on Parallel Graphics and Visualization*, (May 11-12, Braga, Portugal), ACM, New York, USA, 2006, p. 163–170, DOI: 10.2312/EGPGV/EGPGV06/163-170.
- [300] SAALFELD, S.; CARDONA, A.; HARTENSTEIN, V.; TOMANČÁK, P.: CAT-MAID: Collaborative Annotation Toolkit for Massive Amounts of Image Data. *Bioinformatics* 25 (2009) 15, p. 1984–1986, DOI: 10.1093/bioinformatics/btp266.
- [301] LI, Y.; CAO, X.; CHEN, S.; LIU, J.: Visual Multi-object Tracking via Bi-level Association Strategy Within Air-Traffic-Control Surveillance Videos. In: *Integrated Communications Navigation and Surveillance*, (April 19-21, Herndon, USA), IEEE, Piscataway, NJ, USA, 2016, p. 7B2–1, DOI: 10.1109/ICNSURV.2016.7486370.
- [302] BRANSON, K.; ROBIE, A. A.; BENDER, J.; PERONA, P.; DICKINSON, M. H.: High-throughput Ethomics in Large Groups of *Drosophila*. *Nature Methods* 6 (2009) 6, p. 451–457, DOI: 10.1038/nmeth.1328.
- [303] STRAW, A. D.; BRANSON, K.; NEUMANN, T. R.; DICKINSON, M. H.: Multi-camera Real-time Three-dimensional Tracking of Multiple Flying Animals. *Journal of The Royal Society Interface* 1 (2010) 1, p. rsif20100230, DOI: 10.1098/rsif.2010.0230.
- [304] HAO, T.; WANG, Q.; WU, D.; SUN, J.-S.: Multiple Person Tracking Based on Slow Feature Analysis. *Multimedia Tools and Applications* 77 (2017) 3, p. 3623–3637, DOI: 10.1007/s11042-017-5218-4.
- [305] BRANSON, K.; BELONGIE, S.: Tracking Multiple Mouse Contours (without too many samples). In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (June 20-25, San Diego, USA), IEEE, Piscataway, NJ, USA, 2005, p. 1039–1046, DOI: 10.1109/CVPR.2005.349.

-
- [306] HSIEH, J.-W.; YU, S.-H.; CHEN, Y.-S.; HU, W.-F.: Automatic Traffic Surveillance System for Vehicle Tracking and Classification. *IEEE Transactions on Intelligent Transportation Systems* 7 (2006) 2, p. 175–187, DOI: 10.1109/TITS.2006.874722.
- [307] IWASE, S.; SAITO, H.: Tracking Soccer Player Using Multiple Views. In: *Proceedings of the ITE Annual Convention*, 2003, p. 102–105, DOI: 10.11485/itetaikai.2003s.0.87.0.
- [308] JIANG, X.; CAO, X.: Surveillance From Above: A Detection-and-prediction Based Multiple Target Tracking Method on Aerial Videos. In: *Integrated Communications Navigation and Surveillance*, (April 19-21, Hersondon, USA), IEEE, Piscataway, NJ, USA, 2016, p. 4D2–1, DOI: 10.1109/IC-NSURV.2016.7486348.
- [309] MAGEE, D. R.: Tracking Multiple Vehicles using Foreground, Background and Motion Models. *Image and Vision Computing* 22 (2004) 2, p. 143–155, DOI: 10.1016/S0262-8856(03)00145-8.
- [310] QIAN, Z.-M.; WANG, S. H.; CHENG, X. E.; CHEN, Y. Q.: An Effective and Robust Method for Tracking Multiple Fish in Video Image Based on Fish Head Detection. *BMC Bioinformatics* 17 (2016) 1, p. 251, DOI: 10.1186/s12859-016-1138-y.
- [311] CHEEZUM, M. K.; WALKER, W. F.; GUILFORD, W. H.: Quantitative Comparison of Algorithms for Tracking Single Fluorescent Particles. *Biophysical Journal* 81 (2001) 4, p. 2378–2388, DOI: 10.1016/S0006-3495(01)75884-5.
