

# Visualization of Vessel Traffic

## PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van de rector magnificus, prof.dr.ir. C.J. van Duijn, voor een commissie aangewezen door het College voor Promoties in het openbaar te verdedigen op woensdag 21 december 2011 om 16.00 uur

door

Cornelis Martinus Eugenius Willems

geboren te Moergestel

Dit proefschrift is goedgekeurd door de promotor:

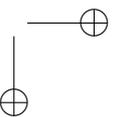
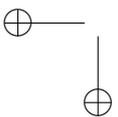
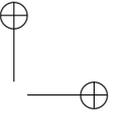
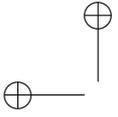
prof.dr.ir. J.J. van Wijk

Copromotor:

dr.ir. H.M.M. van de Wetering

A catalogue record is available from the Eindhoven University of Technology Library

ISBN: 978-90-386-2960-5



Promotor:

prof.dr.ir. J.J. van Wijk (Technische Universiteit Eindhoven)

Copromotor:

dr.ir. H.M.M. van de Wetering (Technische Universiteit Eindhoven)

Kerncommissie:

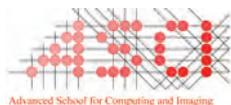
prof.dr. M.J. Kraak (Universiteit Twente)

dr. G. Andrienko (Fraunhofer Institute IAIS)

prof.dr.ir. B. de Vries (Technische Universiteit Eindhoven)



This work has been carried out as a part of the Poseidon project at Thales Nederland under the responsibilities of the Embedded Systems Institute (ESI). This project is partially supported by the Dutch Ministry of Economic Affairs under the BSIK program.



This work was carried out in the ASCI graduate school (Advanced School for Computing and Imaging). ASCI dissertation series number 235.



Published by Technische Universiteit Eindhoven

Printed by Gildeprint Drukkerijen, Enschede

The cover shows a sculpture of a vessel density map of the Netherlands in the harbour of Amsterdam. The concept is inspired by Miska Knapek's (<http://www.knapek.org>) project "Fabrication/InfoVis Physical". Photography by Natalia Voynarovskaya, milling of the sculpture by Marcel Roberscheuten (<http://www.RoboCNC.nl>), and cover design by Niels Willems.

(c) 2011 C.M.E. Willems. All rights reserved. Reproduction in whole or in part is allowed only with the written consent of the copyright owner.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Movement is Everywhere	2
1.2	Vessel Traffic	2
1.3	Visualization for Vessel Traffic	3
1.4	Trajectories	4
1.5	Research Question	6
1.6	Research Goals	6
1.7	Overview	7
1.8	Related Publications	9
1.8.1	Primary Papers	9
1.8.2	Secondary Papers	9
<b>2</b>	<b>Background</b>	<b>11</b>
2.1	Moving Object Analysis	12
2.2	Data Model	13
2.2.1	Continuous Trajectories	13
2.2.2	Trajectory Reconstruction	14
2.3	Characteristics	15
2.4	Analysis Tasks	17
2.4.1	Elementary Tasks	17
2.4.2	Synoptic Tasks	18
2.5	Movement Visualizations	18
2.5.1	Object Movement Features	19
2.5.2	Aggregated Movement Features	21
2.6	Vessel Traffic	23
2.6.1	Navigation Principles	23
2.6.2	Tracking of Vessel Traffic	26
2.6.3	Trajectory Characteristics for Vessel Traffic	27
2.6.4	Tasks for Vessel Traffic	28

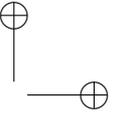
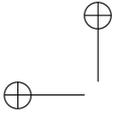
<b>3</b>	<b>Vessel Density</b>	<b>29</b>
3.1	Introduction	30
3.2	Density	30
3.2.1	Model	31
3.2.2	Density Computation	32
3.2.3	Geographic Implementation	32
3.3	Visualization	34
3.4	Real-world Data	36
3.4.1	Trajectory Simplification	36
3.4.2	Application	36
3.4.3	Discussion	39
3.5	Conclusion and Future Work	40
<b>4</b>	<b>Vessel Density Evaluation</b>	<b>41</b>
4.1	Introduction	42
4.2	Related Work	43
4.2.1	Evaluation of the Space-time Cube	43
4.2.2	Evaluation of Animation	44
4.3	Scene Design	44
4.3.1	Data Generation	44
4.3.2	Visualization	47
4.4	Evaluation Setup	48
4.4.1	Trial Design	49
4.4.2	Test Design	49
4.4.3	Test Execution	50
4.5	Evaluation Results	51
4.5.1	Hypotheses	51
4.5.2	Analysis	52
4.5.3	Discussion	55
4.6	Conclusion and Future Work	56
<b>5</b>	<b>Density Maps</b>	<b>57</b>
5.1	Introduction	58
5.2	Density Maps	59
5.2.1	Subsets and Parameters	59
5.2.2	Density Aggregation	60
5.2.3	Rendering	61
5.2.4	Image Composition	61
5.2.5	Multivariate Filters	63
5.3	Implementation	63
5.3.1	Parallelization	63
5.3.2	Performance	65
5.4	Use Cases	65
5.4.1	Vessel Traffic	66
5.4.2	Pedestrians in Urban Planning	70

<b>Contents</b>	<b>iii</b>
5.5 Conclusion and Future Work . . . . .	72
<b>6 Composite Density Maps</b>	<b>73</b>
6.1 Introduction . . . . .	74
6.2 Composite Density Maps . . . . .	75
6.2.1 Convolution Block . . . . .	76
6.2.2 Composition Block . . . . .	76
6.2.3 Counter Block . . . . .	77
6.2.4 Enhancement Block . . . . .	77
6.2.5 Enrichment Block . . . . .	77
6.2.6 Iteration Block . . . . .	78
6.3 Implementation . . . . .	78
6.3.1 Performance . . . . .	79
6.4 Maritime Use Cases . . . . .	79
6.4.1 Detailed Harbours . . . . .	80
6.4.2 Drifters . . . . .	81
6.4.3 Sea Lanes . . . . .	82
6.4.4 Traffic Direction . . . . .	83
6.4.5 Risk Analysis . . . . .	84
6.4.6 Vessel Interactions . . . . .	86
6.5 Conclusion and Future Work . . . . .	88
<b>7 Attribute Exploration</b>	<b>89</b>
7.1 Introduction . . . . .	90
7.2 Related Work . . . . .	91
7.2.1 Creating Artificial Vessel Trajectories with Presto . . . . .	92
7.2.2 Piecewise Linear Segmentation . . . . .	93
7.3 Semantic Web Technology . . . . .	96
7.3.1 Simple Event Model . . . . .	96
7.3.2 Moving Object Knowledge Base . . . . .	97
7.4 Trajectory Contingency Tables . . . . .	98
7.5 Use Cases . . . . .	100
7.5.1 Abnormal Mooring . . . . .	100
7.5.2 Banned Vessels . . . . .	102
7.6 System Design . . . . .	103
7.7 Conclusion and Future Work . . . . .	104
<b>8 Conclusions</b>	<b>105</b>
8.1 Achievements . . . . .	106
8.2 Research Contributions . . . . .	108
8.3 Valorization . . . . .	109
8.4 Next Challenges . . . . .	110
<b>Bibliography</b>	<b>113</b>
<b>Publications</b>	<b>127</b>

iv

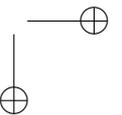
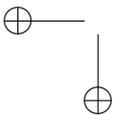
**Contents**

<b>Summary</b>	<b>129</b>
<b>Samenvatting</b>	<b>131</b>
<b>Acknowledgements</b>	<b>133</b>
<b>Curriculum Vitae</b>	<b>135</b>



# Introduction

# 1



*This chapter gives an overview of the research in this project. We elaborate on the position in the society, set the goals for our research, and give an overview of this thesis.*

## 1.1 Movement is Everywhere

In our modern information society, data are stored at a steadily increasing rate. We store data to build a history about what someone or something has been doing. Examples of such data are weather measurements, social networks, and financial transactions. Interesting patterns are encapsulated in these data and domain experts are eager to analyze them to understand, react to, manipulate, or predict the underlying processes of the data. Patterns occur at various levels of detail: there is not only interest in having a global overview of what has happened over time, but also details are of importance, for instance to find outliers.

A special type of historical data is movement data, which contains information on the spatial movement of objects over time. With various digital technology, such as the Global Position System (GPS), wireless sensor networks, telephony networks, Bluetooth, Radio Frequency Identification (RFID), and camera observation networks, we are able to capture and store these movements by means of their location for consecutive moments in time. In different domains, movements of various types of objects are studied with different perspectives and goals, nevertheless, the common goal is finding patterns in the data. Animal research, urban planning, user interface evaluation, and traffic analysis are domains where movement data are analyzed. The study of migration patterns in animal research, the understanding if pedestrians are not getting lost in a city for urban planning, understanding of eye movements for user interface evaluation, and monitoring of traffic jams for car traffic are examples of analyses occurring in these fields. This thesis focusses on the analysis of traffic, in particular vessel traffic, and is driven by the various daily tasks maritime analysts and operators have.

## 1.2 Vessel Traffic

Due to globalization [36] goods need to be transported all over the world, for instance because of low production costs in some areas. Large amounts of goods can be transported by water in a single ship resulting in low transportation costs. In comparison with airplanes for intercontinental transportation and trucks or trains for inland distribution, the travel time is much longer, but many types of goods do not have an urgent need for delivery. The top three harbours [3] shipping the most cargo in 2009 are Shanghai, Singapore, and Rotterdam. Like the rest of the top ten, several megatonnes (Mt) of cargo are transferred per year. Expressed in financial terms [118], this means for instance that Rotterdam harbour had a turnover of half a billion euro in 2010. The capacity of the world merchant fleet in 2010 [147] is 12 megatonnes divided over roughly 45,000 vessels, consisting of oil tankers (4.5 Mt), bulk carries (4.6 Mt), general cargo ships (1.1 Mt), and container ships (1.7 Mt). In the last five years the yearly growth of the fleet was at least 5%.

The overwhelming numbers related to vessel traffic show that the infrastructure at sea is an important part of society. Therefore it needs to be protected against disruptions to keep it available at all time. Coastal surveillance operators monitor vessel movements and react on possible noticeable situations using so-called Maritime Safety and Security (MSS) systems. Monitoring happens at both a local scale where vessels in harbours are managed with a Vessel Traffic Service (VTS) and at a larger scale where coast guards monitor the coastal area of a country. Examples of such noticeable situations are accidents that should be avoided or solved quickly, in particular in busy areas where they may have a large impact on other traffic. Some areas need extra attention, such as national parks, which need to be protected against illegal fishery and pollution, and wind parks, where no regular traffic is allowed. Next to safety issues, there are security issues. For example illegal immigration and smuggling are relatively easy across a natural border formed by a coast line.

### 1.3 Visualization for Vessel Traffic

In some areas, often in front of major harbours, shipping intensity is high. To allow the operator to manage the traffic, understanding of the shipping movements is required. Support by computer systems is desirable, because the information volume and complexity are high. Despite the astonishing processing capabilities of computers, automated support is not straightforward, because it is hard to specify what an operator is looking for, which makes it hard to *automatically detect* these patterns.

One way of dealing with this computational complexity is by presenting the data with interactive graphical representations: *Visualizations*. With graphics we can use the visual cognition of the operator. The human visual system is a unique organ with powerful capabilities to recognize some visual features [158]. When data is encoded appropriately in these visual features, patterns in data can be quickly recognized as well, even when there is no exact specification. By adding interaction to the graphical representations, users are able to adjust the graphics to their needs, which allows them to focus their analyses for understanding the data.

Today, for instance at IEEE VisWeek [80], it is common to divide the visualization domain in three disciplines: scientific visualization, information visualization, and visual analytics. Scientific visualization deals with the rendering of data of geometrical objects, such as an MRI scan, to understand physical phenomena in their own representation. Information visualization shows more abstract data, such as tabular data or collections of text. Visual analytics [143] deals with the process to visually obtain insights in data with techniques from various fields, such as machine-learning and visual cognition. The visualization techniques we propose, interactive cartographical maps, are part of geographical visualization, which focusses on spatial data, and is intertwined with all these visualization disciplines. Movement visualizations show physical phenomena (movements on the earth), with abstract data (tabular data), analyzed with multidisciplinary techniques (semantic web or machine learning). Next to visualization, the analysis of vessel traffic is part of a more general research field where the movement of objects is studied (see Chap. 2).

## 1.4 Trajectories

The movement of objects is often analyzed by means of their *trajectories*. We consider a trajectory, describing the movement of a single object, as all consecutive states during a time period. A state contains attribute values describing the condition of the object at a certain moment in time. Examples of such attributes are *identification number*, *time*, *position*, velocity, object type, size, and fuel consumption. The italicized attributes are first-class citizens and are required to capture the geometry of the movement, i.e., the path of the object. Within trajectories we search for *movement features*, which are patterns within states that can be labeled with meaningful semantics.

In the last decade, the analysis of movements has made a shift from moving points to moving objects. In the past only the geometry has been studied, while the current trend is that additional information is taken into account in the analysis as well. There are two types of non-geometrical data: object data and contextual data. Object data are additional attributes for a state of a trajectory, which are obtained with sensors, from registries, or computed from existing data. Contextual data describe the environment the object is moving in, such as the spatial environment or weather conditions. The data sources together should result in an improved understanding of the semantics of the trajectory of the object. Another trend appearing in the last couple of years is that it has become more and more easy to capture movement data at a large scale: many objects can be tracked for a long period of time with a large number of sensors at relatively low costs. This results in massive data sets for analysis.

This thesis is part of the Poseidon project [48], where our research aims at improving *situational awareness* for operators and analysts of vessel traffic. Situational awareness is a research topic on its own, investigating the understanding of complex dynamic situations and processes using large-scale systems, such as tactical (defense) systems [54], plants [51], or air traffic control systems [50]. Within Poseidon situational awareness boils down to the basic question: “*What was going on?*”. Until 2007 vessel traffic was mainly monitored using radar systems. This technology is accurate for finding ship positions, but it is not possible to identify the observed ship. After 2007, the Automatic Identification System (AIS) [83] (see also Sec. 2.6.2) has become a mandatory device for all passenger ships and merchant vessels engaged on international voyages at sea. With AIS, a ship broadcasts its identification and other data, such as time, location, size, ship type, and destination. Coast guards capture AIS data with a sensor network, which allows them to monitor these vessels. A Maritime Safety and Security (MSS) system is a basis for monitoring vessels by means of AIS, and fused together with other data sources such as radar, camera observations, and the internet.

Our methods are mainly illustrated using an AIS logging of one week of vessel traffic in front of the Dutch coast in the beginning of June 2007. The Dutch coast is an interesting region since there is intensive shipping (260,000 movements in 2009 [94]), both professionally and for pleasure, and there are different sites (157 occurrences in 2009 [94]) that should be protected against shipping, such as sand winning areas, fishing areas, wind parks, nature, harbours, mining, and military training areas. Nevertheless, the number of noticeable events is rather low in 2009 [94]: 94 drifters, 298 traffic violations (194 found with AIS and radar), and 158 newly discovered explosives.

## Trajectories

5

In Figure 1.1 the trajectories of ships for one day are rendered with polylines. In this straightforward rendering we can see the major maritime highways and the locations of some harbours, but the story behind this data is lost. For instance, it is not possible to see when certain shipping movements occurred. When we use the other information included in the AIS data we may be able to get more insight about what was going on during this day. Besides, the usage of the identification allows us to recall an older trajectory of a vessel when it enters the area under investigation. Also by knowing common properties of ships, such as a ship type, we are able to generalize behaviour in a sensible way.



Figure 1.1: *A polyline rendering of one day of vessel traffic in front of the Dutch coast containing trajectories of 695 vessels. In this figure only the spatial attribute is visualized, which reveals, for instance, the major maritime highways. However, if other attributes, such as time or vessel type, would be displayed as well, we could see other interesting patterns.*

## 1.5 Research Question

Our research is driven by the following question to improve situational awareness for vessel traffic:

*How can analysts be supported to understand large amounts of trajectories with multiple attributes by using interactive, visual representations?*

To answer this research question we define several research goals, each dedicated to a specific problem. The solutions to these problems are developed by designing visualization concepts, implementing these in software prototypes, and evaluating the prototypes with users. Our visualization techniques are kept generic, by means of a domain independent model for trajectories and limited assumptions about the maritime domain. We demonstrate this for some of the techniques using data from different domains. The outcome of these studies are described in the next chapters and could be a basis for further research and product development.

## 1.6 Research Goals

Recent research agendas [10, 11, 44], formulated by key researchers in the field, have guided us in formulating research goals aiming for effective methods to visualize movement data. Visualization methods for large amounts of movement data, such as vessel traffic, are considered to be an open issue as well as maps of multi-sensory data.

There are at least three possible approaches to visualize movements as proposed by Andrienko *et al.* [11]: *direct depiction*, *summarization*, and *pattern extraction*. Each of these approaches show a different level of abstraction. Our main contributions are based on summarization, more specific on aggregation, while some minor additional contributions also consider the other two approaches. The specific goals we address are the following:

- I. Movement visualizations should **summarize the relation between the location and time**. Location and time are required attributes to capture the movement itself. As a result it is necessary to understand the relation between them to get insight in the movement. There are different interpretations of time in relation to location: absolute time, relative time, and duration. The former two indicate possible interactions between objects, while the latter relates to the probability to find an object at a location.
- II. New movement visualizations should be **evaluated for the performance in comparison with existing methods**. Quantitative evidence is required to prove that new visualizations are indeed a step forward. By means of formal evaluation it is possible to compare new methods with existing ones under certain circumstances and for given tasks.

- III. Movement visualizations should also **display distributions of other attributes than location and time** given by the data. The semantics of movement may be derived from additional attributes, such as a weight or a type. Summaries should show areas with common values for certain attributes indicating classes of similar movement and suggesting a possible meaning for that area. Examples for vessel traffic are anchor zones, where vessels drop anchor, or specific harbours, such as oil depots or passenger terminals where only ships of the same type moor.
- IV. Movement visualizations should **display distributions of derived attributes**. Semantics of movement may be given by derived attributes, such as physical impact during a collision or average speed per object type. Derived attributes contain more sophisticated definitions of the features of interest and hence are potentially more valuable than the base attributes. We expect that by allowing the users to define these derived attributes themselves and show them in the way they like, improves the user experience as well.
- V. Movement visualizations should enable the user to **explore correlations between different attributes**. When movement is captured with multiple sources it is likely that some attributes in the data are related. To figure out which attributes are related, explorative tools are needed to find correlations occurring in the data. One of the challenges is to find a way to depict correlations.

## 1.7 Overview

The remainder of this thesis describes the context in more detail, presents proposals for visualization techniques, and discusses the steps made towards the research goals of the previous section. Chapter 2 gives a brief introduction in the field of moving object analysis, vessel traffic, and related visualization techniques in literature. Chapters 3 to 7 present and evaluate the visualization techniques we developed and form the main contribution of this thesis. The relations between these chapters is graphically depicted in Figure 1.2. All proposed visualization techniques display trajectories with attributes. There are two different directions of research. In Chapters 3 to 6 we present a summarization method with various extensions and in Chapter 7 we show an explorative method by means of direct depiction and pattern extraction.

Chapter 3 presents a multi-scale visualization technique [162] for showing an aggregated summary of the relation between location and duration of large amounts of movements. The contribution consists of two parts. The first part is a density model inspired by Kernel Density Estimation (KDE) that describes the density of vessels, averaged over time. For different scales, obtained by varying the kernel size, different movement features appear. The second part is a multi-scale rendering technique, where two density fields are displayed simultaneously using color mapping and an illuminated height field. As a result multiple types of movement features popup. The movement features are tested with vessel data and show that the movement features shown are important maritime movement patterns.

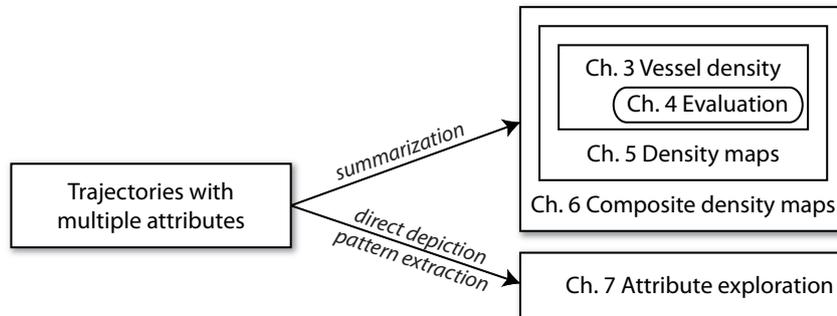


Figure 1.2: Overview of the chapters and their relations between them.

Chapter 4 evaluates vessel density as introduced in Chapter 3. This is done via a formal user study [163] with maritime inspired tasks defined to test the performance of the vessel density visualization, in comparison to other well-known visualization techniques. In the study we use synthetic data allowing us to create data matching with the tasks. Real-world data is simulated by varying the number of trajectories and their visual complexity. It turns out that vessel density excels for finding stoppers and does not perform significantly worse for finding busy routes and fast movers.

Chapter 5 is an extension of Chapter 3 where multiple attributes in the data are taken into account in the visualization technique [129, 130], called density maps. Instead of computing two density fields for different kernel sizes as in the previous approach, we compute multiple density fields for various subsets of the data. The subsets are defined by criteria on the attributes. The visualization technique combines these fields in two different ways: density-based or image-based. For the density-based approach, the density fields are combined in a single density field that is rendered to an image. For the image-based approach, the renderings of the individual density fields are composed into a single image. We propose for both approaches some variations, each with their specific pros and cons in visualizing the data.

In Chapter 6 the approach of Chapter 5 is generalized. Mathematical expressions can be used to relate multiple attributes to a density field instead of just a filtering criterium [128]. Furthermore, a composite density map is created via a composition of predefined building blocks containing a process to create or enhance a density field based on trajectory data and other density fields. These two improvements together have more expressiveness than the density maps in Chapter 5, which enables users to define more sophisticated computations related to their analysis.

Chapter 7 presents a visual analytics toolkit [164] for finding correlations between attributes of trajectories. This work is an integration of results obtained in the Poseidon project [48]. With multiple views we enable the user to find distributions within a single attribute and use this information to drill down or to compare the attribute with another attribute to find possible correlations. The trajectories with attributes are stored in a knowledge base that allows us to easily extend the data set with other data sources and relate them at a semantic level. Another way to use the knowledge

base is by using its reasoning engine to derive additional attributes for trajectories, not only by means of the combination of attributes of trajectories, but also with contextual data. The system as a whole is a flexible pipeline to combine data sources and visually examine hypotheses about trajectories.

Finally, in Chapter 8 we discuss the presented visualization techniques and reflect on our research goals. Furthermore, we provide suggestions for future research.

## 1.8 Related Publications

Chapters 3 to 7 are based on papers presented at conferences and published in journals in the field of information visualization and geographic information science. The papers are categorized as primary and secondary. In the primary papers, the research is fully conducted by Niels Willems and his supervisors. The secondary papers contain work where Niels Willems collaborated with researchers of the visualization group of Eindhoven University of Technology, with researchers in the Poseidon project, and with international key researchers. For all these papers Niels Willems has taken the lead in the concept design and the writing and editing of the papers.

### 1.8.1 Primary Papers

NIELS WILLEMS, HUUB VAN DE WETERING, AND JARKE J. VAN WIJK, Visualization of Vessel Movements. *In Computer Graphics Forum (Proceedings of EuroVis 2009)*, 28(3), pp. 959–966, 2009, [162].

Related chapter: 3

NIELS WILLEMS, HUUB VAN DE WETERING, AND JARKE J. VAN WIJK, Evaluating the Visibility of Vessel Movement Features in Trajectory Visualizations. *In Computer Graphics Forum (Proceedings of EuroVis 2011)*, 30(3), pp. 801–810, 2011, [163].

Related chapter: 4

NIELS WILLEMS, ROELAND SCHEEPENS, HUUB VAN DE WETERING, AND JARKE J. VAN WIJK, Visualization of Vessel Traffic. *In Poseidon: Situational Awareness with Systems of Systems*, Pierre van de Laar, Jan Tretmans, and Michael Borth, Eds., Embedded Systems Institute (ESI), 2012, *to appear*, [161].

Overview paper related to all chapters

### 1.8.2 Secondary Papers

ROELAND SCHEEPENS, NIELS WILLEMS, HUUB VAN DE WETERING, AND JARKE J. VAN WIJK, Interactive Visualization of Multivariate Trajectory Data with Density Maps. *In Proceedings of IEEE Pacific Visualization Symposium*, pp. 147–154, 2011, [129].

Related chapter: 5

ROELAND SCHEEPENS, NIELS WILLEMS, HUUB VAN DE WETERING, AND JARKE J. VAN WIJK, Interactive Density Maps for Moving Objects. *In IEEE Computer Graphics and Applications*, 32(1), 2012, [130].

Related chapter: 5

ROELAND SCHEEPENS, NIELS WILLEMS, HUUB VAN DE WETERING, GENNADY ANDRIENKO, NATALIA ANDRIENKO, AND JARKE J. VAN WIJK, Composite Density Maps for Multivariate Trajectories. *In IEEE Transactions on Visualization and Computer Graphics (Proceedings of InfoVis 2011)*, 17(12), pp. 2518-2527, 2011, [128].

Related chapter: 6

NIELS WILLEMS, WILLEM ROBERT VAN HAGE, GERBEN DE VRIES, JEROEN H.M. JANSSENS, AND VÉRONIQUE MALAISÉ, An Integrated Approach for Visual Analysis of a Multisource Moving Objects Knowledge Base. *In Proceedings of the AGILE Workshop Geospatial Visual Analytics: Focus on Time*, 2010, [165].

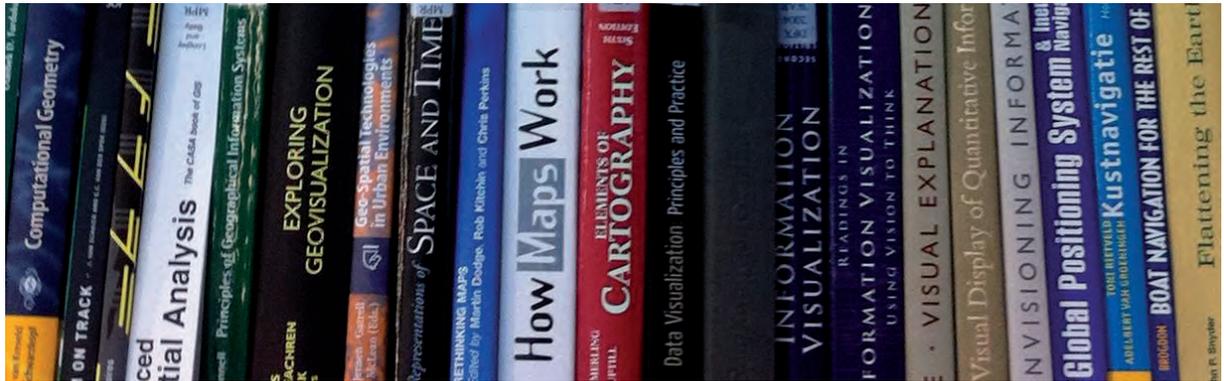
Related chapter: 7

NIELS WILLEMS, WILLEM ROBERT VAN HAGE, GERBEN DE VRIES, JEROEN H.M. JANSSENS, AND VÉRONIQUE MALAISÉ, An Integrated Approach for Visual Analysis of a Multisource Moving Objects Knowledge Base. *In International Journal of Geographical Information Science*, 24(10), pp. 1543–1558, 2010, [164].

Related chapter: 7

## Background

# 2



*In this chapter we present the background for the visualization techniques as proposed in the following chapters. Visual vessel traffic analysis is one of the applications in the field of moving object analysis. We briefly introduce the basic concepts from a visualization perspective and address moving object analysis, maritime domain knowledge about shipping, and the state-of-the-art in visualization research.*

## 2.1 Moving Object Analysis

The study of the movement of objects originates from time geography as introduced by Hägerstrand [67] in the early fifties. However, before this time people were already struggling with the notion of movement and how to present it. Evidence is found in the chart (see Fig. 2.1) that Minard created to show the movement of Napoleon’s army against the Russians in 1812. Minard’s chart is even today still an inspiration for research and design [58]. Nowadays, the movement of objects is studied in many different research areas in computer science. We see evidence for this in European projects like MODAP [104] and MOVE [79] where many different fields within computer science are present, such as computational geometry, spatio-temporal databases, semantic web technology, formal methods, and visualization. Although the approaches in these fields are different, they all rely on a common model for movement: *trajectories*.

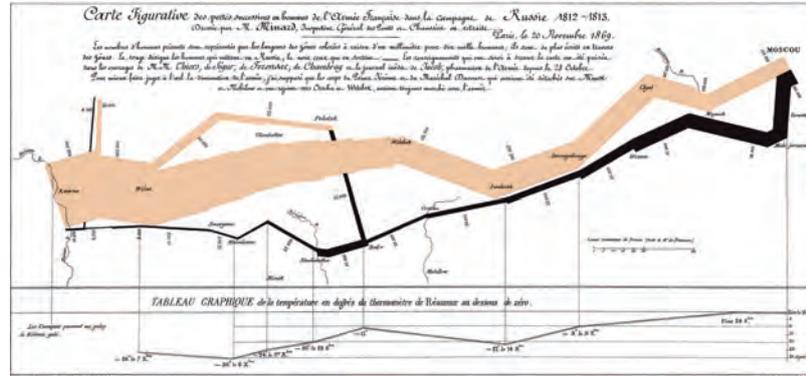


Figure 2.1: Charles Minard’s chart showing the losses in men, their movements, and the temperature during Napoleon’s 1812 Russian campaign.

Trajectories are a specialization of time-series data. Time-series are visualized with a wide range of tools [74, 153, 157]. Furthermore, trajectories are a specialization of spatio-temporal data [115], such as spread of a disease over time [100], spread of eddies in oceans [166], or migration patterns [25] (see Fig. 2.2). However, a full treatment of these two data types is out of the scope of this thesis. For further reading we recommend Andrienkos’ textbook [13] on spatio-temporal data visualization. The remainder of this chapter presents basic concepts of moving object analysis and visualization.



Figure 2.2: *Flowrates* [25] show global migration patterns with origin-destination edges, which are decorated with an intensity plot varying over time. Image by courtesy of Ilya Boyandin.

## 2.2 Data Model

In this section we give a data model for trajectories to represent object movements. A trajectory contains the conditions under which an object moved. These conditions change over time and are captured in multivariate data, data with multiple attributes. Attribute values of moving objects are functions of time and defined for any moment. In practice, these functions are sampled, since computers have only a finite amount of memory. Some calculations require to reconstruct the continuous movement from these samples. In this section we discuss these concepts.

### 2.2.1 Continuous Trajectories

We consider a set of objects  $O$ . The movement of an object  $o \in O$  is described continuously in time, which is modeled by a continuous trajectory  $\alpha^o$ . At each moment in time  $t$  the conditions of the object movement are captured in a state  $\alpha^o(t)$ , which is a tuple of attributes. Consequently, the attributes are varying with time. For a single object  $o$ , we abbreviate the state as  $\alpha(t)$ . A trajectory requires the state to contain an identification  $id$ , a timestamp  $t$ , and a position  $\mathbf{p}(t)$ . A multivariate trajectory may also contain other attributes, such as a velocity  $v(t)$ , a width  $w$ , and a mass  $m$ .

### 2.2.2 Trajectory Reconstruction

In real-world data a continuous trajectory  $\alpha^o$  is sampled at various moments in time, which results in a sequence  $\tau_o$  of tuples  $\alpha_i^o$ . For a single object  $o$ , we abbreviate a tuple  $\alpha_i^o$  as  $\alpha_i$  and its attribute values appear as  $t_i$  for a timestamp,  $\mathbf{p}_i$  for a position,  $v_i$  for a velocity, and so forth. The tuples are ordered by time and most attributes can be interpolated between consecutive tuples to approximate the original continuous trajectory. We can use a number of standard interpolation schemes, such as nearest neighbor, linear, or spline interpolation to obtain transitions between consecutive interpolated tuples. We can interpolate for example in time or use the displacement assuming a constant velocity. For many attribute types these schemes give result in a sufficient approximation of the reconstructed values.

For position, however, we propose to use a different interpolation scheme, since not only the position and time, but also the magnitude of the velocity may be known in both sampled tuples. Our way allows for a more accurate approximation, and as a result less tuples are needed for a faithful reconstruction of the original movement. Consider two consecutive tuples, for instance the first two tuples  $\alpha_0$  and  $\alpha_1$ . We assume that objects move over a straight line segment between points  $\mathbf{p}_0$  and  $\mathbf{p}_1$ . For line segment  $\mathbf{p}_0\mathbf{p}_1$  we obtain the reconstructed positions  $\tilde{\mathbf{p}}$ , such that  $\tilde{\mathbf{p}}(t_0) = \mathbf{p}_0$  and  $\tilde{\mathbf{p}}(t_1) = \mathbf{p}_1$ , with

$$\tilde{\mathbf{p}}(t) = \mathbf{p}_0 + x(t) \frac{\mathbf{p}_1 - \mathbf{p}_0}{\|\mathbf{p}_1 - \mathbf{p}_0\|}, \quad (2.1)$$

where displacement  $x(t)$  along the line segment at time  $t \in [t_0, t_1]$  fulfills the following boundary conditions:

$$x(t_0) = 0 \wedge x(t_1) = \|\mathbf{p}_1 - \mathbf{p}_0\|. \quad (2.2)$$

We assume that the acceleration is constant, and hence kinematics gives us the following expression for the displacement  $x(t)$ :

$$x(t) = \frac{1}{2} \tilde{a}(t - t_0)^2 + \tilde{v}(t - t_0), \quad (2.3)$$

with unknown variables  $\tilde{a}$  for the acceleration and  $\tilde{v}$  for the velocity. To find the unknowns we minimize energy  $E$ , which is the sum of the squared differences between the modeled velocities  $\dot{x}(t_i)$  and the given velocities  $v_i$ , i.e.,

$$E = (\dot{x}(t_0) - v_0)^2 + (\dot{x}(t_1) - v_1)^2. \quad (2.4)$$

We substitute  $\alpha := \tilde{a}(t_1 - t_0)^2$  and  $\beta := \tilde{v}(t_1 - t_0)$  in the displacement  $x(t)$  given in Equation (2.3) resulting in

$$x(t) = \frac{1}{2} \alpha \left( \frac{t - t_0}{t_1 - t_0} \right)^2 + \beta \frac{t - t_0}{t_1 - t_0}. \quad (2.5)$$

From the boundary condition in  $x(t_1)$  follows that  $\alpha = 2(\|\mathbf{p}_1 - \mathbf{p}_0\| - \beta)$ , and minimizing  $E$ , i.e., solving  $\frac{d}{d\beta} E = 0$ , results in

$$\beta = \|\mathbf{p}_0 - \mathbf{p}_1\| - (t_1 - t_0) \frac{v_1 - v_0}{2}. \quad (2.6)$$

Substitution of  $\beta$  in  $\alpha$  gives  $\alpha = (t_1 - t_0)(v_1 - v_0)$ . We reverse the substitution and derive the reconstructed acceleration  $\tilde{a}$  and velocity  $\tilde{v} = \dot{x}(t_0)$  in  $\mathbf{p}_0$ , which are given by

$$\tilde{a} = \frac{v_1 - v_0}{t_1 - t_0} \wedge \tilde{v} = \frac{\|\mathbf{p}_1 - \mathbf{p}_0\|}{t_1 - t_0} - \frac{v_1 - v_0}{2}. \quad (2.7)$$

By inverting Equation (2.3) we find  $t$  given the position  $x$  if we assume that vessels do neither stop nor return, that is when  $\dot{x}(t) > 0$ :

$$t(x) = \begin{cases} t_0 - \tilde{v}/\tilde{a} + \sqrt{\tilde{v}^2 + 2\tilde{a}x/\tilde{a}}, & \text{if } \tilde{a} \neq 0; \\ t_0 + x/\tilde{v}, & \text{if } \tilde{a} = 0. \end{cases} \quad (2.8)$$

This reconstruction model is used throughout the following chapters.

## 2.3 Characteristics

The data model enables uniform description of arbitrary moving objects, despite the strongly varying characteristics of the data. In this section we elaborate on some of the reasons for the varying characteristics.

Firstly, sensors should fit the object in order not to influence its motion. For instance, a bird is tracked with a small device to allow the bird to fly as usual, while for a cargo ship a larger device can be used, since it has plenty of space. A small tracker device is less accurate and records less often the current position than a larger one, due to the limited capabilities resulting from limited energy resources. Detailed movements

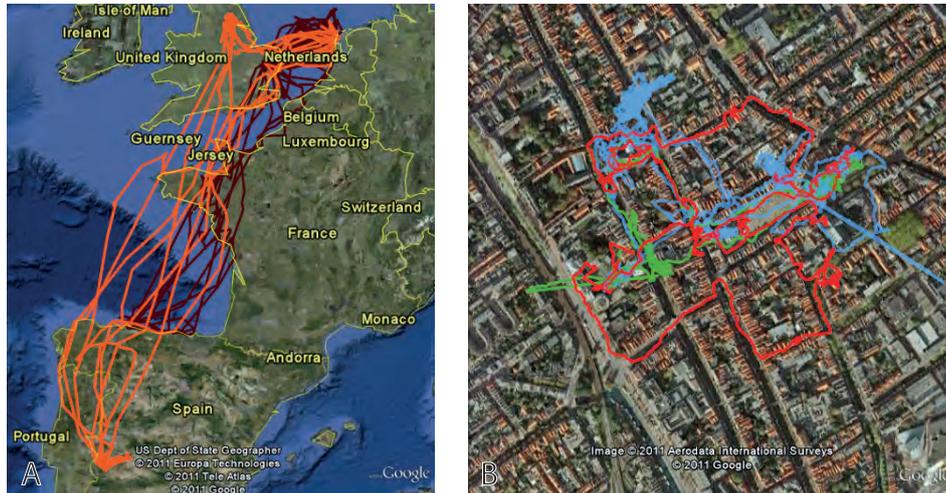


Figure 2.3: Different sampling rates for coarse and fine trajectories visualized as a polyline in Google Earth [61]. (A) A coarse sampling of two lesser black-backed gulls [136]. (B) A fine sampling of three pedestrians in the city center of Delft [149].

are hence not visible in the data. On the other hand, with large trackers movements may be too frequently recorded, resulting in huge data sets. Figure 2.3 shows some examples of different samplings for birds and pedestrians.

Objects are constrained in their movement by their environment or by their own abilities. For spatial movement there is a scale from network-constrained movement to free movement. A train can only follow rail tracks, while a mouse can move freely in his cage. Large vessels are constrained in some areas where they should move in maritime highways, which are wide and allow for minor spatial deviations. Another spatial constraint is accessibility, since in some areas it is not possible to move, for example, a ship cannot sail on land. Apart from the spatial constraints, there are other constraints given by the capabilities of the object. For example, a large vessel cannot make sharp turns.

Two other trajectory characteristics are scale and complexity. Scale is the spatial coverage of movements: the more area is traversed, the larger the scale to see all movements. Complexity is related to the maneuverability of the object. Trajectories with little turns and little spatial crossings are simple, while trajectories with more turns and crossings are complex. Scale and complexity are independent as shown with examples of the combination of these two in Figure 2.4. Despite the independence of the two characteristics, for visualization there is a relation, since some ratio between scale and complexity results in similarly looking pictures. For example, the real-world examples of Dutch vessel traffic and pedestrians in Delft shown in Chapter 5 tend to look similar: although Delft is hundreds times smaller, the number of movements tends to be of the same order of magnitude.