- [312] FUJISAKI, K.; HAMANO, A.; AOKI, K.; FENG, Y.; UCHIDA, S.; ARASEKI, M.; SAITO, Y.; SUZUKI, T.: Detection and Tracking Protein Molecules in Fluorescence Microscopic Video. In: *First International Symposium on Computing and Networking*, (December 4-6, Matsuyama, Japan), IEEE, Piscataway, NJ, USA, 2013, p. 270–274, DOI: 10.1109/CANDAR.2013.47.
- [313] FÉLIX-FÉLIX, J.; SALINAS-TAPIA, H.; BAUTISTA-CAPETILLO, C.; GARCÍA-ARAGÓN, J.; BURGUETE, J.; PLAYÁN, E.: A Modified Particle Tracking Velocimetry Technique to Characterize Sprinkler Irrigation Drops. *Irrigation Science* 35 (2017) 6, p. 515–531, DOI: 10.1007/s00271-017-0556-6.

- [314] KAUCIC, R.; BROOKSBY, G.; KAUFHOLD, J.; HOOGS, A.; ET AL.: A Unified Framework for Tracking through Occlusions and Cross Sensor Gaps. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (June 20-25, San Diego, USA), IEEE, Piscataway, NJ, USA, 2005, p. 990–997, DOI: 10.1109/CVPR.2005.53.
- [315] BETKE, M.; HIRSH, D. E.; BAGCHI, A.; HRISTOV, N. I.; MAKRIS, N. C.; KUNZ, T. H.: Tracking Large Variable Numbers of Objects in Clutter. In: *IEEE Conference on Computer Vision and Pattern Recognition*, (June 17-22, Minneapolis, USA), IEEE, Piscataway, NJ, USA, 2007, p. 1–8, DOI: 10.1109/CVPR.2007.382994.
- [316] MERAD, D.; IGUERNAISSI, R.; AZIZ, K.-E.; FERTIL, B.; DRAP, P.: Tracking Multiple Persons under Partial and Global Occlusions: Application to Customers Behavior Analysis. *Pattern Recognition Letters* 81 (2016) 1, p. 11–20, DOI: 10.1016/j.patrec.2016.04.011.
- [317] KUMAR, P.; RANGANATH, S.; SENGUPTA, K.; WEIMIN, H.: Cooperative Multitarget Tracking with Efficient Split and Merge Handling. *IEEE Transactions on Circuits and Systems for Video Technology* 16 (2006) 12, p. 1477–1490, DOI: 10.1109/TCSVT.2006.885715.
- [318] GENOVESIO, A.; OLIVO-MARIN, J.-C.: Split and Merge Data Association Filter for Dense Multi-target Tracking. In: *Proceedings of the 17th International Conference on Pattern Recognition*, (August 26, Cambridge, UK), IEEE, Piscataway, NJ, USA, 2004, p. 677–680, DOI: 10.1109/ICPR.2004.1333863.
- [319] KUMAR, P.; BROOKS, M. J.; DICK, A.: Adaptive Multiple Object Tracking using Colour and Segmentation Cues. In: *Asian Conference on Computer Vision*, (November 18-22, Tokyo, Japan), Springer, Berlin, Heidelberg, Germany, 2007, p. 853–863, DOI: 10.1007/978-3-540-76386-4.81.
- [320] STEPHAN, T.; GRINBERG, M.: Probabilistic Handling of Merged Detections in Multi Target Tracking. In: *IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*, (September 18-21, Beijing, China), IEEE, Piscataway, NJ, USA, 2012, p. 355–361, DOI: 10.1109/AVSS.2012.56.

-
- [321] ARDEKANI, R.; BIYANI, A.; DALTON, J. E.; SALTZ, J. B.; ARBEITMAN, M. N.; TOWER, J.; NUZH DIN, S.; TAVARÉ, S.: Three-dimensional Tracking and Behaviour Monitoring of Multiple Fruit Flies. *Journal of The Royal Society Interface* 10 (2013) 78, DOI: 10.1098/rsif.2012.0547.
- [322] STEIN, S. C.; THIART, J.: TrackNTrace: A Simple and Extendable Open-source Framework for Developing Single-molecule Localization and Tracking Algorithms. *Scientific Reports* 6 (2016) 1, p. 37947, DOI: 10.1038/srep37947.
- [323] XING, J.; AI, H.; LAO, S.: Multi-object Tracking through Occlusions by local Tracklets Filtering and Global Tracklets Association with Detection Responses. In: *IEEE Conference on Computer Vision and Pattern Recognition*, (June 20-25, Miami, USA), IEEE, Piscataway, NJ, USA, 2009, p. 1200–1207, DOI: 10.1109/CVPR.2009.5206745.
- [324] CHEN, J.; CAI, Y.; WEI, C.; YANG, L.; ALBER, M. S.; CHEN, D. Z.: Segmentation and Tracking of *Pseudomonas Aeruginosa* for Cell Dynamics Analysis in Time-lapse Images. In: *IEEE 13th International Symposium on Biomedical Imaging*, (April 13-16, Prague, Czech Republic), IEEE, Piscataway, NJ, USA, 2016, p. 968–971, DOI: 10.1109/ISBI.2016.7493426.
- [325] SRINIVAS, C.; HOOGS, A.; BROOKSBY, G.; HU, W.; ET AL.: Multi-object Tracking through Simultaneous Long Occlusions and Split-merge Conditions. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (June 17-22, New York, USA), IEEE, Piscataway, NJ, USA, 2006, p. 666–673, DOI: 10.1109/CVPR.2006.195.
- [326] TAO, J.; FRANKE, U.; KLETTE, R.: Context-based Multi-target Tracking with Occlusion Handling. *Machine Vision and Applications* 27 (2016) 8, p. 1339–1349, DOI: 10.1007/s00138-016-0770-x.
- [327] KHAN, Z.; BALCH, T.; DELLAERT, F.: Multitarget Tracking with Split and Merged Measurements. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (June 20-25, San Diego, USA), IEEE, Piscataway, NJ, USA, 2005, p. 605–610, DOI: 10.1109/CVPR.2005.245.
- [328] SONG, X.; NEVATIA, R.: Robust Vehicle Blob Tracking with Split/Merge Handling. In: *International Evaluation Workshop on Classification of Events*,

- Activities and Relationships*, (April 6-7, Southampton, UK), Springer, Berlin, Heidelberg, Germany, 2006, p. 216–222, DOI: 10.1007/978-3-540-69568-4_18.
- [329] BOSE, B.; WANG, X.; GRIMSON, E.: Multi-class Object Tracking Algorithm that Handles Fragmentation and Grouping. In: *IEEE Conference on Computer Vision and Pattern Recognition*, (June 17-22, Minneapolis, USA), IEEE, Piscataway, NJ, USA, 2007, p. 1–8, DOI: 10.1109/CVPR.2007.383175.
- [330] PATEL, M.; LEGGETT, S. E.; LANDAUER, A. K.; WONG, I. Y.; FRANCK, C.: Rapid, Topology-Based Particle Tracking for High-Resolution Measurements of Large Complex 3D Motion Fields. *Scientific Reports* 8 (2018) 1, p. 5581, DOI: 10.1038/s41598-018-23488-y.
- [331] LI, Y.; ROSE, F.; DI PIETRO, F.; MORIN, X.; GENOVESIO, A.: Detection and Tracking of Overlapping Cell Nuclei for Large Scale Mitosis Analyses. *BMC Bioinformatics* 17 (2016) 1, p. 1–15, DOI: 10.1186/s12859-016-1030-9.
- [332] THIRUSITTAMPALAM, K.: *Cellular Tracking and Mitosis Detection in Dense In-vitro Cellular Data*. PhD Thesis, Dublin City University, 2012.
- [333] CARTER, B. C.; SHUBEITA, G. T.; GROSS, S. P.: Tracking Single Particles: A User-friendly Quantitative Evaluation. *Physical Biology* 2 (2005) 1, p. 60, DOI: 10.1088/1478-3967/2/1/008.