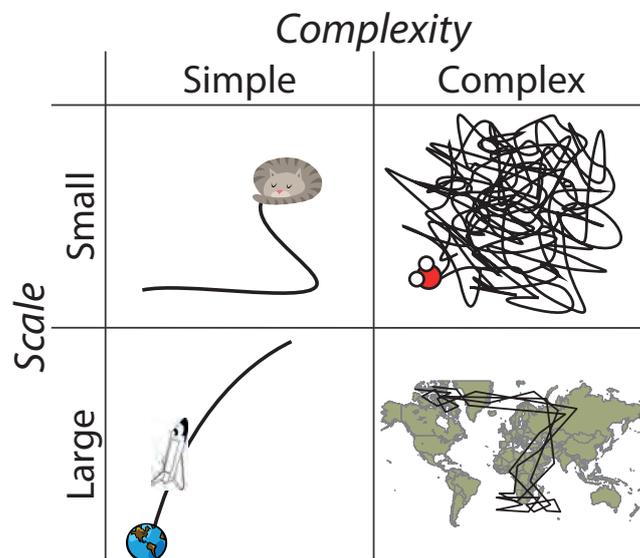


Figure 2.4: Trajectory sketches for different scale and complexity, from left to right and from top to bottom: a sleeping cat, a boiling water molecule, a departing space shuttle, and a migrating bird.

The last characteristic is interaction, which relates to dependencies between the movements of different objects. We discuss two types of interactions occurring in our research. The first type of interaction is imitation, where one objects copies the movement of another object. For animals these interactions may occur as a flock, for human movements it may occur as marching, and in vessel traffic it occurs when a tug pulls a tanker into the harbour. The second type of interaction is co-location, which happens when multiple objects go to or move via the same location. For traffic such locations may be roads, and for humans these locations may be supermarkets where we buy our groceries every day. If we also take time into account, the co-location turns to a meeting, where objects meet at a location and proceed moving together. For humans meetings may occur as a rendezvous, or for cars they may occur as a collision. One of the properties that makes a meeting interesting is the fact that goods can be exchanged from one object to another. In vessel traffic such an exchange may be an indication of smuggling, for instance when two small boats meet each other at open sea.

## 2.4 Analysis Tasks

In the visual analysis of moving objects there are a number of generic tasks applicable for an arbitrary data set. The complexity of the tasks depends on the role a user has while analyzing the data. We distinguish between two types of user roles: operators and analysts. Operators are often involved in real-time monitoring, for instance of car traffic during rush hours. They need to have an overview of the situation, find potential dangerous situations, and should be able to drill down in the data to find out more about specific objects. The complexity of these tasks is relatively small, since operators are trained to find and react on known situations. On the other hand, analysts are involved in more complex tasks, since they often analyze the data ad-hoc and for less well-defined tasks. Analysts do not only need to have an overview of the movements, but often need to understand the movements of many objects simultaneously to draw conclusions.

When considering the tasks of both operator and analyst, Andrienko and Andrienko [13] suggest, with the inspiration of Bertin [22], to divide the tasks in two classes: elementary tasks and synoptic tasks. This classification is intended for spatio-temporal data, and hence also applies for movement data. Elementary tasks are related to individual objects, while synoptic tasks are related to the data set as a whole, or a significantly large subset. For both roles there are tasks in both classes. We give a brief description and some examples for these tasks.

### 2.4.1 Elementary Tasks

A user conducting an elementary task aims at finding out details about individual objects. For instance, they want to know for which objects a certain property holds. Based on these insights the user may take action for these individuals. Some elementary tasks for trajectory data are:

- Identification of simple movement patterns that can be directly read from the attribute values, such as stops and turns.
- Identification of abnormal movements based on extreme attribute values, such as objects moving faster than they theoretically could.
- Identification of objects or movements of a certain class based on their attribute values, such as filtering by object type.
- Comparison of objects or movements by attribute values, such as the speed differences between two areas.

The main characteristic of elementary tasks are the direct reading of attribute values, while in synoptic tasks the user searches for more global measures.

### 2.4.2 Synoptic Tasks

Synoptic tasks are related to the overview of trajectories. In these tasks users are often interested in behavioural characteristics, such as changes of movement over time, number of objects matching a certain criterium, and typical movements in a certain area. These insights may result in actions that often have impact on all the movements, such as changes of the environment. Some examples of synoptic tasks for trajectory data are:

- Identification of trends in movements, such as similar movements that occur in a certain area.
- Identification of abnormal movements based on normal behaviour extracted from historical movements.
- Display of the distribution of where certain classes of movements occur.
- Identification of how movements in two areas differ.

Most of these tasks can be solved in combination with visualizations. In the next section we illustrate the analysis tasks further with visualizations that are developed for concrete tasks.

## 2.5 Movement Visualizations

Movements can be visualized in many ways. In this section we discuss the state-of-the-art of movement visualization techniques for some of the tasks described in the previous section. Since visualizations often only support an analyst in finding one type of movement feature, we have grouped them by similar types of features. At the top level, we distinguish object and aggregated movement features. Object movement features concern semantics given to movements of a specific object or set of objects and support the user to perform some elementary tasks. Aggregated movement features show trends in large amounts of trajectories and are a tool to support for some synoptic tasks.

### 2.5.1 Object Movement Features

By investigating the shape of the path of a small number of objects, we can attach semantics to the movement. Dodge *et al.* [41] defined a taxonomy with examples of movement features that may occur in a movement data set. Examples of these patterns are co-location in space, concentration, flocking, periodicity, meeting, and leadership. We distinguish three groups of visualizations based on the type of data. The first group contains event-based visualizations, where trajectories are displayed with a coarse sampling and where detailed reconstruction of the trajectory is hard. The second group of visualizations shows trajectories only by means of space and time. The last group displays multivariate trajectories, which are trajectories that include additional information.

#### Event-based Visualizations

Sequences of spatio-temporal events, records of what happened at a certain time and place, can sometimes be considered as a trajectory, when the records are about objects. Often these sequences have a low sampling rate and as a result it is hard to reconstruct the detailed movements of objects. However, these events are sufficient to find reasons why an object moved from one place to another. For instance, Weaver *et al.* [159] analyze hotel visits and find some interesting patterns, such as frequent visitors or the influence of weather. Another approach for event data is given by Andrienko *et al.* [12] to find events of public interest.

In the early fifties, Hägerstrand [67] introduced, among other concepts, the space-time cube, where time is seen as the third dimension perpendicular to the spatial plane (see Fig. 2.5). Space-time paths are the 3D polylines generated by connecting events at a location and time in this cube and show various patterns, such as meetings by bundles of lines and stops by straight lines in the time direction. The path can be extended with a space-time prism, which is a volume of the reachable area over time from a given location or between two given positions. In recent research, the space-time cube is still used, for instance, Eccles *et al.* [46] use the space-time cube for story telling based on spatio-temporal events. The space-time cube is not only useful for event-based data, but also for more detailed trajectories with many events, which we show in the next visualization type.

#### Trajectory Visualizations

Newly developed location trackers automatically generate data for detailed trajectories and have been a stimulus for new research. For example, Kraak [92] uses the space-time cube as an interactive geovisualization, which is an early development of modern visual analytics tools. Romero *et al.* [125] use an interactive space-time cube to analyze activities in a video stream. Forer [57] visualized Hägerstrand's space-time prisms. Trajectories of eye movements can be analyzed using a gaze plot, or a space-time cube as shown by Li *et al.* [96], who adapted the latter with the TimeWave principle, where a clock is traced along a time line, to find and compare repetitive patterns.

Next to the space-time cube there are more ways to visualize trajectories. For instance, repetitive patterns can be found by drawing trajectories on a circular diagram as proposed by Zhao *et al.* [173]. Another pattern is the relative motion between objects, which is emphasized in a visualization proposed by Crnovrsanin *et al.* [37], where abstract representations may solve possible confusions when movements are only displayed spatially. Anomalous behaving objects can be found by means of wavelets [85] as demonstrated with pedestrian data or with a combination of machine learning techniques and visual analytic tools as shown by Liao *et al.* [97] with trajectories of taxis. Apart from historical data analysis, there is a range of visualizations for real-time traffic, such as for air traffic control [19].

### Multivariate Trajectory Visualizations

Trajectories may be measured with multiple sensors resulting in multivariate data. Noticeable attribute values appearing in the trajectories may be directly related to the semantics. Standard multivariate data visualizations, such as parallel coordinate plots [81], small multiple visualizations [21, 35, 126, 138], and interactive scatterplot matrices [47, 66], are not sufficient, because data points of the same trajectory are not connected. A common approach for visualizing multivariate trajectories is by displaying them with multiple linked views. Dykes *et al.* [45] analyze these kind of trajectories with a combination of spatial views, such as maps and video, and standard multivariate views, such as parallel coordinate plots. Multiple views are used to understand reasons

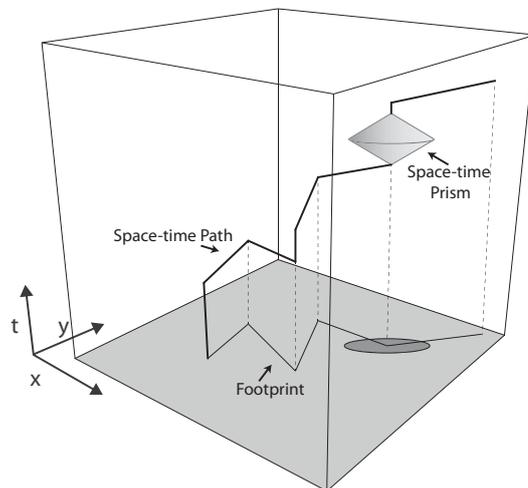


Figure 2.5: A space-time cube visualizing a single space-time path. The cube consists of two spatial axes and one time axis with the start time at the bottom. The projection of the space-time path is called a footprint. A space-time prism allows for insights about unknown relations between space and time, such as for missing data or future movements.

why certain movements have happened based on significant patterns appearing in one view. By means of linking and brushing patterns can be connected to other views. A visual analytics tool to find anomalous behaving vessels is demonstrated by Riveiro *et al.* [122], which uses similar views, but also includes machine learning techniques, such as self-organizing maps shown in one of the views. Guo *et al.* [65] propose a multiple view approach to investigate traffic crossing a road intersection and where direction of traffic plays a major role. Direction is addressed by a ThemeRiver-inspired graph with arrow glyphs and brushing for trajectories coming from a similar direction.

Other approaches show multivariate trajectories in a single view. For example, Slingsby *et al.* [134] define a hierarchy-based on attributes of trajectories and use tree maps to show the resulting distribution. Hurter *et al.* [77] pioneer with high-end graphics hardware to visualize various attributes along airplane trajectories using interactive masking and the rolling-the-dice paradigm [47] used for scatterplot matrices.

### 2.5.2 Aggregated Movement Features

A special kind of movement visualization is trajectory aggregation, where an overview of large amounts of movements is shown for analyzing trends in the movements [17]. An overview is created by means of aggregating parts of trajectories that match a similar entity. Such an entity can be, for instance, a location or an area. In the remainder of this section we discuss different entities with various visualization techniques.

#### Grid-based Aggregation

A common approach to aggregate trajectories is with a regular grid structure, where entities are areas represented by cells. Andrienko and Andrienko [7] show the aggregated cells with glyphs on top of a map, which shows the usage of a road network at various times simultaneously. An example for vessel traffic is shown in Figure 2.6.

Another grid-based approach is density, which is based on convolution and shows a smoothed view on the data. Smoothing is a common visualization technique, which is used in for instance motion blur [139], semantic depth of field [91], or hotspot analysis [4, 55]. One of the applications of data smoothing is called Kernel Density Estimation (KDE) [133], which computes, based on a set of samples of a random variable, an estimation of the corresponding probability density function. The samples are treated as data points and convolved with a kernel, often a Gaussian. The estimated probability density function reduces sampling artifacts.

KDE is defined for data points, but it is also possible to smooth curves. In this case the kernel is moved along the curve and results in a fat curve with a smooth fall-off. Using graphics hardware curves can be smoothed fast, however the performance depends on the quality of the approximation of the smoothing. Hurter *et al.* [78] splat smoothed points on a map to obtain accumulation maps of air traffic. These accumulation maps can also be used as the height in a 3D polyline rendering of trajectories as shown in Figure 2.7. World-wide trajectories and their aggregating 2D histograms are displayed by Grundy *et al.* [63] in a special version of the space-time cube where the cube is folded around the globe. A more accurate approximation for density is given

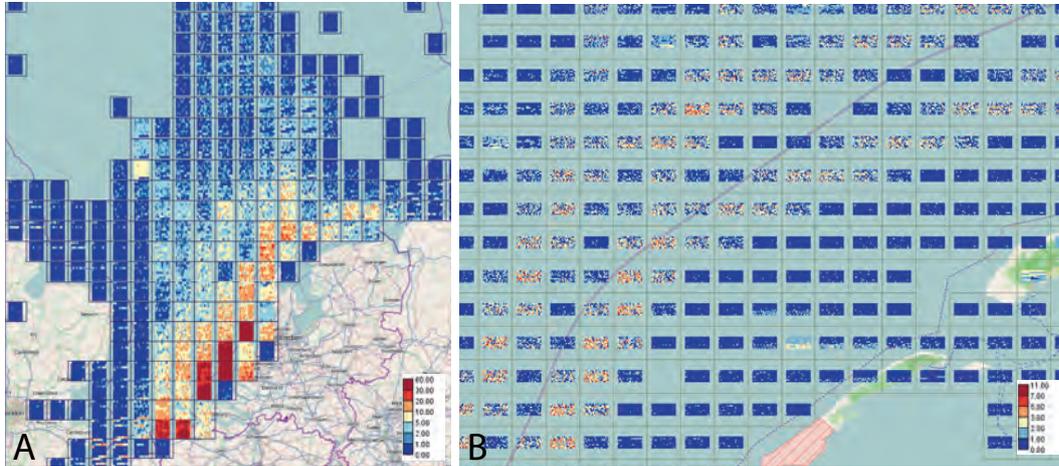


Figure 2.6: A glyph-based aggregation [7] of vessel traffic in front of the Dutch coast. Each glyph is a bitmap, where a row represents a day and the colors represent the number of trajectories. (A) Full coverage with columns of two hours. (B) The Frisian Islands area in the North of the Netherlands with columns of one hour. Images by courtesy of Gennady and Natalia Andrienko.

by Lampe and Hauser [95] who convolve trajectory segments with a constant speed to show the intensity of vessel traffic (see Fig. 2.8). For each segment three textures are splat: two at the end points and one around the segment, where the fall-off perpendicular to the segment is the same from begin to end. Convolved line segments occur for instance in continuous parallel coordinate plots [72] as well and can be computed analytically using polynomial kernels [87].

Downs [43] proposes a density for trajectories by splatting a kernel adapted to the length of a segment and assumes a constant velocity. Peeters and Krisp [114] use a splatting technique with an anisotropic kernel for short line segments to show aggregate directions. By convolving the path in a space-time cube with a 3D kernel and display the result with volume rendering techniques, temporal information can be revealed as well, as proposed by Demšar and Virrantaus [40] and displayed in Figure 2.9.

### Aggregation with Irregular Entities

Trajectories may also be aggregated by means of other structures than a grid. For instance, Andrienko and Andrienko [16] extract a graph from trajectories and aggregate them along the graph to construct a flow map. This approach has as advantage that the number of aggregate entities, viz. routes represented by edges, is low while it preserves the natural structure of the data, since in a grid approach a large grid is required. Another way of preserving the natural structure is by clustering [8] the trajectories in similar groups, as demonstrated with traffic data to get a clear and simple overview of

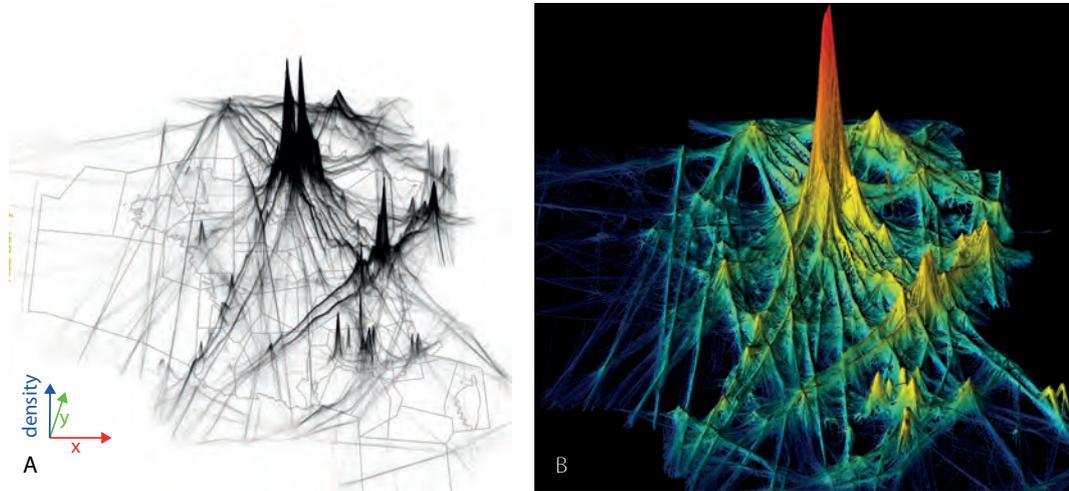


Figure 2.7: 3D accumulation maps [78] of French air traffic. (A) Trajectories drawn as 3D polylines with density as third dimension. (B) A color mapped surface of the density. Images by courtesy of Christophe Hurter.

traffic situations on a map. Bak *et al.* [20] use given locations to aggregate at significant locations within mouse trajectories, such as watering places. The aggregates show for instance the duration with glyphs. Finally, when only the aggregate entities are given, they can be grouped with self-organizing maps as proposed by Andrienko *et al.* [9].

## 2.6 Vessel Traffic

In this section we present basic domain knowledge necessary to understand vessel traffic trajectories. We describe the navigation principles, usage of trackers, and vessel tracking systems.

### 2.6.1 Navigation Principles

A position  $(\phi, \lambda)$  on the earth is specified with geographical coordinates consisting of latitude  $\phi$  and longitude  $\lambda$ , which are given in a number of degrees. Notice that geographical coordinates are independent of the radius of the earth  $R$ , which varies between 6,357km and 6,378km. For small-scale maps, like ours, (up to 1:5,000,000) the earth can be approximated by a sphere with radius 6371km, where large-scale maps require a more complex elliptical model [88].