- [334] KHAN, Z.; BALCH, T.; DELLAERT, F.: MCMC-based Particle Filtering for Tracking a Variable Number of Interacting Targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (2005) 11, p. 1805–1819, DOI: 10.1109/TPAMI.2005.223.
- [335] MILLER, N.; GERLAI, R.: Automated Tracking of Zebrafish Shoals and the Analysis of Shoaling Behavior. In: *Zebrafish Protocols for Neurobehavioral Research*, p. 217–230, Springer, Luxemburg, DOI: 10.1007/978-1-61779-597-8_16, 2012.
- [336] CHENG, X. E.; QIAN, Z.-M.; WANG, S. H.; JIANG, N.; GUO, A.; CHEN, Y. Q.: A Novel Method for Tracking Individuals of Fruit Fly Swarms

- Flying in a Laboratory Flight Arena. *PLOS ONE* 10 (2015) 6, p. e0129657, DOI: 10.1371/journal.pone.0129657.
- [337] KOHLHOFF, K. J.; JAHN, T. R.; LOMAS, D. A.; DOBSON, C. M.; CROWTHER, D. C.; VENDRUSCOLO, M.: The iFly Tracking System for an Automated Locomotor and Behavioural Analysis of *Drosophila Melanogaster*. *Integrative Biology* 3 (2011) 7, p. 755–760, DOI: 10.1039/C0IB00149J.
- [338] SADEGHIAN, A.; ALAHI, A.; SAVARESE, S.: Tracking The Untrackable: Learning To Track Multiple Cues with Long-Term Dependencies. *IEEE International Conference on Computer Vision* (2017), p. 300–311, DOI: 10.1109/ICCV.2017.41.
- [339] YANG, F.; VENKATARAMAN, C.; STYLES, V.; KUTTENBERGER, V.; HORN, E.; VON GUTTENBERG, Z.; MADZVAMUSE, A.: A Computational Framework for Particle and Whole Cell Tracking Applied to a Real Biological Dataset. *Journal of Biomechanics* 49 (2016) 8, p. 1290–1304, DOI: 10.1016/j.jbiomech.2016.02.008.
- [340] HILSENBECK, O.; SCHWARZFISCHER, M.; SKYLAKI, S.; SCHAUBERGER, B.; HOPPE, P. S.; LOEFFLER, D.; KOKKALIARIS, K. D.; HASTREITER, S.; SKYLAKI, E.; FILIPCZYK, A.; ET AL.: Software Tools for Single-cell Tracking and Quantification of Cellular and Molecular Properties. *Nature Biotechnology* 34 (2016) 7, p. 703–706, DOI: 10.1038/nbt.3626.
- [341] BISE, R.; YIN, Z.; KANADE, T.: Reliable Cell Tracking by Global Data Association. In: *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, (March 30 - April 2, Chicago, USA), IEEE, Piscataway, NJ, USA, 2011, p. 1004–1010, DOI: 10.1109/ISBI.2011.5872571.
- [342] CUI, Y.; ZHANG, Y.; JIA, P.; WANG, Y.; HUANG, J.; CUI, J.; LAI, W. T.: Three-Dimensional Particle Tracking Velocimetry Algorithm Based on Tetrahedron Vote. *Experiments in Fluids* 59 (2018) 2, p. 31, DOI: 10.1007/s00348-017-2485-9.
- [343] LIN, M.-Y.: Deterministic Particle Tracking Simulation of Pollutant Discharges in Rivers and Estuaries. *Environmental Fluid Mechanics* (2018), p. 1–24, DOI: 10.1007/s10652-018-9590-z.

- [344] SKYLAKI, S.; HILSENBECK, O.; SCHROEDER, T.: Challenges in Long-term Imaging and Quantification of Single-cell Dynamics. *Nature Biotechnology* 34 (2016) 11, p. 1137–1144, DOI: 10.1038/nbt.3713.
- [345] VALLOTTON, P.; VAN OIJEN, A. M.; WHITCHURCH, C. B.; GELFAND, V.; YEO, L.; TSIIVALIARIS, G.; HEINRICH, S.; DULTZ, E.; WEIS, K.; GRÜNWARD, D.: Diatrack Particle Tracking Software: Review of Applications and Performance Evaluation. *Traffic* 18 (2017) 12, p. 840–852, DOI: 10.1111/tra.12530.
- [346] SCHOTT, B.; STEGMAIER, J.; TAKAMIYA, M.; MIKUT, R.: Challenges of Integrating A Priori Information Efficiently in the Discovery of Spatio-Temporal Objects in Large Databases. In: *Proceedings. 25. Workshop Computational Intelligence*, (November 26-27, Dortmund, Germany), 2015, p. 1–22.
- [347] KELLER, P. J.: Imaging Morphogenesis: Technological Advances and Biological Insights. *Science* 340 (2013) 6137, p. 1234168, DOI: 10.1126/science.1234168.
- [348] SCHOTT, B.; TRAUB, M.; SCHLAGENHAUF, C.; TAKAMIYA, M.; ANTRITTER, T.; BARTSCHAT, A.; LÖFFLER, K.; BLESSING, D.; OTTE, J.; KOBITSKI, A.; NIENHAUS, G.; STRÄHLE, U.; MIKUT, R.; STEGMAIER, J.: EmbryoMiner: A New Framework for Interactive Knowledge Discovery in Large-scale Cell Tracking Data of Developing Embryos. *PLoS Computational Biology* 14 (2018) 4, p. e1006128–e1006128, DOI: 10.1371/journal.pcbi.1006128.
- [349] TRAUB, M.: *Design, Implementation and Evaluation of a Visual Interface to Guide the Knowledge Discovery Process within Large Biological Datasets*. Masters Thesis, Karlsruhe Institute of Technology, 2016.
- [350] MAŠKA, M.; ULMAN, V.; SVOBODA, D.; MATULA, P.; MATULA, P.; EDERRA, C.; URBIOLA, A.; ESPAÑA, T.; VENKATESAN, S.; BALAK, D. M.; ET AL.: A Benchmark for Comparison of Cell Tracking Algorithms. *Bioinformatics* 30 (2014) 11, p. 1609–1617, DOI: 10.1093/bioinformatics/btu080.
- [351] KANKAANPÄÄ, P.; PAAVOLAINEN, L.; TIITTA, S.; KARJALAINEN, M.; PÄIVÄRINNE, J.; NIEMINEN, J.; MARJOMÄKI, V.; HEINO, J.; WHITE,

- D.: BioImageXD: An Open, General-Purpose and High-Throughput Image-Processing Platform. *Nature Methods* 9 (2012) 7, p. 683–689, DOI: 10.1038/nmeth.2047.
- [352] ALBAUM, G.: The Likert Scale Revisited. *Market Research Society Journal* 39 (1997) 2, p. 1–21, DOI: 10.1177/147078539703900202.
- [353] ÖZPOLAT, B. D.; HANDBERG-THORSAGER, M.; VERVOORT, M.; BAL-
AVOINE, G.: Cell Lineage and Cell Cycling Analyses of the 4D Micromere
Using Live Imaging in the Marine Annelid *Platynereis Dumerilii*. *eLife* 6
(2017), p. e30463, DOI: 10.7554/eLife.30463.
- [354] WHITEHEAD, L. W.; MCARTHUR, K.; GEOGHEGAN, N. D.; ROGERS,
K. L.: The Reinvention of Twentieth Century Microscopy for Three-
dimensional Imaging. *Immunology and Cell Biology* 95 (2017) 1, p. 520–524,
DOI: 10.1038/icb.2017.36.
- [355] STEGMAIER, J.; MIKUT, R.: Fuzzy-based Propagation of Prior Knowl-
edge to Improve Large-scale Image Analysis Pipelines. *PLoS ONE* 12
(2017) 11, p. e0187535, DOI: 10.1371/journal.pone.0187535.
- [356] WOLPERT, L.; TICKLE, C.; ARIAS, A. M.: *Principles of Development*. Ox-
ford University Press, Oxford, UK, 2015.