A map projection  $m$  with  $(x, y) = m(\phi, \lambda)$  maps geographic coordinates to a plane, and often has an inverse  $m^{-1}$  [135]. Thousands of map projections with various properties have been developed. The main properties aimed of are preservation of angles (conformal, e.g., Mercator projection  $M$ ), preservation of distance (equidistant, e.g.,

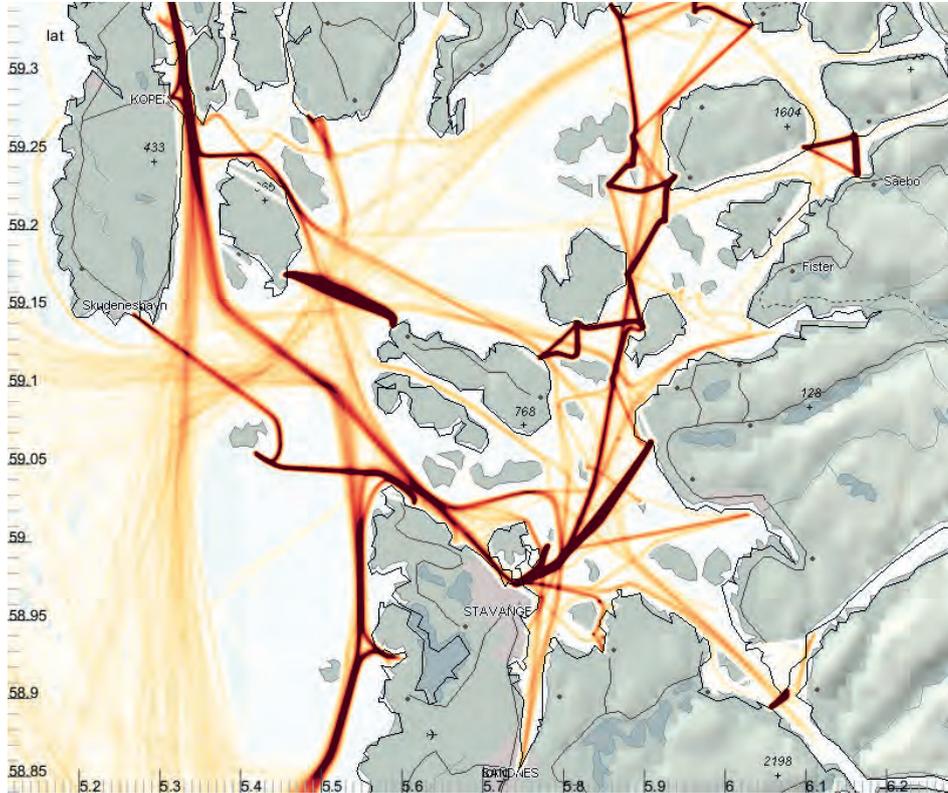


Figure 2.8: A GPU-accelerated kernel density estimation visualization [95] of vessel traffic in the South of Norway. Image by courtesy of Ove Daae Lampe.

equiarectangular projection  $E$ ), and preservation of area (equal area, e.g., sinusoidal projection  $S$ ). No map projection satisfies all these properties.

The intersection of the surface of the earth and a plane through its center is called a great circle, such as the prime meridian or the equator. The great circle distance  $d_{gc}(\mathbf{p}_0, \mathbf{p}_1)$  is the shortest distance between two points on the earth and is measured along the great circle through these points. Other circles on the earth are small circles.

Means of transportation that are not route-bound, such as vessels and airplanes, navigate using a compass directions to move to a destination. A direction on the earth is often given by a number of degrees to the reference direction towards North (see Fig. 2.10 for a vessel). A vehicle moves in the course over ground direction. When a vehicle is not attached to the ground, it can drift due to wind or sea current, which introduces an additional direction, the heading, which is the actual orientation of the vehicle. A vehicle drifts when the difference between these two directions is large.

A route for these vehicles is determined by using a line segment starting in point  $\mathbf{p}_0$  and proceeding with a fixed compass direction towards point  $\mathbf{p}_1$ : a rhumb line segment  $\overline{\mathbf{p}_0\mathbf{p}_1}$  [2]. A rhumb line segment  $\overline{\mathbf{p}_0\mathbf{p}_1}$  is given by a straight line segment equation

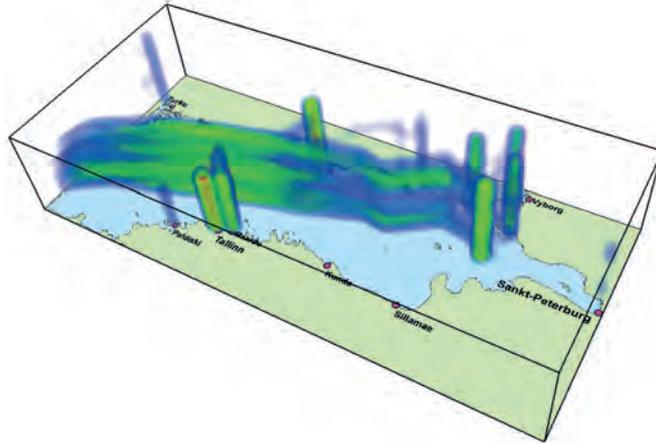


Figure 2.9: *Space-time density [40]: Convolved vessel trajectories in a space-time cube around the South of Finland. Image by courtesy of Urška Demšar.*

in a Mercator projection and has length  $\|\widehat{\mathbf{p}_0\mathbf{p}_1}\|$ . A natural parametrization  $RL(\mathbf{p}_0, \mathbf{p}_1, s)$  between points  $\mathbf{p}_i$  for  $s \in [0, 1]$ , is given by a dead reckoning [53] principle, which constructs a rhumb line segment starting in  $\mathbf{p}_0$  with the same direction as  $\widehat{\mathbf{p}_0\mathbf{p}_1}$  and length  $s\|\widehat{\mathbf{p}_0\mathbf{p}_1}\|$ . A rhumb line is not equal to a great circle, and hence its path may not be the shortest, however, for a short distance they hardly differ.

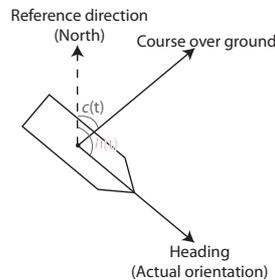


Figure 2.10: *The course over ground and heading of a vessel.*

In areas with dense shipping, area usage is established by rules. Sea lanes are mandatory for large vessels, with restrictions for draught and direction: high tonnage vessels sail in lanes far from the coast line. The bidirectional traffic is separated by two unidirectional lanes. Small vessels have to avoid contact with large vessels, vessels are not allowed to move slowly, and lanes should be crossed perpendicularly. In anchor zones vessels drop anchor while waiting for a tug to enter a harbour. Other area usages are oil platforms and wind parks; both should be avoided with reasonable distance during a sea trip.

## 2.6.2 Tracking of Vessel Traffic

Vessels are tracked either on board of a ship to avoid collisions or by authorities on the mainland to manage the vessel traffic. In this section we focus on the latter. For decades, vessels are tracked with radar systems, which emit electromagnetic waves to the environment and measure when the signal returns. The time between sending and receiving determines the position of an object. With additional processing it is possible to extract trajectories of the vessels. Nowadays radar systems are accurate, but they are unable to identify vessels. Also, the coverage is limited, since radar systems deployed at the shore line only track within the line of sight. Some of these shortcomings are solved with an advanced GPS device, which has become recently obligatory for large professional sea going vessels: the Automatic Identification System (AIS) [83]. The device broadcasts the vessel's status to the neighbourhood of the vessel. A captain is more aware of the situation around, using the statuses of neighbourhood vessels to prevent collisions. The status contains both vessel and trajectory information. Vessel information consists of identification numbers, a name, dimensions, and a type, e.g., passenger ship or tanker. Trajectory information consists of position, time, velocity, course, rate of turn, destination, expected time of arrival, draught, and navigational status, e.g., at anchor, moored, sailing, or fishing. A Very High Frequency (VHF) transmitter broadcasts trajectory information upto every two seconds depending on the speed and whether or not the vessel is changing course. Vessel information is broadcasted every six minutes. To avoid clashes in transmission, time-division multiplexing is used, which creates slots that can be assigned to a broadcast status of a vessel. As a result, in dense areas the transmission channel can in theory be overloaded, and the coverage can drop.

With a professional sensor network of antennas it is possible to record these statuses for a large area in real-time, as sketched in Figure 2.11. For the Netherlands such a network is operational and in 2010 there is almost full coverage of the Dutch coast. The covered area has a size of  $400 \times 400$ km and about half a gigabyte of AIS data is gathered from about five hundred vessels every day. Areas with no coverage are called “blind spots” and are worth to investigate. The data broadcast by a vessel to one antenna can be matched via the identification numbers with data from the same vessel but received by another antenna. From the matched data, we can infer whether a vessel went via the shortest route or had the opportunity to meet another vessel allowing it to exchange possibly illegal goods. From 2007, AIS is mandatory for sea faring vessels, hence we have a good view on the vessel traffic with this data source. Currently in 2011, progress has been made to also equip the river fleet with AIS. AIS data is also used by websites which deployed their own AIS antennas, such as Vessel tracker [154], The Marine Traffic project [141], VT Explorer [18], and FleetMon [84].

Not all the attributes in the AIS data are equally reliable. During installation fixed properties of the vessel are entered, such as the identification, e.g., name, IMO, and MMSI, and the size, e.g., width and height. While sailing some other properties are entered by the captain, such as the destination and the estimated time of arrival. Since these data are entered manually, human factors [71] occur, and as a result, the data entered may be missing, wrong, written in various languages, contain typing errors,

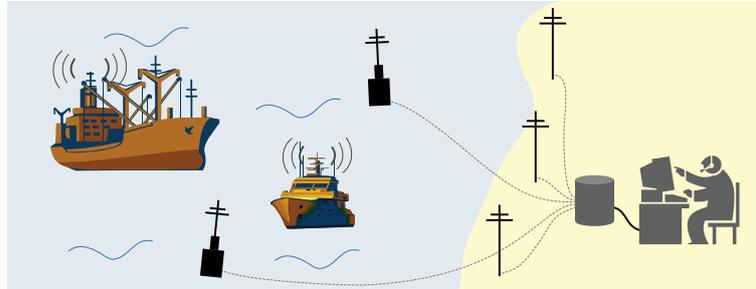


Figure 2.11: *The data gathering process for vessel trajectories using AIS.*

or is out of date. Typically, the properties manually set during sailing are unreliable. Other data is automatically captured by sensors, which gives more reliable data.

One of the sensors worth to mention is the GPS device [52], which determines its position based on the distance to at least three satellites for fixed altitudes, like ships have, or four satellites for variable altitudes. These satellites are at geostationary positions at an altitude of 35,786 km above sea level. Accurate, synchronized clocks in the satellite and GPS device enable to compute the distance based on the time shift between the sending and receiving. The distances between the GPS device and the satellites determines the location, which is often accurate up to a couple of meters, however, positional errors may occur, for instance due to atmospherical effects.

The sensor networks to gather AIS data were not intended to be part of the AIS system, however, live tracking is well received by operators. In the near future, another system will be used for live tracking, called Long Range Identification and Tracking (LRIT) [82]. The system uses satellites to gather the data and as a result the coverage will increase dramatically. LRIT only reports the position of ships a few times per day.

Like AIS for vessels there is a similar system for airplanes, called Automatic Dependent Surveillance-Broadcast (ADS-B) [119]. Airplanes broadcast every second their status with attributes like identifications, time, position, altitude, and position accuracy. ADS-B data is shown at sites such as FlightRadar24 [56] and Casper [33].

### 2.6.3 Trajectory Characteristics for Vessel Traffic

Vessel traffic data seems to be valuable input data for moving object analysis research in general. There are a number of reasons for that when looking at the characteristics as mentioned in Section 2.3. The AIS data as we use it (see Fig. 1.1 on page 5) is not negatively influenced by the size of the tracking device. Therefore, the data is accurate in both time and position, and as a result the actual movement is reconstructible with a small error. It is debatable if captains change their behaviour knowing that they are tracked, since they are responsible for a professional vessel and are expected not to make illegal movements. For pleasure craft it is more likely that the captains behave more according to the rules when tracked. The Dutch AIS data covers a large area and contains different types of objects with their own behaviour. For instance, it contains

both network-constrained objects, like tankers, and free moving objects, like rescue vessels. Since much research in moving object analysis is based on poorly sampled or noisy data, such as in animal research, we can focus with vessel data on the actual movement and do not need to bother ourselves with the interpretation of the data, which is a challenge on its own.

#### 2.6.4 Tasks for Vessel Traffic

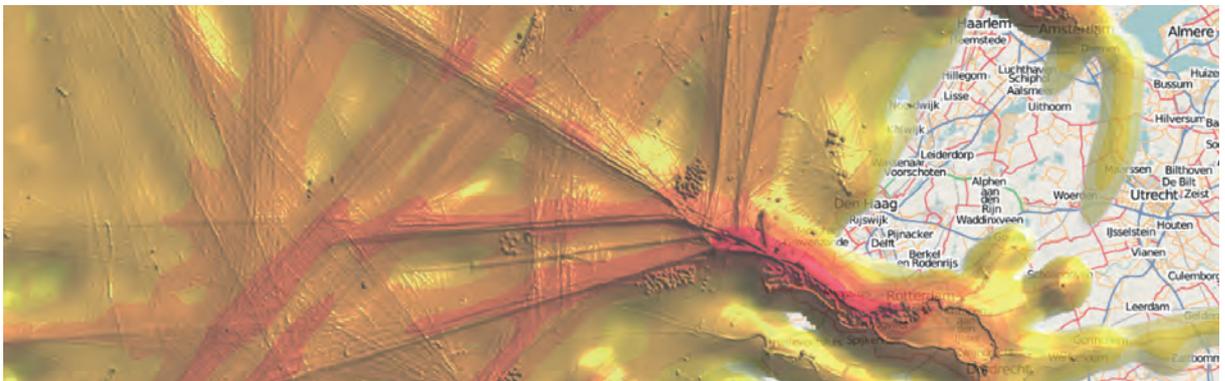
As mentioned in Section 2.4, there are a number of tasks for moving objects. In this section we discuss in more detail how these generic tasks are related to concrete examples for vessel traffic. For vessel traffic we clearly see the roles for operators and analysts; we find operators at positions managing current vessel traffic and analysts collect aggregated data for reporting and decision making.

Current vessel traffic is mainly managed by harbour authorities and the national coast guard. For both authorities the main task is to guide traffic to avoid accidents and secure the area for potential threats. In harbours the shipping movements are scheduled and monitored, and the harbour is secured against, for instance, attacks or illegal immigration. The coast guard mainly monitors for accident prevention and movements in forbidden areas, with the possibility for intervention if needed. In their work they make limited use of movement data. For instance, many interventions related to accidents origin from manual calls of neighboring ships and the administration of goods happens with waybills. Experienced operators are still key for handling traffic. However, there remain a few tasks related to movement data. First of all, operators need real-time tracking for finding ships moving in areas where they are not allowed. For instance, windmill parks and natural parks are forbidden. Secondly, stopping is not everywhere allowed, for instance in sea lanes. Thirdly, large vessels are obliged to sail via the sea lanes in the prescribed direction. Fourthly, vessels with dangerous cargo need to sail via sea lanes far from the coast. And lastly, vessels should only moor in harbours compatible with the vessel type. In fact, these tasks are elementary tasks, since for individual vessels we want to know if they obey the rules.

Investigations for decision making are done by institutes like the Dutch maritime research institute MARIN. Their main tasks are synoptic, however sometimes collisions between two vessels are analyzed for answering the question whether or not the accident could have been prevented, which is an elementary task. Other tasks are often related to quantitative analysis of vessel behaviour. For instance, a risk analysis is made for finding a safe area for new wind parks, where the question is raised: “Will there be an accident if a vessel makes a small deviation?”. The answer is based on large amounts of vessel movements; if no vessel entered the projected area, it is a safe area. Another task is related to reporting. Aggregated amounts of vessel movements tell about the available capacity of, for example, locks. In case the aggregate number is high in the area around a lock compared to its capacity, new developments are needed, like building an additional lock. All these tasks are synoptic, since many trajectories of vessels are needed to identify vessel behaviour and to draw eventually a conclusion.

# 3

## Vessel Density



*In this chapter we propose a geographical visualization to support operators of coastal surveillance systems and decision making analysts to get insights in vessel movements. For a possibly unknown area, they want to know where significant maritime areas, like sea lanes and anchoring zones, are located on a map. Our visualization is based on density fields that are derived from smoothing dynamic vessel positions with a kernel. The density fields are shown as illuminated height maps, which combines of two fields to provide both overview and detail. A large kernel provides an overview of area usage revealing vessel highways. Details of speed variations of individual vessels are shown with a small kernel, highlighting anchoring zones where vessels stop.*

### 3.1 Introduction

For surveillance purposes, AIS data is collected using a sensor network and send to operators, such as Coast Guard operators, who search for potential danger. Visual comparison of moving icons representing the current situation and map overlays with area usage help them to find outliers. We focus on two area usages at sea, namely sea lanes and anchor zones, and propose a method to extract them from historical trajectories.

Area usage can be computed using kernel density estimation, which convolves points with a kernel, resulting in smoothed data. The Dutch Hydrographic Office [108] published an example of smoothed vessel traffic, which is a density plot based on sparse human observations of vessels from an airplane during three years. The relatively small number of observations result in a global overview, but details are too noisy to be useful. Instead of manual point observations, AIS can be used as an automatic data provider, resulting in detailed trajectories as input to compute area usage. Instead of smoothing points, we then smooth the path of a trajectory. Additionally in our method, we take speed into account while smoothing the trajectories to show the dynamics of the movement. With a novel multi-scale density visualization different features are revealed at each scale: *vessel density*.

This chapter is organized as follows. A density model is defined in Section 3.2, and visualized in Section 3.3. Using real-world data we explore the capabilities of our method in Section 3.4. In the final section, we draw conclusions and suggest next steps.

### 3.2 Density

This section describes the first part of our method: a model for the contribution of a ship to the density in its neighbourhood as a result of sailing along a trajectory. In Section 3.2.1 a continuous density model is defined. Section 3.2.2 describes the computation of density from sampled trajectories. In Section 3.2.3, Euclidean space is left for geographical space to compute a geographic density overlay.

---

*This chapter is based the paper “Visualization of Vessel Movements” [162] presented at the Eurographics/IEEE Symposium on Visualization organized in Berlin, Germany in 2009*

### 3.2.1 Model

Consider an object  $o$  that moves in Euclidean space along a continuous path  $\mathbf{p}_o$  of a trajectory  $\tau_o$ , defined for points  $\mathbf{p}_o(t)$  for  $t$  in time window  $[0, T]$ . We model the contribution of  $o$  at a point  $\mathbf{q}$  in its environment, e.g., the weighted time duration  $o$  spends in the neighbourhood of  $\mathbf{q}$ . This contribution is modeled with a convolution where the signal at position  $\mathbf{s}$  is  $\delta(\mathbf{s} - \mathbf{p}_o(t))$  with  $\delta$  the Dirac delta function, and the response at position  $\mathbf{q}$  as a result of the object at position  $\mathbf{s}$  is given by the kernel  $k(\mathbf{q} - \mathbf{s})$ . The contribution at time  $t$  at position  $\mathbf{q}$  can be written as

$$C_o(\mathbf{q}, t) = \iint_{\mathbb{R}^2} \delta(\mathbf{s} - \mathbf{p}_o(t)) k(\mathbf{q} - \mathbf{s}) d\mathbf{s}. \quad (3.1)$$

Applying the sifting property then simply gives  $k(\mathbf{q} - \mathbf{p}_o(t))$  for  $C_o(\mathbf{q}, t)$ . The contribution  $C_o(\mathbf{q})$  in point  $\mathbf{q}$  taken into account the whole trajectory of  $o$  then results from integration over time:

$$C_o(\mathbf{q}) = \frac{1}{T} \int_0^T k(\mathbf{q} - \mathbf{p}_o(t)) dt, \quad (3.2)$$

where the factor  $\frac{1}{T}$  normalizes the contribution in time to enable comparison of contributions at different time scales. The total contribution of a set of objects  $O$  at  $\mathbf{q}$  is the sum of all individual contributions:

$$C(\mathbf{q}) = \sum_{o \in O} C_o(\mathbf{q}). \quad (3.3)$$

The contribution sampled with a cell  $Q$  can be represented by a density  $D$  given by the equation

$$D(Q) = \rho \iint_Q C(\mathbf{q}) d\mathbf{q}, \quad (3.4)$$

where  $\rho$  is a scaling factor for unit conversion, e.g., from a number of vessels per  $\text{km}^2$  to a number of vessels per  $1000\text{km}^2$ . The density is often investigated on a logarithmic scale  $\log(D(Q)+1)$ , to display different orders of magnitude. Figure 3.1 shows the effect of taking a variable speed into account for convolution along a line.



Figure 3.1: Convolution of a trajectory with a slow starting point and a fast ending point. (A) Point convolution. Line convolution at (B) constant speed and (C) accelerated.