- [357] PASTOR-ESCUREDO, D.; LOMBADOT, B.; SAVY, T.; BOYREAU, A.;
GOICOLEA, J. M.; SANTOS, A.; BOURGINE, P.; DEL ALAMO, J. C.;
PEYRIERAS, N.; CARBAYO, M. J. L.: Kinematic Analysis of Cell Lineage
Reveals Coherent and Robust Mechanical Deformation Patterns in Ze-
brafish Gastrulation. *bioRxiv* (2016), DOI: 10.1101/054353.
- [358] TAKAMIYA, M.; STEGMAIER, J.; KOBITSKI, A.; SCHOTT, B.; WEGER,
B. D.; MARGARITI, D.; CERECEDA-DELGADO, A. R.; GOURAIN, V.;
SCHERR, T.; YANG, L.; SORGE, S.; OTTE, J. C.; SCHÄFER, B.; HART-
MANN, V.; VAN WEZEL, J.; STOTZKA, R.; REINHARD, T.; SCHLUNCK, G.;
DICKMEIS, T.; RASTEGAR, S.; MIKUT, R.; NIENHAUS, G. U.; STRÄHLE,
U.: Two Distinct Waves of Neural Crest Cells Pattern the Proximodistal
Axis of the Eye in a pax6 Dependent Manner (2018), In Preparation.

- [359] SAXENA, A.; PENG, B. N.; BRONNER, M. E.: Sox10-Dependent Neural Crest Origin of Olfactory Microvillous Neurons in Zebrafish. *eLife* 2 (2013) 1, p. e00336, DOI: 10.7554/eLife.00336.
- [360] THEVENEAU, E.; STEVENTON, B.; SCARPA, E.; GARCIA, S.; TREPAT, X.; STREIT, A.; MAYOR, R.: Chase-and-Run Between Adjacent Cell Populations Promotes Directional Collective Migration. *Nature Cell Biology* 15 (2013) 7, p. 763–772, DOI: 10.1038/ncb2772.
- [361] RONNEBERGER, O.; FISCHER, P.; BROX, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, Berlin, Heidelberg, Germany, 2015, p. 234–241, DOI: 10.1007/978-3-319-24574-4_28.
- [362] ÇIÇEK, Ö.; ABDULKADIR, A.; LIENKAMP, S. S.; BROX, T.; RONNEBERGER, O.: 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer International Publishing, 2016, p. 424–432, DOI: 10.1007/978-3-319-46723-8_49.
- [363] SADANANDAN, S. K.; RANEFALL, P.; LE GUYADER, S.; WÄHLBY, C.: Automated Training of Deep Convolutional Neural Networks for Cell Segmentation. *Scientific Reports* 7 (2017) 1, p. 7860, DOI: 10.1038/s41598-017-07599-6.
- [364] CAICEDO, J. C.; ROTH, J.; GOODMAN, A.; BECKER, T.; KARHOHS, K. W.; MCQUIN, C.; SINGH, S.; THEIS, F.; CARPENTER, A. E.: Evaluation of Deep Learning Strategies for Nucleus Segmentation in Fluorescence Images. *bioRxiv* (2018), p. 335216, DOI: 10.1101/335216.
- [365] MAIER-HEIN, L.; EISENMANN, M.; REINKE, A.; ONOGUR, S.; STANKOVIC, M.; SCHOLZ, P.; ARBEL, T.; BOGUNOVIC, H.; BRADLEY, A. P.; CARASS, A.; ET AL.: Is the Winner Really the Best? A Critical Analysis of Common Research Practice in Biomedical Image Analysis Competitions, arXiv:1806.02051 (2018).
- [366] HE, T.; MAO, H.; GUO, J.; YI, Z.: Cell Tracking using Deep Neural Networks with Multi-task Learning. *Image and Vision Computing* 60 (2017) 1, p. 142–153, DOI: 10.1016/j.imavis.2016.11.010.

- [367] LIU, K.; QIAO, H.; WU, J.; WANG, H.; FANG, L.; DAI, Q.: Fast 3D Cell Tracking with Wide-Field Fluorescence Microscopy Through Deep Learning. *arXiv:1805.05139* (2018).