### 3.2.2 Density Computation

In this section we discuss methods to compute density  $C_o(\mathbf{q})$  in Equation (3.2) for a single object  $o$  and a single line segment, for instance the first one  $\mathbf{p}_0\mathbf{p}_1$ , in a point  $\mathbf{q}$ . If a kernel is independent of position and orientation of the line segment, then the integral becomes invariant under rigid transformation. Without loss of generality the points can be written as

$$\mathbf{p}_0 = (x_0, 0) \wedge \mathbf{p}_1 = (x_1, 0) \wedge \mathbf{q} = (0, y) \text{ with } y \geq 0 \quad (3.5)$$

and the integral reduces to

$$C(\mathbf{q}) = \frac{1}{T} \int_{t_0}^{t_1} k((-x(t), y)) dt. \quad (3.6)$$

This integral is numerically approximated. For kernel  $k$  we take a cone kernel, which is an isotropic kernel with finite support and radius  $r$  where  $C(\mathbf{q}) = 0$  for all  $\mathbf{q}$  with  $\|\mathbf{q}\| > r$ . The computational costs of  $C(\mathbf{q})$  can be reduced, since for a line segment  $\mathbf{p}_0\mathbf{p}_1$  partially within distance  $r$  of  $\mathbf{q}$ , tighter integration bounds  $t_i$  exist. Let  $X$  be the interval on the  $x$ -axis where the distance to  $\mathbf{q}$  is at most  $r$ . If  $X \cap [x_0, x_1] = \emptyset$  the integral is zero, otherwise  $X \cap [x_0, x_1] = [a_0, a_1]$  for some  $a_i$  and the new integration bounds  $t_i = t(a_i)$  are given by Equation (2.8) on page 15.

### 3.2.3 Geographic Implementation

We move from an Euclidian coordinate system to a geographic coordinate system, to investigate movements of objects on the earth. Density is computed geographically, where we distinguish three kinds of spaces (Fig. 3.2): grid space, geographic space, and image space. To compute density using the input data for equal sized cells, we regularly sample at equal distance in grid space, and hence irregularly in longitude  $\lambda$  and latitude  $\phi$  in geographic space. The data and grid are displayed as an image by transforming all geographic coordinates to a map using a map projection.

#### Rasterization

The density in Equation (3.4) is computed in a grid of cells, which represent equally sized blocks on the earth with a scalar value. An equal area map projection prescribes

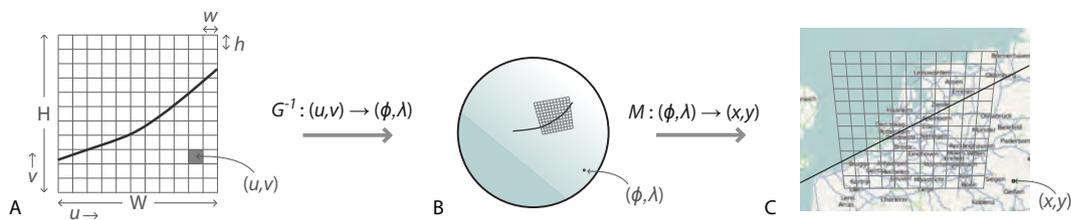


Figure 3.2: Computation in (A) grid space, (B) geographic space, and (C) image space.

## Density

33

a relation between positions on the earth and a regular  $(u, v)$ -space, such that the area on the earth of a shape in  $(u, v)$ -space stays invariant under any translation in  $(u, v)$ -space. We use the sinusoidal equal area projection  $S^{-1}(u, v)$  [135]

$$\phi = v \quad (3.7)$$

$$\lambda = \lambda_0 + u / \cos(v). \quad (3.8)$$

with  $\lambda_0$  as meridian. The sinusoidal projection is a combination of great circle angles in the  $v$  dimension and small circles in the  $u$  dimension. A factor  $\sigma(\phi)$  given by

$$\sigma(\phi) = \frac{2\pi R}{360^\circ} \cos(\phi), \quad (3.9)$$

defines the length on the surface of the earth per degree, along a small circle at latitude  $\phi$ . Notice that for  $\phi = 0$ , we get the length per degree for a great circle. To get intuitive parameters, the  $(u, v)$ -space is scaled horizontally with  $w/\sigma(0)$  and vertically with  $h/\sigma(\phi)$  resulting in cells of  $w$  km wide and  $h$  km high. In this space we define a grid of  $W$  km wide and  $H$  km high. To minimize the distortion of the rectangularity of the cells we introduce a standard parallel  $\phi_0$  and transpose the center of the grid to the origin. This results in an adapted sinusoidal projection  $G^{-1}(u, v)$

$$\phi = \phi_0 + \frac{h}{\sigma(\phi_0)}(v - \frac{1}{2}H) \quad (3.10)$$

$$\lambda = \lambda_0 + \frac{w}{\sigma(\phi)}(u - \frac{1}{2}W). \quad (3.11)$$

The sinusoidal projection is chosen for efficiency reasons. Since every cell needs to be projected, the number of expensive function calls, such as trigonometric functions and square roots, needs to be small. Other common equal-area map projections, such as Bonne, Molleweide, Albers, and Lambert’s, are computationally more expensive.

## Geographical Mapping

A real-world trajectory of a non-route bound object consists of a sequence of rhumb line segments (see Sec. 2.6.1) instead of line segments. The computation of vessel density is implemented in Algorithm 1. For each rhumb line segment, the density as displayed in Equation (3.2) is computed for all grid cells. For finite-support kernels only the cells in the Oriented Bounding Box (OBB) at distance  $r$  to the rhumb line segment need to be considered.

To compute the density of a rhumb line segment  $\widehat{\mathbf{p}_0\mathbf{p}_1}$  for cell at position  $\mathbf{q}$ , we assume the kernel small to not take the curvature of the earth into account within the kernel. Therefore the integration in geographic space uses tight integration bounds for  $C_o(\mathbf{q})$  in Equation (3.6) where  $x_0$ ,  $x_1$ , and  $y$  are given by

$$x_0 = \|\widehat{\mathbf{p}_0\mathbf{p}_k}\| \wedge x_1 = \|\widehat{\mathbf{p}_k\mathbf{p}_1}\| \wedge y = d_{gc}(\mathbf{p}_k, \mathbf{q}). \quad (3.12)$$

with  $\mathbf{p}_k$  on  $\widehat{\mathbf{p}_0\mathbf{p}_1}$  with smallest great circle distance to  $\mathbf{q}$ .

---

**Algorithm 1** Vessel density()

---

```

 $D \leftarrow \mathbf{0}_{H,B}$  {Density D becomes zero matrix}
for  $o \in \text{Objects}$  do
  for tuples  $\alpha_i$  and  $\alpha_{i+1}$  in trajectory  $\tau_o$  do
     $\mathbf{p}_0 \leftarrow \mathbf{p}_o(t_i); \mathbf{p}_1 \leftarrow \mathbf{p}_o(t_{i+1})$ 
     $OBB \leftarrow$  oriented bounding box at distance  $K$  to  $\widehat{\mathbf{p}_0\mathbf{p}_1}$ 
    for  $(u, v) \in OBB$  do
      {Get geographical coordinate of the cell center}
       $\mathbf{q} \leftarrow G^{-1}(u, v)$ 
      {Assign density based on Equation (3.2)}
       $D(u, v) \leftarrow D(u, v) + C_{\widehat{\mathbf{p}_0\mathbf{p}_1}}(\mathbf{q})$ 
return  $D$ ;

```

---

### 3.3 Visualization

A density  $D$  computed in Algorithm 1 can be considered as a height field to be visualized on a map using classical cartographic methods [29, 124], such as color mapping with a continuous color map (Fig. 3.3A) or a discrete color map (Fig. 3.3B), which implicitly shows contours. For quantitative analysis, a contour plot is preferred over a continuous color map, since value estimation by colors is perceptually hard [99]. A fancy way of creating contour plots is described by van Wijk and Telea [152]. In our

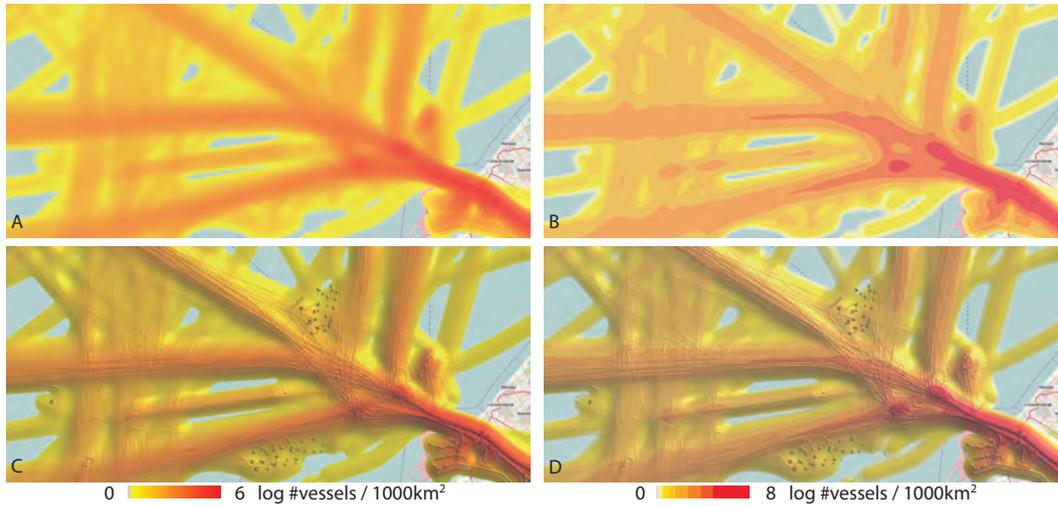


Figure 3.3: Vessel density convolved with a kernel of 1.5km in front of the harbour of Rotterdam of a single day. The multi-scale density is decorated with convolved trajectories with a kernel of 100m. (A) Density with a continuous color map. (B) Density with a discrete color map. (C) Multi-scale density with a continuous color map. (D) Multi-scale density with a discrete color map.

case we have chosen the class intervals of the discrete color map to show the orders of magnitude of the density by using a logarithmic scale. Improvements for the choice of class intervals are described by Andrienko *et al.* [6].

These classical approaches provide an overview of area usage, but the absolute temporal aspect of trajectories disappears, since in the smoothing we integrate over time. By using the velocity in the smoothing another temporal aspect pops up, namely the average duration that an object stays at a location. The kernel size influences the variation of the durations shown in the density, for a small kernel the variation is high, while for a large kernel the variation is low. As a result the features visible in the density differ between kernel sizes. For small kernel sizes we see the variation of the speed of individual objects, because trajectories are not spread to a large neighbourhood and hence not many trajectories are aggregated together. For large kernel sizes the variation in speed is low and the number of aggregated trajectories is high, therefore more global patterns appear such as common routes.

To get both an overview and a detail view, we separately compute a density  $D_{\text{large}}$  with a large kernel and a density  $D_{\text{small}}$  with a small kernel and display them simultaneously in the same area. To speed up the computation,  $D_{\text{large}}$  is computed on a lower resolution than  $D_{\text{small}}$  without noticeable loss of features. The density resolutions are synchronized by resampling  $D_{\text{large}}$  using bilinear interpolation.

Our visualization simultaneously displays both densities by shading the color mapped  $D_{\text{large}}$  according to a height map  $H$  of the accumulated densities (Fig. 3.3C and 3.3D). Consider a cell  $Q$ , then the color  $C(Q)$  is defined as

$$H(Q) = \alpha D_{\text{large}}(Q) + \beta D_{\text{small}}(Q) \quad (3.13)$$

$$C(Q) = I_p(H(Q)) \text{ColorMap}(D_{\text{large}}(Q)). \quad (3.14)$$

The intensity of the shading  $I_p$  is obtained with Phong illumination [116] of  $H$ , using one white light source at infinity. A schematic overview of the complete rendering pipeline of vessel density is given in Figure 3.4. The grid is georeferenced on a map using a map projection. To speed up rendering, corners of a rectangle of cells that are almost flat are map projected and the cells in between are bilinearly interpolated.

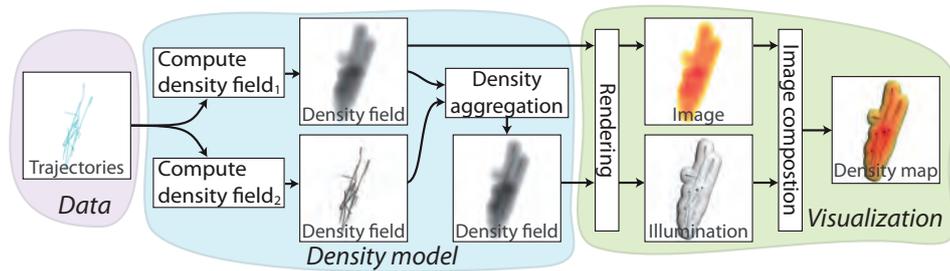


Figure 3.4: The rendering pipeline of vessel density. From left to right, trajectories are smoothed with a large and a small kernel resulting in two density fields. In the rendering, the large kernel density field is used for color mapping and the aggregated density fields are used for the illumination. The last step multiplies the color image and the gray-scale image with the illumination.

## 3.4 Real-world Data

In this section we explore the capabilities of our method with data from the real world. We adapt a simplification method to reduce the size of data sets with attributed trajectories. Furthermore, we conduct two case studies, and discuss advantages.

### 3.4.1 Trajectory Simplification

We simplify trajectories for the position and speed dimensions with an adapted version of the Douglas-Peucker (DP) line simplification algorithm [42], which is known for its deterministic error bounds in the simplified line.

The original DP line simplification removes points within a given position error bound  $\epsilon \geq 0$ . It is a recursive algorithm for a pair of point indices  $(i, j)$  that computes an index  $k \in [i, j]$  such that  $\mathbf{p}_k$  has the largest distance to line segment  $\mathbf{p}_i\mathbf{p}_j$  and checks if the distance  $d(k, i, j)$  of  $\mathbf{p}_k$  to line segment  $\mathbf{p}_i\mathbf{p}_j$  is smaller than  $\epsilon$ ; if so, the line segment is accepted, otherwise the procedure repeats for  $(i, k)$  and  $(k, j)$ . We use this same scheme twice to first accept points based on their speed and subsequently simplify the line segments between the accepted points based on *geographic* coordinates; to do so we have two version of the distance function  $d$  and two error bounds  $\epsilon$ .

A good reconstruction includes points with extreme speed in a simplified trajectory. The following distance is compared to error  $\epsilon_v$  in the speed reconstruction:

$$d(k, i, j) = |\dot{x}_{i,j}(t_k) - v_k| \quad (3.15)$$

where  $x_{i,j}$  is given by Equation (2.3) on page 14 after replacing subscript 0 by  $i$  and 1 by  $j$ . For all resulting accepted point indices  $(i, j)$  we spatially simplify the path  $\mathbf{p}_i \dots \mathbf{p}_j$  within error  $\epsilon_p$  using the DP scheme but with distance given by

$$d(k, i, j) = d_{gc}(\mathbf{p}_k, \mathbf{r}(k, i, j)) \quad (3.16)$$

where  $\mathbf{r}(k, i, j)$  is the simplified position of the object on  $\widehat{\mathbf{p}_i\mathbf{p}_j}$  at time  $t_k$ :

$$\mathbf{r}(k, i, j) = RL(\mathbf{p}_i, \mathbf{p}_j, x_{ij}(t_k)/x_{ij}(t_j)). \quad (3.17)$$

with  $RL$  the natural parametrization of  $\widehat{\mathbf{p}_i\mathbf{p}_j}$  (see Sec. 2.6.1 on page 23).

Table 3.1 shows the amount of ignored data for one data set of one day consisting of 1460 vessel trajectories. The straight line nature of vessel trajectories yields a reduction of 79% for small errors, while speed reconstruction is preserved.

### 3.4.2 Application

We have implemented vessel density in a custom build geographical information system, which allows drawing on an OpenStreetMap map [110]. Our prototype is developed in Java with OpenGL (JOGL), and tested on a machine with an Intel Core2 Duo T7300 (2.0 GHz) processor, 2 GB RAM, NVidia Geforce 8600M GT, and Windows Vista (32 bits). Figure 3.5 can be computed in 10 minutes per day of data, containing 100,000 rhumb line segments, resulting in an overlay of  $2000 \times 2000$  pixels covering an area of  $400 \times 400$  km. In this section we apply our method with two use cases.

	(%)	$\epsilon_p$ (m)		
		50	100	250
$\epsilon_v$ (knot)	0.5	79.0	80.4	80.8
	1	83.2	84.9	85.3
	5	91.9	94.0	94.8

Table 3.1: Amount of ignored data with our trajectory simplification for one day of vessel traffic.

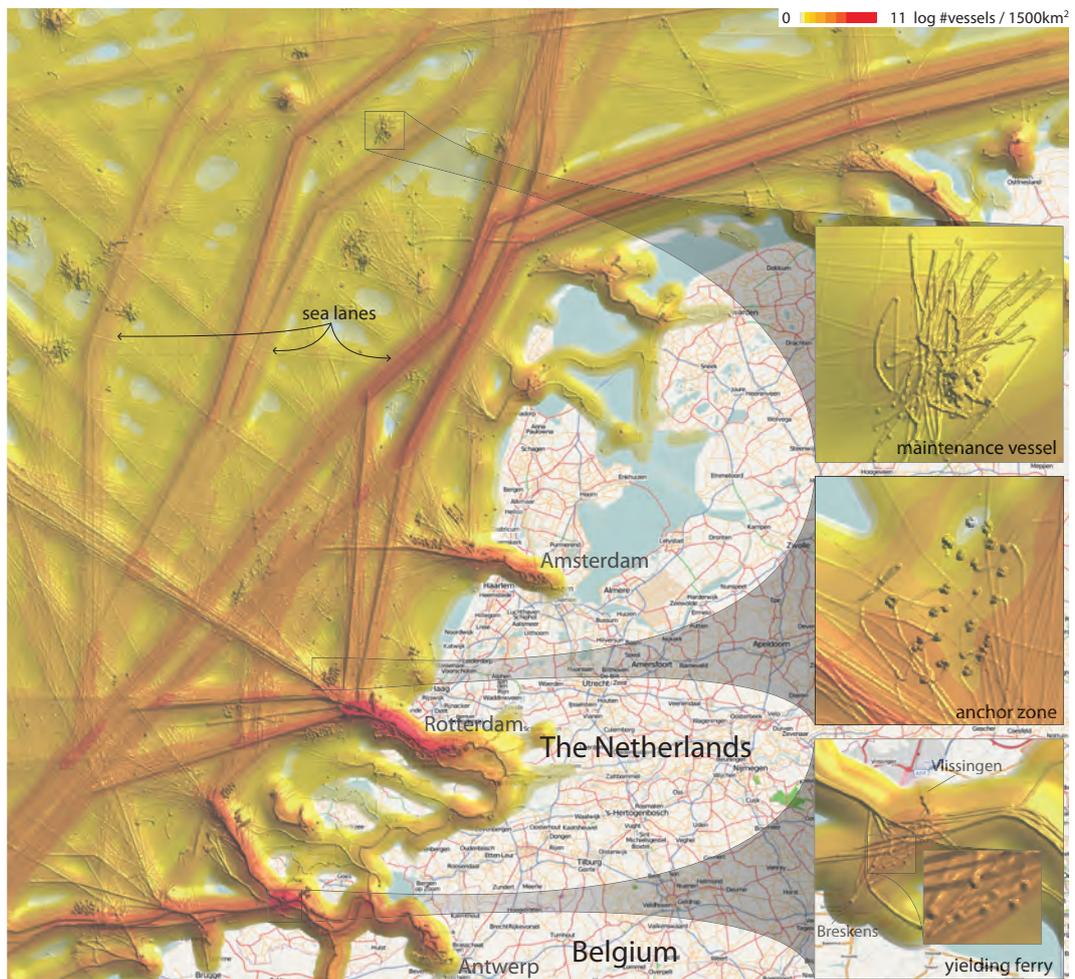


Figure 3.5: Vessel density of the Dutch coast: Trajectories of a week covering 160,000km<sup>2</sup>, and convolved with kernels of 5km and 200m. The picture represents 3.5GB of data. The anchor zone and yielding ferry inset are renderings of a day using kernels of 1.5km and 100m.

### Weather

The base case is investigation of shipping movement during smooth weather. In Figure 3.3 we show the entrance of Rotterdam harbour, where anchor zones pop up by the highlighted dots of the individual trajectories. On a large scale the same method results in similar patterns. In Figure 3.5 we see that three North-South sea lanes appear. Furthermore, some maintenance vessels move slowly in a small area, typically around an oil platform or a wind mill park, which can be observed by the intense individual trajectories. Lastly, in the South the ferry between Vlissingen and Breskens yields regular traffic which can be observed by intense dots just before the sea lane.

Shipping movements differ during a gale. In Figure 3.6, a North-West gale of force 8 on the scale of Beaufort appears during the day. In general, there are fewer shipping movements and sea lanes are less used compared to Figure 3.3D. Captains try to keep course, but from the twisty individual trajectories we see that they drift slightly. Furthermore, anchor zones are less often used, and vessels slowly sail along the wind towards the anchor zones to avoid rolling, as can be observed by intense left-to-top to right-bottom trajectories.

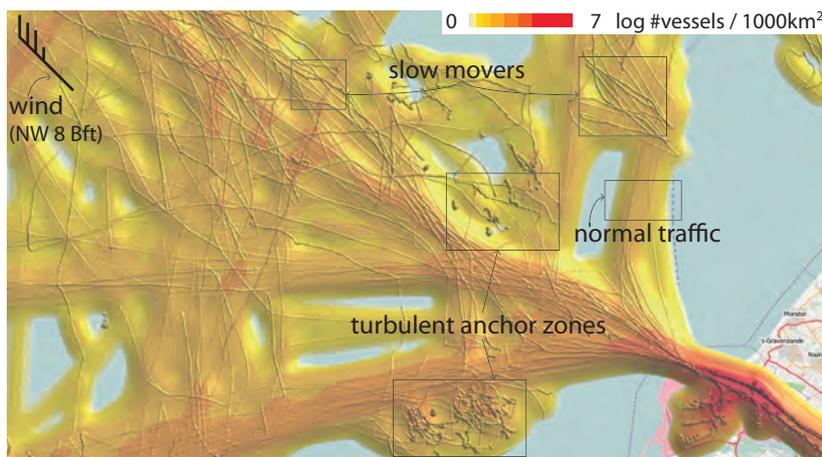


Figure 3.6: *Vessel density of a stormy day: North-West wind with force 8 on the Beaufort scale change the movement pattern of vessels.*

### Slow Movers

Slow movers in sea lanes are dangerous and are therefore not allowed. In this use case we investigate if they appear anyway. In Figure 3.7 the large kernel vessel density is displayed only for trajectory segments where vessels move less than three knot. The density is displayed using a blue-red-blue color map to avoid too much attention to high density. This is done by setting a threshold for the highest density value shown in red and interpolating back to blue for higher density values. As a result, high density

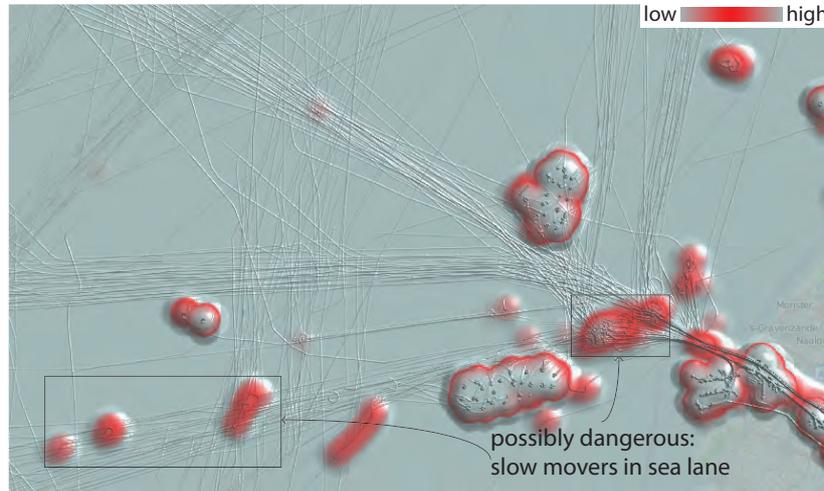


Figure 3.7: *Vessel density of areas where vessels sail less than 3 knot during smooth weather.*

areas are encircled with soft contours. A notable pattern in Figure 3.7 appears in the lowerleft corner where slow movers appear in a sea lane.

### 3.4.3 Discussion

A simple approach to compute densities is to convolve trajectories assuming a constant speed of all vessels, whereas in our approach speed is taken into account. In Figure 3.8 these two approaches are compared. We notice three differences when velocity is taken into account. First of all, slow movers are highlighted as shown in the anchor zones. Secondly, the North-South sea lanes differ, which means that the probability to meet a vessel in the right lane is higher than in the left one. Lastly, our model fades distracting, large positional sensor errors as shown in the green box. In the left picture we see a trajectory that is hidden in the transparent lower values of the color map in the right. Since in the reconstruction of the trajectory the begin and end speed are adapted to the distance, the vessel hardly contributes for far erroneous positions.

In general, our density shows speed variations with vessel data, but the slow and steady nature of vessel movements does not result in clear acceleration patterns. However, in other application domains more dynamic data sets exist, such as car, airplane, or mouse trajectories, in which acceleration patterns are expected to show up.

#### Brownian Bridges

The density model as presented in this chapter is a simplified version of the Brownian bridge [75] model. In this model the probability of the occurrence of an object at a certain place is modeled, given the location and the uncertainty of the end points and given the maneuverability of the object. In between the end points no movement is

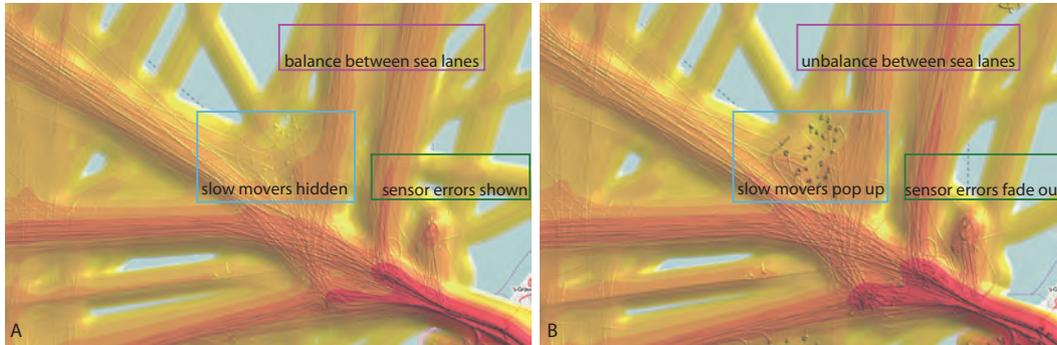


Figure 3.8: Comparison of convolution with (A) 15 knot fixed speed and (B) velocity adjusted vessel density. The colored boxes annotate some of the differences between the convolution methods.

measured, instead a Brownian motion is assumed, which is convolved with a Gaussian kernel. The larger the difference between the distance from end point to end point and the largest possible displacement given the location and time around the end points, the more freedom the object has to move. As a result, a trajectory segment is smoothed with a Gaussian, however the kernel size varies along the segment. The formulas tend to be similar in shape, however the basic differences are that our reconstruction of the trajectory path  $\mathbf{p}$  is a single trajectory where speed is taken into account and not a statistical process.

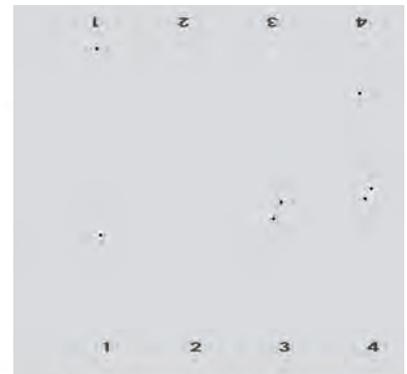
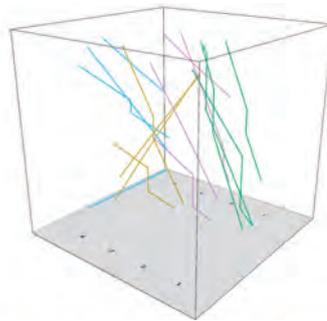
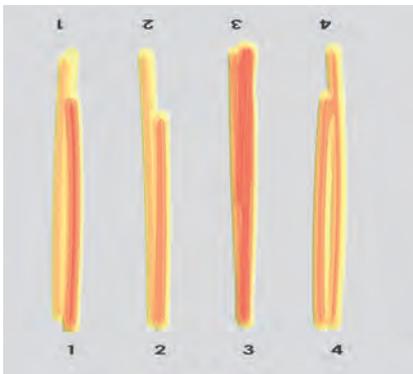
### 3.5 Conclusion and Future Work

We have presented a novel method to compute and visualize moving objects in such a way that speed is taken into account. In order to do so, we have defined a convolution method inspired by kernel density estimation, which returns the contribution per moving point to the overall density. Furthermore, we created an overview + detail visualization using a multi-scale approach for simultaneously displaying two densities. Each density field, one with a small and one with a large kernel, shows different features. Velocity adapted convolution is applied in the maritime domain showing maritime significant areas, such as sea lanes and areas where vessels move slowly or stop. Lastly, our method is applied to a data set collected during abnormal weather conditions, resulting in drastically different patterns.

For future work we suggest to improve the performance, for instance by using graphics hardware and simplifications in the density model, such as computations fully in Euclidian space. Besides, there is a need for interactive tools to enable comparison of various sets of densities, such as for multiple times of day, multiple weather conditions, or multiple ship types. One of the possible solutions for this is described in Chapter 5. Furthermore, we would like to see the reliability of the data, since AIS may not have full spatial coverage. Finally, we would like to investigate vessel density with trajectories from other domains.

# 4

## Vessel Density Evaluation



*In this chapter we conduct a user study where we test the performance of vessel density as presented in Chapter 3 in comparison to well-known trajectory visualizations, such as an animation of moving dots and the space-time cube. We test these visualizations with common maritime analysis tasks by investigating the ability of users to find stopping objects, to find fast moving objects, and to estimate the busiest routes in vessel trajectories. We test the robustness of the visualizations with respect to scalability of both the number and complexity of the trajectories. The test shows that each visualization type excels for correctness for a specific movement feature. Vessel density performs best for finding stopping objects, but does not perform significantly less than the remaining visualizations for the other features. Therefore, vessel density is a nice extension in the toolkit for analyzing trajectories of moving objects, in particular for vessel movements.*

## 4.1 Introduction

For decades, object movements are analyzed to find movement features with different approaches, such as geometric algorithms [5], reasoning [164], clustering [8], and visualization [173]. We focus on visualization as a means to extract vessel movement features in trajectories, and measure the performance among different visualizations.

Many types of visualizations can be used to display moving objects, and most of them have in common that they are an aggregated view on the  $xyt$ -space as defined by Andrienko and Andrienko [15]. We evaluate the performance of a dot animation, the space-time cube, and vessel density [162]. In terms of aggregated views, an animation of dots moving along the position of the objects is an aggregated view of all current locations  $(x, y)$  for varying times  $t$ . The other two visualizations are an aggregated view showing the complete trajectories, either in a 3D representation with separation of time for a space-time cube, or in a single plane aggregating time for vessel density. The performance of a view to determine whether or not a certain movement feature occurs in the visualized trajectories is expected to vary between different views.

For common maritime tasks in the analysis of vessel trajectories, we are interested in the performance of vessel density with respect to other well-known trajectory visualizations. The response time and correctness is investigated for solving tasks matching with the following vessel movement features: *co-location in space*, *stops*, and *fast movement*. In vessel traffic monitoring, co-located ships appear in sea lanes and stops occur in front of harbours. Stops should be noticed in sea lanes to avoid dangerous situations. Fast movement at sea may indicate a potential threat of, for instance, intruders. Furthermore, real-world data consist of many, noisy, trajectories that are tracked with slightly inaccurate measuring devices. Therefore, we also test for the influence of trajectory complexity and scalability with respect to the number of trajectories of these

---

*This chapter is based the paper “Evaluating the Visibility of Vessel Movement Features in Trajectory Visualizations” [163] presented at the Eurographics/IEEE Symposium on Visualization organized in Bergen, Norway in 2011.*

visualizations. To our understanding these properties have not been tested before.

The chapter is organized as follows: First, we elaborate on existing work in Section 4.2. In Section 4.3, we describe how a single trial in the experiment consisting of data and visualizations is created. In Section 4.4, we explain how trials are composed into a user test and explain how this test is executed. The test results are analyzed in Section 4.5. Finally, in Section 4.6, we draw conclusions and suggest future work.

## 4.2 Related Work

In this section we elaborate on existing user studies conducted to the two alternative visualizations for vessel density, space-time cube, and animation.

### 4.2.1 Evaluation of the Space-time Cube

Demissie [39] evaluates the space-time cube against static maps by means of a single map with polylines or small multiples with polylines, and an animation of dots moving along a polyline. Upto four trajectories from pedestrians walking in a city are analyzed. Subjects count the following movement features: stops, returns along a same path, returns to a same location, and speed changes. Based on 16 participants they conclude that animation excels for most features, since it captures the temporal dynamics best.

Patterns of pedestrian movements are evaluated by Kristensson *et al.* [93] in a space-time cube and a map with polylines annotated with labels containing timestamps. Subjects solve spatio-temporal queries with various levels of complexity for upto four trajectories. Based on the data captured from 30 participants, they conclude that the space-time cube is more prone to error for simple queries with equal response time, while for complex queries the response time for the space-time cube is better. In future work they suggest to test the space-time cube for data with higher density.

In virtual environments, the space-time cube can be rendered in either strong or weak 3D. Kjellin *et al.* [89] conduct a user study to see the effect between these rendering techniques for finding the largest spread between various disease propagation. The test is executed by 32 participants, from which they conclude that there is no significant difference between the two rendering techniques.

The space-time cube paradigm can also be used for displaying sequences of events by using abstract  $xy$ -axes, such as participants and activities. In a user study conducted by Vrotsou *et al.* [155] the effect of 2D versus 3D representations is studied for different camera positions of the space-time cube: front view (2D), side view (2D), and rotation along a single axis as interaction (3D). The task is to find simultaneously occurring events in different sequences. Based on the data of 20 participants, they argue that 3D works significantly better than 2D, not only for simple tasks containing a small number of sequences, but also for more complex tasks containing more sequences. Moreover, they expect that 2D results in a memory overload for complex tasks.

Our test differs from these studies, since we investigate properties of real-world data, such as a scalability and complexity, and, furthermore, we investigate other movement features.

## 4.2.2 Evaluation of Animation

Animated graphics can be used to show, among others, state transitions, view transitions, explanatory graphics, and trends in data. By animating the movement of objects, we may understand the situation and judge whether or not the movements were normal. Many user studies comparing the effectiveness of animations and static representations have been conducted. Tversky *et al.* [146] conducted several studies that concern animations used in teaching complex systems, and concluded that animation has not been proven to be more effective. They suggest the *congruence principle* and the *apprehension principle* to come to at least successful animated graphics. Gonzalez [60] evaluates the effectiveness of animation in decision making and suggests to use realistic graphics, smooth transitions, and an appropriate interaction style. In their study of effectiveness of animation in trend visualization Robertson *et al.* [123] compare animated bubble charts to two static visualizations and find that animations are both more time consuming and less accurate. On the other hand they find that the users preferred working with animations.

## 4.3 Scene Design

We define a scene to be a combination of a data set and a visualization. In Section 4.3.1 we explain how we can generate variations of trajectory data and in Section 4.3.2 we elaborate on various well-known trajectory visualizations.

### 4.3.1 Data Generation

In order to test subjects we need data that triggers for the phenomena we want to test. In our case, we want to trigger the subject to differentiate between variations for a certain movement feature with vessel traffic data. With real-world data it is hard to control these variations, therefore we generate synthetic data using a model that allows us to modify the data according to these variations. We aim for realistic synthetic data, which leads to the following requirements, based on observations of vessel traffic:

- vessels typically do not travel along the same path;
- vessels hardly occur at the same time and position;
- vessels often move in straight one-way sea lanes;
- a cross-section of a sea lane has a normal distribution,
- vessels move with a nearly constant speed.

We generate data with these requirements and introduce variables with some fixed settings to vary the number of trajectories  $\mathcal{C}$ , the visual clutter  $\mathcal{N}$ , and different movement features  $\mathcal{M}$  within a scene. In the next sections we describe the generation process to obtain data for one scene, which consist of four lanes where ships are moving back and forth. First, we describe a model to generate a trajectory, then we model a lane, and, finally, we model a set of lanes. These models are parameterized in such a way that we can generate the input data for a scene with the desired settings for  $\mathcal{C}$ ,  $\mathcal{N}$ , and  $\mathcal{M}$ .

### Trajectory Model

A trajectory  $\tau$  models the movement of an object and maps a clock time to a position in two-dimensional screen space. We model a trajectory with several parameters that are varied later on to create lanes, sets of trajectories. First we model the geometry of a ‘horizontal’ trajectory as a parametric function in  $u \in [0, 1]$  representing half a period of a sine:  $(u, B \sin(\pi u))$ , with bending parameter  $B$ . We modulate this trajectory along its normal with an offset function  $o(u) = A \sin(\omega u)$  to control the visual clutter with parameters amplitude  $A$  and frequency  $\omega$ . To get trajectory geometry with given start point and end point, an affine transformation is applied.

In a second step, we add timing information to the geometry. Assuming that we have a natural parametrization  $\mathbf{p}(s)$  of the geometry and given a fixed velocity parameter  $v$  and a start time  $t_0$  the timing of the curve is given by  $\mathbf{p}((t - t_0)v)$ . This can easily be extended for the additional timing parameters we require in this model: a starting position given by a time  $t_1$ , a stop time  $t_2$ , and a stop duration  $\Delta t$ . A stop occurs just once in a trajectory to avoid ambiguity later on in the task of counting stopping objects.

For generating either simple trajectories or complex trajectories we have defined two settings for visual clutter  $\mathcal{N}$ : for simple data  $\mathcal{N}_{\text{simple}} \equiv \{B = 0.02, A = 0, \omega = 0\}$  and for complex data  $\mathcal{N}_{\text{complex}} \equiv \{B = 0.05, A = 0.04, \omega = 2\}$ .

### Lane Model

A lane is a set of trajectories and takes the following parameters: the number of trajectories  $N$ , the number of stopping objects  $S$ , the average speed  $V$  of all objects, a start point  $\mathbf{p}_0$ , and an end point  $\mathbf{p}_1$ . A lane is created by generating  $N$  trajectories according to the settings for  $\mathcal{N}$  and with normally distributed start and end points with  $\mu = \mathbf{p}_0$  and  $\mu = \mathbf{p}_1$ , respectively, and a fixed variance  $\sigma_{\mathbf{p}} = (0.01, 0.01)$  and ignoring points at a distance larger than  $2\sigma$ . The begin time  $t_0$  of a trajectory is normal distributed with  $\mu = 0$  and with variance  $\sigma_t$  varying between 3 and 10, depending on the movement feature, which is described in the next paragraph. A uniformly distributed time  $t_1$  is chosen for some of the trajectories that will not start in  $\mathbf{p}_0$ . The speed is chosen with a normal distribution with  $\mu = V$  and small variance  $\sigma_v$  to obtain a more realistic movement profile within a lane. For  $S$  of these trajectories stopping time  $t_2$  is chosen with a uniform distribution in the center 80% of the trajectory. The stop duration is chosen uniformly distributed between 4% and 15% of the total duration of the trajectory. The stopping parameters are chosen according to a pilot study to tune for clear visibility of the stop features, in particular in a space-time cube containing many trajectories.

### Lane Composition

A scene consists of four lanes. The begin points and end points of lane  $l$ , with  $0 \leq l < 4$ , are given by  $\mathbf{p}_0 = (l/3, 0)$  and  $\mathbf{p}_1 = (l/3, 1)$ , resulting in trajectories in a vertical configuration. All lanes have the same generation parameters, except for one parameter. Depending on the movement feature  $\mathcal{M}$  we fix this parameter for two lanes, the base level  $b$ , and differ the other two by a factor  $\gamma$ ; their parameters are  $b\gamma$  and  $b/\gamma$ . We have

chosen  $\gamma = 1.5$  to obtain subtle, but noticeable, differences between lanes. Furthermore, the order of the lanes is randomized. In table 4.1 we give a mapping between movement feature  $\mathcal{M}$  and generation parameters for lanes: number of trajectories  $N$ , the number of stopping objects  $S$ , the average speed  $V$ . This mapping contains two properties, the number of trajectories per lane  $\#\tau$ , and a basic speed  $v$ . The concrete number for  $\#\tau$  is given in the next section for generating a scene. The basic speed  $v$  is one-tenth of the distance between the endpoints of the lane per second, resulting in trajectories lasting approximately ten seconds.

Parameter	$\mathcal{M}_{\text{stop}}$	$\mathcal{M}_{\text{lanes}}$	$\mathcal{M}_{\text{fast}}$
$N$	$\#\tau$	$\#\tau$	$\#\tau$
$S$	$\#\tau/\gamma$	0	0
$V$	$v$	$v$	$v$

Table 4.1: An overview of the lane generation parameters for various movement features. Blue colored values in the cell are base level settings and are varied between lanes.

### Input Data

In the previous sections, the model is defined to generate data for one scene. We can create variations of scenes with the settings of parameters  $\mathcal{C}$ ,  $N$ , and  $\mathcal{M}$ . The number of trajectories  $\mathcal{C}$  is varied to test for busy lanes. We have defined settings  $\mathcal{C}_{\text{sparse}}$  and  $\mathcal{C}_{\text{dense}}$ , with  $\#\tau = 4$  and  $\#\tau = 15$  trajectories respectively.  $\mathcal{C}_{\text{sparse}}$  is chosen to trigger pre-attentive vision [70], while  $\mathcal{C}_{\text{dense}}$  forces a possible cognitive overload as a result of a high density of trajectories at one location.  $\mathcal{C}_{\text{dense}}$  has a similar local density as real-world data as we encountered in the maritime domain.

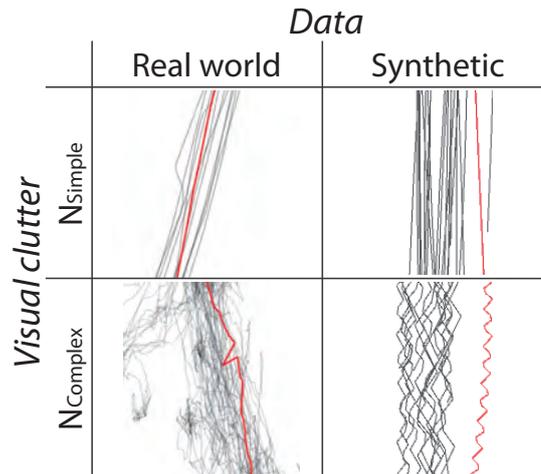


Figure 4.1: Real-world versus synthetic movement data, with visual clutter resulting from simple (ships in a sea lane) or complex (pedestrians in a street [148]) trajectories.

For visual clutter  $\mathcal{N}$ , we distinguish simple trajectories generated using  $\mathcal{N}_{\text{simple}}$  and complex trajectories using  $\mathcal{N}_{\text{complex}}$ . Simple trajectories are associated with inert large tankers that tend to move along straight lines. Complex trajectories appear in ship traffic during stormy weather, but also in animal and pedestrian movements. Combined complex trajectories result in entangled trajectories with more visual clutter. Figure 4.1 shows how our data generation compares to real-world data for visual clutter with vessels ( $\mathcal{N}_{\text{simple}}$ ) and pedestrians ( $\mathcal{N}_{\text{complex}}$ ).

Trajectories should include a certain movement feature  $\mathcal{M}$  to create sensible tasks. The movement features are limited to those occurring in vessel traffic. We aim to evaluate the claims about strong properties of vessel density: stoppers  $\mathcal{M}_{\text{stop}}$ , fast movers  $\mathcal{M}_{\text{fast}}$ , and lanes  $\mathcal{M}_{\text{lanes}}$ . Since vessel density does not support showing direction, we have excluded related movement features. Vessel traffic is composed of many lanes. We do not address this complexity in our study, but focus on single lanes, to keep the experiment manageable. However, we argue that understandability of vessel traffic in single lanes is a prerequisite for understanding more complex scenarios.

### 4.3.2 Visualization

This section describes visualizations  $\mathcal{V}$  that we investigate in our study: An animation of moving dots  $\mathcal{V}_{\text{anim}}$ , the space-time cube  $\mathcal{V}_{\text{cube}}$ , and vessel density  $\mathcal{V}_{\text{dens}}$ . Animation is chosen, since it appears often in vessel monitoring systems. The space-time cube is chosen because it is a visualization showing many movement features, can be used as a geographical visualization [92], and is widely evaluated (see Section 4.2.1), which sets a baseline for our comparison. To avoid that extra features result in cross-effects, only the core elements of the visualizations are implemented.

#### Animation

Trajectories are animated by displaying a sequence of frames  $[f_0, \dots, f_T]$ , where each frame  $f_t$  contains the position at a certain time  $t$  for all trajectories. The objects are rendered as circular dots with a diameter of 8 pixels. The whole animation takes about ten seconds. We do not allow any interaction, i.e., no pause and no rewind, since for the chosen movement features  $\mathcal{M}$  interaction does not bring extra value: Indeed, the animation is long enough to make a decision, hence a rewind action is not needed. For  $\mathcal{M}_{\text{stop}}$  and  $\mathcal{M}_{\text{fast}}$  a pause action is pointless, since these features are not distinguishable anymore in a single frame. For  $\mathcal{M}_{\text{lanes}}$  a pause action could be beneficial to count the dots, though the differences between lanes is large enough to see the difference without explicit counting. The speed of the animation is tuned to be acceptable for an average person.

#### Space-Time Cube

In the seventies, Hågerstrand [68] proposed to show trajectories in a 3D cube with two axes for space and one for time: the space-time cube. Trajectories are displayed as a 3D curve  $(x(t), y(t), t)$ , called the space-time path. The footprint, the projection of

the space-time path on the  $xy$ -plane, is omitted for possible cross-effects with vessel density, which is a static 2D map as well. In Figure 4.2 the space-time cube reveals a busy lane as a lane containing many space-time paths, a stop as a vertical segment in a space-time path, and a fast object as a space-time path with a flat slope. When using the space-time cube, occlusion is a problem that often occurs. For this reason we force occlusion by testing with four lanes instead of two or three lanes. To overcome the resulting overdraw we have used alpha blending with an opacity of 0.85, which shows a subtle highlight for dense areas. Furthermore, to keep the lanes distinguishable for comparison, each lane is rendered with perceptually balanced colors generated with PaletView by Wijffelaars *et al.* [160]. Interaction is minimalized to prevent occlusion effects; the subject may rotate the cube with the mouse around the time axis and change the angle to look into the cube.

### Vessel Density

Vessel density as presented in Chapter 3 convolves trajectories in the  $xy$ -plane with a kernel moving along the path with the same speed as the object. The density represents the sum of the durations averaged over time that objects are in the neighbourhood. So, fast objects have a relatively low contribution and slow objects have a relatively high contribution. Just density is ambiguous, since many objects passing by also give a high contribution. To distinguish between a slow vessel and many fast vessels, two density fields with two kernel sizes (large and small) are used simultaneously. The rendering then consists of a color mapped large kernel density and an illuminated height map of the sum of both a small and a large kernel density. By means of the colors we can see the global route structure, while in the shading detailed trajectories pop up, in particular stopping vessels, which are shown as dark dots in the shading (see inset in Figure 4.2). The density ambiguity solved by comparing the color mapped density with the number of trajectories in the shading.

## 4.4 Evaluation Setup

We have everything in place to generate trials, exercises for the subjects. Table 4.2 shows an overview of the independent variables and their settings in our user test, as given in the previous section. In this section we generate trials and compose them together in a complete user test.

Name	Symbol	Settings
Movement feature	$\mathcal{M}$	$\mathcal{M}_{\text{stop}}, \mathcal{M}_{\text{lanes}}, \mathcal{M}_{\text{fast}}$
Visual clutter	$\mathcal{N}$	$\mathcal{N}_{\text{simple}}, \mathcal{N}_{\text{complex}}$
Trajectory count	$\mathcal{C}$	$\mathcal{C}_{\text{sparse}}, \mathcal{C}_{\text{dense}}$
Visualization	$\mathcal{V}$	$\mathcal{V}_{\text{anim}}, \mathcal{V}_{\text{cube}}, \mathcal{V}_{\text{dens}}$

Table 4.2: An overview of the experiment parameters.

### 4.4.1 Trial Design

The task in each trial is a multiple choice question that differs per movement feature (see Table 4.3). The question is first shown on a blank screen, and when the user presses the space bar, a scene with four numbered lanes is shown, and the response time measuring starts. For clarity the question is repeated at the bottom of the screen. To avoid reading mistakes the three questions are displayed in distinct colors. The subject chooses a lane by entering its number on the keyboard after which the response time is measured and the correctness is evaluated (correct scores 100%, wrong 0%).

Setting	Question
$\mathcal{M}_{\text{fast}}$	In which lane do objects have the highest speed?
$\mathcal{M}_{\text{stop}}$	Which lane contains the most stopping objects?
$\mathcal{M}_{\text{lanes}}$	Which lane contains the most objects?

Table 4.3: Questions per movement feature.

### 4.4.2 Test Design

The user test is a within-subject evaluation, is executed according to the schedule given in Table 4.4, and takes about one hour per subject. First, the subjects are instructed using Figure 4.2 in poster format. This figure contains screenshots of all the visualizations with all different movement features to explain the significant visual features. Then the subjects get familiar with the test environment using test trials with twice all combinations of visualizations and movement features for a small number of trajectories. In the first run the subjects see only two lanes to get familiar with the visualizations and in the second run four lanes to get familiar with the size of the scene in the actual test.

	Description	Duration
1.	Explain test with poster	10 min
2.	Test trials with feedback	10 min
3.	Test (180 trials à 10 sec)	30 min
4.	Fill out questionnaire	10 min
	Total	60 min

Table 4.4: The user test procedure.

The test consists of a sequence of trials generated by combinations with all settings for  $\mathcal{V}$ ,  $\mathcal{M}$ ,  $\mathcal{N}$ , and  $\mathcal{C}$ . These sequences are grouped by the nine possible pairs of visualizations in  $\mathcal{V}$  and movement features in  $\mathcal{M}$ . Each subject gets a sequence in a different order to compensate for learning effects and fatigue effects. For each pair, five groups of trials consisting of all four randomly ordered combinations of visual clutter  $\mathcal{N}$  and trajectory count  $\mathcal{C}$  are generated. This results in 180 trials per subject. After completion of the test, the subject is interviewed by means of a questionnaire about preferences for visualizations and possible reasons for imperfect performance.

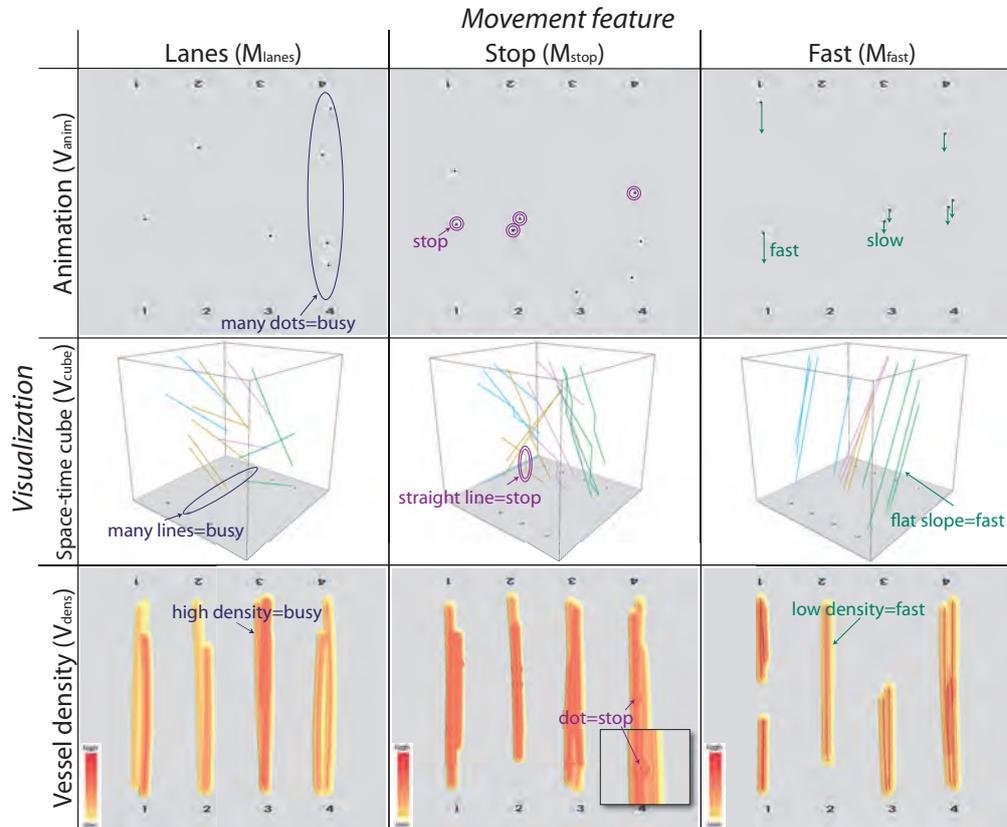


Figure 4.2: A table showing all movement features with all visualizations. Each data set with a certain movement feature contains the same lanes with trajectories, but the order is changing in each trial and hence in each visualization shown.

### 4.4.3 Test Execution

The tests are conducted with a 20.1” LCD screen with 1680x1050 resolution and 60cm view distance. The framework is implemented using Java and JOGL, with shader programs for generating vessel density on-the-fly. The trials are shown in full-screen mode to avoid distractions.

The test participants are students and staff of the mathematics and computer science department of Eindhoven University of Technology. The age of the 17 participants ranges between 23 and 49 years old, all are right-handed, none of them is color-blind, and all are familiar with information visualization concepts. None of them is experienced with the space-time cube and two are experts in density visualizations.

## 4.5 Evaluation Results

In this section we evaluate the results from the test described in the previous section. First we describe the hypotheses as raised before executing the user test. Then we analyze the data and statistically test these hypotheses, and finally, we discuss the results and distillate the lessons learned from this experiment.

### 4.5.1 Hypotheses

We expect the visualizations to perform according to the following six hypotheses:

*H1: There is a high positive correlation between response time and correctness.*

If a visualization takes longer to explore, the answers are more likely to be correct.

*H2: No visualization excels for all movement features.*

*A priori*, we can see that some visualizations perform better than others:  $\mathcal{V}_{\text{anim}}$  may perform best for detecting fast objects, and  $\mathcal{V}_{\text{dens}}$  is likely to perform best for lanes, but has to compete with  $\mathcal{V}_{\text{cube}}$  for stops.

*H3:  $\mathcal{V}_{\text{cube}}$  is (a) least robust to clutter, (b) least scalable.*

Both the complexity introduced in the trajectory data and showing more trajectories will clutter the view on the space-time paths, resulting in less readability.

*H4: For  $\mathcal{M}_{\text{stop}}$ , (a)  $\mathcal{V}_{\text{anim}}$  performs worst, independent of the number of trajectories, (b)  $\mathcal{V}_{\text{cube}}$  performs best for a small number of trajectories, while (c)  $\mathcal{V}_{\text{dens}}$  performs best for a large number of trajectories, all independent of clutter.*

We expect that a static representation will perform better than  $\mathcal{V}_{\text{anim}}$ , since animation lacks giving an overview.  $\mathcal{V}_{\text{cube}}$  is expected to perform best for a small number of trajectories, since the vertical lines in the space-time path tend to be more visible than the small dots in  $\mathcal{V}_{\text{dens}}$ , though for many trajectories  $\mathcal{V}_{\text{dens}}$  performs best, since then in  $\mathcal{V}_{\text{cube}}$  the vertical lines are not visible due to occlusion.

*H5: For  $\mathcal{M}_{\text{lanes}}$ , (a)  $\mathcal{V}_{\text{dens}}$  performs best, followed by (b)  $\mathcal{V}_{\text{cube}}$  and (c)  $\mathcal{V}_{\text{anim}}$ , independent of the number of trajectories or clutter.*

$\mathcal{V}_{\text{dens}}$  is expected to perform best, due to a clean representation for lanes, counting happens instantly.  $\mathcal{V}_{\text{anim}}$  is expected to perform least, since no overview is given.

*H6: For  $\mathcal{M}_{\text{fast}}$ , (a)  $\mathcal{V}_{\text{dens}}$  performs worst, independent of the number of trajectories, (b)  $\mathcal{V}_{\text{cube}}$  performs best for a small number of trajectories, while (c)  $\mathcal{V}_{\text{anim}}$  performs best for a large number of trajectories, all independent of clutter.*

It is expected to be hard to see fast moving objects in  $\mathcal{V}_{\text{dens}}$ , since the smoothing differences in density are smaller than the differences in speed.  $\mathcal{V}_{\text{anim}}$  is expected to perform best for a large number of trajectories, since there is a flow of dots appearing that catches the eye. This flow is missing for a small number of trajectories, in which case we expect the opposite order for  $\mathcal{V}_{\text{anim}}$  and  $\mathcal{V}_{\text{cube}}$ .

### 4.5.2 Analysis

The performance of the test is measured in terms of correctness (overall score:  $\mu = 77\%$ ,  $\sigma = 42.3\%$ ) and response time in seconds (overall score:  $\mu = 8.8s$ ,  $\sigma = 8.3s$ ). In Figure 4.3 the mean correctness and mean response time per subject are displayed in scatter plots with different visualizations for all data and for each movement feature. In the scatter plot with all data we see clear clusters for  $\mathcal{V}_{\text{dens}}$  performing good (high correctness in short response time) and  $\mathcal{V}_{\text{anim}}$  performing less good (lower correctness than  $\mathcal{V}_{\text{dens}}$  and with low response time), while the spread for  $\mathcal{V}_{\text{cube}}$  is high in both correctness and response time. By differentiating between movement features, we can see that the spread of  $\mathcal{V}_{\text{cube}}$  is mainly due to a high spread for stops. Since for all data there is no high positive correlation per visualization given in Figure 4.3, we reject H1.

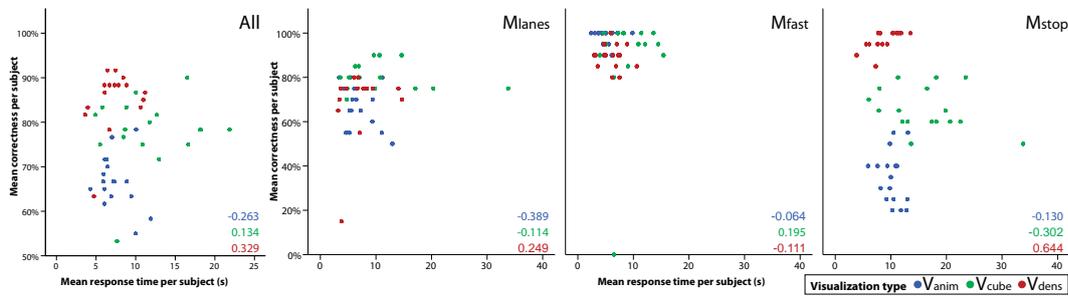


Figure 4.3: Scatter plots with the mean correctness and mean response time per subject. The Pearson's correlation coefficients are shown in the bottom right corner. From left to right: All data and data grouped by  $\mathcal{M}_{\text{lanes}}$ ,  $\mathcal{M}_{\text{fast}}$ , and  $\mathcal{M}_{\text{stop}}$ .

The response time is not an appropriate measure for performance, since each visualization is explored differently due to interaction in  $\mathcal{V}_{\text{cube}}$  and the obligation to watch  $\mathcal{V}_{\text{anim}}$  for a while to be able to answer the question. Therefore, we only consider correctness as a measure in the performance analysis for the remaining hypotheses.

In Figure 4.4 the mean correctness is given for each visualization and grouped by movement feature, with a 95% confidence interval. The difference in correctness between the visualizations for stops is noticeably high, while for the other movement features the correctness seems to be in the same order of magnitude. Based on the mean correctness we can see that each visualization excels for one movement feature. In order to find out if the visualization with the highest correctness is significantly better than the others, we analyze the data with a one-way ANOVA test and a Tukey's HSD post hoc test for pairwise comparison. These tests can be done for each movement feature, in fact for any subset of the data. As an example, we show an ANOVA test with Tukey's HSD post hoc test for  $\mathcal{M}_{\text{lanes}}$ , which results in homogeneous subsets (HS) of visualizations, i.e., visualizations without significant differences are contained in one homogeneous subset, while between these subsets a significant difference exists.

The ANOVA analysis with the subset of the data containing  $\mathcal{M}_{\text{lanes}}$  shows that there is a significant effect for the mean correctness between the various visualizations, since

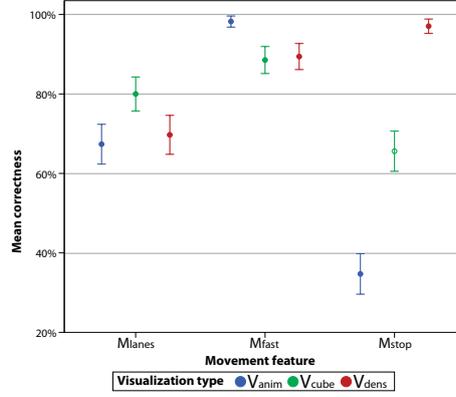


Figure 4.4: The mean correctness with error bars (95% confidence interval) for visualizations per movement feature.

$F(2, 1017) = 7.785$ , for  $p < 0.001$ . From Figure 4.4 for  $\mathcal{M}_{lanes}$ , we can see an ordering in the mean correctness for the three visualizations. To reveal whether the pairwise differences are significant, we conduct a Tukey’s HSD test. The results are shown in Table 4.5 with significant differences in blue. From this table we can derive the following HSS:  $[\mathcal{V}_{cube}, \{\mathcal{V}_{anim}, \mathcal{V}_{dens}\}]$ , which should be read as:  $\mathcal{V}_{cube}$  has a significantly higher correctness than the others, while  $\mathcal{V}_{anim}$  and  $\mathcal{V}_{dens}$  do not have a significant difference.

	$\mathcal{V}_{anim}$	$\mathcal{V}_{cube}$	$\mathcal{V}_{dens}$
$\mathcal{V}_{anim}$	X	5.39	1.24
$\mathcal{V}_{cube}$	5.39	X	4.14
$\mathcal{V}_{dens}$	1.24	4.14	X

Table 4.5: Tukey’s HSD post hoc test results. For  $\alpha = 0.05$ , 3 treatments (visualizations), and a within-groups  $df > 1000$  (for our data  $df = 1017$ ), all values larger than 3.31 are shown in blue and have a significant difference.

Similarly as in the paragraph above, we have computed HSSs for all the combinations of the independent variables  $\mathcal{M}$ ,  $\mathcal{N}$ , and  $\mathcal{C}$ , which are shown in the graph in Figure 4.5, with the result of the example displayed in the green node  $\mathcal{M}_{lanes}$ . The root  $All$  is shown in dark purple, representing all data, from which we branch in all movement features  $\mathcal{M}$  colored in various colors, and we branch further in all the combinations of visual clutter  $\mathcal{N}$  and trajectory count  $\mathcal{C}$ . Two nodes are connected if just one new variable is added to the list of variables in the direction from the root to a leaf. With this tree we can analyze the remaining hypotheses.

H2 cannot be rejected for correctness, since from nodes  $\mathcal{M}_{stop}$ ,  $\mathcal{M}_{lanes}$ , and  $\mathcal{M}_{fast}$  follows that each movement feature has a visualization that performs the best.

For the robustness to clutter in H3a, we have to look at the purple node  $All - \mathcal{N}_{complex}$  with  $[\mathcal{V}_{dens}, \mathcal{V}_{cube}, \mathcal{V}_{anim}]$ , from which we conclude that  $\mathcal{V}_{anim}$  is the least robust to clutter and therefore we reject H3a. For scalability in H3b, we inspect node  $All - \mathcal{C}_{dense}$  with

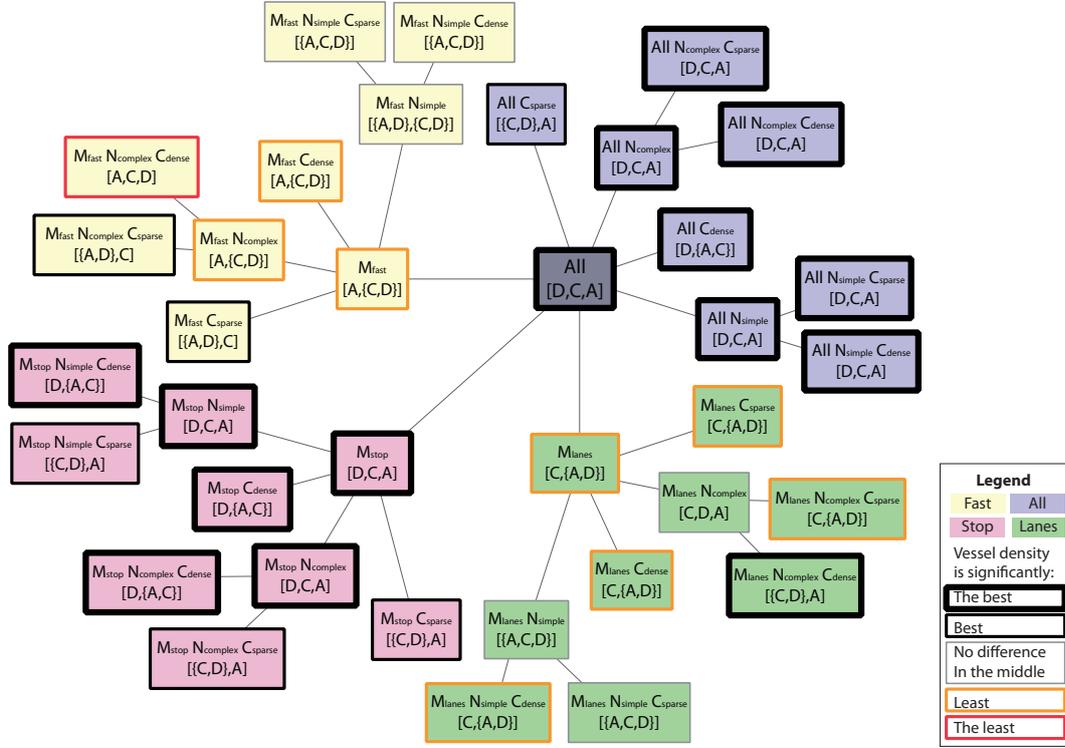


Figure 4.5: A tree showing in each node subsets of the test data given by the filter criteria. For each node the homogeneous subsets of visualizations ( $A=\mathcal{V}_{anim}$ ,  $C=\mathcal{V}_{cube}$ ,  $D=\mathcal{V}_{dens}$ ) are given, which are obtained by conducting an ANOVA test with correctness as a dependent variable and visualization as an independent variable with Tukey's HSD post hoc tests for  $\alpha = 0.05$ .

$[\mathcal{V}_{dens}, \{\mathcal{V}_{anim}, \mathcal{V}_{cube}\}]$  and conclude that indeed  $\mathcal{V}_{cube}$  is in the least performing HS, though together with  $\mathcal{V}_{anim}$ , therefore we cannot reject H3b.

For stoppers in all red nodes,  $\mathcal{V}_{anim}$  is always in the least performing HS, therefore we cannot reject H4a. For  $\mathcal{M}_{stop}^*-\mathcal{C}_{sparse}$  nodes,  $\mathcal{V}_{cube}$  is always in the best performing HS, though always together with  $\mathcal{V}_{dens}$ , therefore we cannot reject H4b. For  $\mathcal{M}_{stop}^*-\mathcal{C}_{dense}$  nodes,  $\mathcal{V}_{dens}$  is always the best performing visualization, therefore we can also not reject H4c. Combined with the observations in H4b, we can say that  $\mathcal{V}_{dens}$  is the best performing visualization independent of visual clutter and number of trajectories.

Regarding detecting lanes,  $\mathcal{V}_{dens}$  is only in the best performing HS for  $\mathcal{M}_{lanes}-\mathcal{N}_{complex}-\mathcal{C}_{dense}$ ,  $\mathcal{M}_{lanes}-\mathcal{N}_{simple}$ , and  $\mathcal{M}_{lanes}-\mathcal{N}_{simple}-\mathcal{C}_{sparse}$ , therefore we reject H5a.  $\mathcal{V}_{cube}$  is for all green nodes for lanes in the best performing HS and  $\mathcal{V}_{anim}$  is always in the least performing HS, therefore we reject H5b and we cannot reject H5c.

For finding fast movers,  $\mathcal{V}_{dens}$  is not performing worst only for  $\mathcal{M}_{fast}-\mathcal{C}_{sparse}$ , and  $\mathcal{M}_{fast}-\mathcal{N}_{complex}-\mathcal{C}_{sparse}$ . Due to the dependence on the number of trajectories we should

reject H6a. For all  $\mathcal{M}_{\text{fast}}\text{-}^*\text{-}\mathcal{C}_{\text{sparse}}$  nodes  $\mathcal{V}_{\text{cube}}$  performs the worst, hence we reject H6b, while for all  $\mathcal{M}_{\text{fast}}\text{-}^*\text{-}\mathcal{C}_{\text{dense}}$  nodes  $\mathcal{V}_{\text{anim}}$  performs best, therefore we cannot reject H6c.

Next to the analysis for the hypotheses we observe another noticeable significant difference. When not considering the differentiation between movement features, that are the nodes  $All\text{-}^*$ , we can see that  $\mathcal{V}_{\text{dens}}$  performs best overall, independent of the number of trajectories and independent of the complexity of the trajectories. From Figure 4.4 we should conclude that this high overall performance is due to a high mean performance of  $\mathcal{V}_{\text{dens}}$  for stops, which cannot be compensated by the other movement features.

### 4.5.3 Discussion

From the analysis in the previous section we can conclude that  $\mathcal{V}_{\text{dens}}$  performs best for  $\mathcal{M}_{\text{stop}}$ ,  $\mathcal{V}_{\text{anim}}$  for  $\mathcal{M}_{\text{fast}}$ , and  $\mathcal{V}_{\text{cube}}$  for  $\mathcal{M}_{\text{lanes}}$ . The first combination seems to be logical, since stops are shown in a clean overview using density, while for the others stops are not always visible. That animation performs best for fast movements is logical as well, since it is easier to compare motion than comparing the other visual features. We are surprised that the space-time cube does not perform best too, since comparing slopes tends to be easy as well, however in this particular visualization the occlusion and viewpoint might obscure the slope, as also mentioned by the subjects in the questionnaire. It is remarkable that the space-time cube works best for the lanes task, since the density is expected to perform best here, due to the nice overview of sea lanes in [162]. Reasons why density does not excel for comparing lanes could be that: (1) due to the smoothing, variations between the number of trajectories diminish; (2) due to the spatial variation, the lane with the highest number of trajectories may not have the highest density if the lane is slightly wider than the other lanes; (3) for a small number of trajectories, which may not cover the whole lane, holes may occur as in Figure 4.2 for  $\mathcal{M}_{\text{fast}}$ .

To come to a final judgement about the performance of vessel density in comparison to the other visualizations, we have annotated the nodes in Figure 4.5 with borders for vessel density being *the best* (fat black), *best* (thick black), *least* (orange), and *the least* (red). Based on these annotations we can conclude that density performs best for finding stoppers. For the other movement features density is competitive with the other visualizations, since only in a single case ( $\mathcal{M}_{\text{fast}}\text{-}\mathcal{N}_{\text{complex}}\text{-}\mathcal{C}_{\text{dense}}$ ) density performs the least, whereas in all other cases it performs as good as another visualization. Furthermore, from feedback of the participants, density is often chosen to be the favorite visualization: overall 88%, for stoppers 100%, for lanes 82%, and for many, for simple, and for complex trajectories each 71%. Overview, no interaction, clear features, no constant attention while watching, and aesthetics are the reasons why participants preferred density.

This study is focussed on vessel traffic, but the chosen movement features also occur for other objects, such animals stopping at drinking places. However, vessel density excels for route-bound traffic, where straight lanes often occur, resulting in a strong visual cue. The results are likely to generalize to real-world data with the tested movement features occurring within a single lane. For movement features that are not tested

it is unknown how vessel density will perform, but we know that features related to direction are invisible in vessel density. Since the tested movement features are not heavily dependent on direction, we expect the results to be similar for two-way lanes. For movements with more vivid speed variations than the assumed constant speed, the results are expected to differ, since for stops and fast movers it will be harder to determine what a stop or fast mover is. For busy lanes the results may be similar, since the temporal component is not much of influence. The settings for the number of trajectories  $\mathcal{C}$  and visual clutter  $\mathcal{N}$  are optimized for vessel traffic, and need recalibration for non-route bound traffic.

## 4.6 Conclusion and Future Work

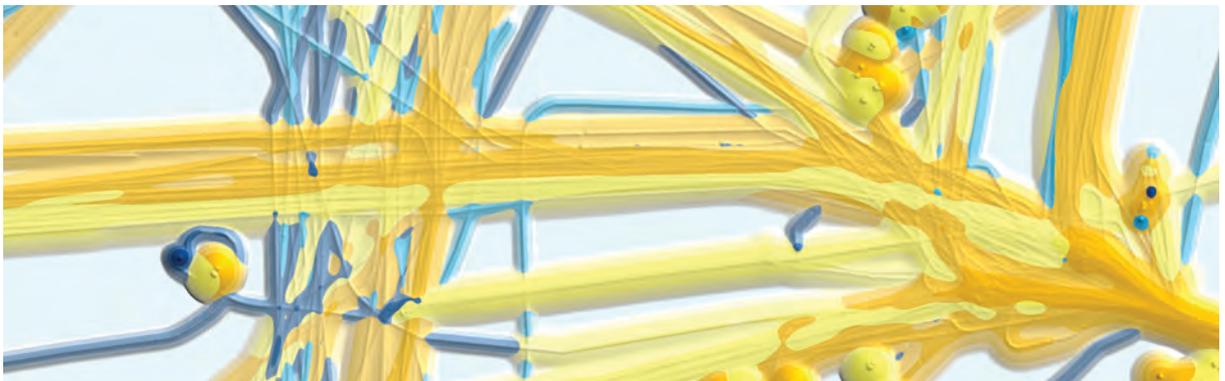
We have presented a controlled user experiment to test how vessel density performs with respect to animation and the space-time cube for sets of simulated vessel trajectories with properties of real-world data. Therefore, we have tested three different maritime movement features (stoppers, busy lanes, fast movers), for robustness with respect to complexity and scalability. For stops density performs significantly best, is robust to visual clutter and scales well. For the other features, density performs at least as good as the other visualizations.

For future work we would like to test vessel density with real operators to find out how useful it is in the real-world. Furthermore, in our test the settings for visual clutter and scalability are limited to two cases and we would like to find the critical values by increasing these number of cases. This was not yet possible due to the combinatorial explosion of four independent variables. A follow-up test with combinations of the visualizations, such as an animation with vessel density is of interest. Other movement features, such as congestions, should be tested to find out how vessel density performs in the whole spectrum of tasks. Finally, we are curious if other variables than visual clutter and scalability play a role in finding movement features in real-world data.

*We would like to thank the participants of our user study for their time.*

# 5

## Density Maps



*In this chapter we present a method to interactively explore multiple attributes in trajectory data using density maps, i.e., images that show an aggregate overview of massive amounts of data. So far, a density map has mainly been used to visualize a single attribute. Density maps are created in a two-way procedure; first smoothed trajectories are aggregated in a density field, and then the density field is visualized. In our approach, the user can explore attributes along trajectories by calculating a density field for multiple subsets of the data. These density fields are then either combined into a new density field or first visualized and then combined. Using a widget, called a distribution map, the user can interactively define subsets in an effective and intuitive way, and, supported by high-end graphics hardware the user gets fast feedback for these computationally expensive density field calculations. We show the versatility of our method with use cases.*

## 5.1 Introduction

With current technology, we are able to track massive amounts of objects with other attributes than just *time* and *position*, such as forces, object type, or object size. Additional knowledge about objects can be added by attributes obtained by reasoning or using web sources as described in Chapter 7. Current state-of-the-art aggregation methods for moving objects often do not take these additional attributes into account. Kernel Density Estimation (KDE) [133] is an aggregation method suitable for showing an overview of massive amounts of data and has been adapted for trajectory data, but only a few of these KDE-based methods take attributes into account. For instance, in Chapter 3 we only encode the velocity of ships and Peters *et al.* [114] only encode the direction of airplanes in a density field. Since a single density field has limited capabilities for further extensions towards including more attributes, we focus on combining multiple density fields for subsets defined with multiple attributes.

In this chapter we present a new approach for the interactive exploration of density maps of trajectories with multiple attributes. Interactive exploration requires efficient and effective methods to both select subsets of interest and to generate density fields. As a result, multiple density fields need to be shown simultaneously in a density map to allow the user to determine similarities and differences. With a generic model, we are able to generate a wide range of different maps, which support story telling to explain what happened with the objects in the data set.

The chapter is organized as follows. In Section 5.2 we describe the main features of our density maps and elaborate on the implementation on graphics hardware in Section 5.3. Section 5.4 shows the versatility of density maps with use cases. In the final section, we conclude the chapter and suggestions for future work.

---

*This chapter is based the paper “Interactive Visualization of Multivariate Trajectory Data with Density Maps” [129] presented at the IEEE Pacific Visualization Symposium organized in Hong Kong in 2011 and the extended version “Interactive Density Maps for Moving Objects” [130] published in IEEE Computer Graphics and Applications.*

## 5.2 Density Maps

We present a framework to enhance vessel density maps, as described in the Chapter 3, with additional attributes. Figure 5.1 shows an overview of the new architecture for density maps, in which the architecture (see Fig. 3.4 on page 35) for creating vessel density maps recurs. The main principle we apply is that the data is filtered resulting in various subsets, which are aggregated into a single representation in either the density model or the visualization phase. In the remainder of this section we discuss the various aggregations in more detail. Examples of how these features can be used in real-world scenarios are shown in Section 5.4.

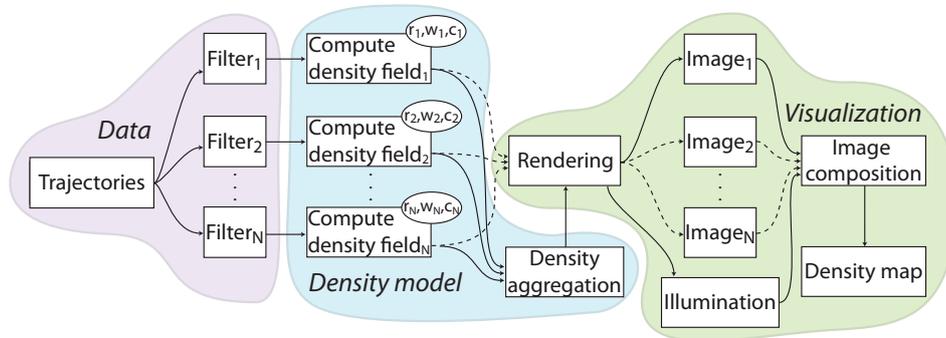


Figure 5.1: The architecture of our framework to generate density maps. From left to right, the trajectory data is split with various filters resulting in a number of subsets for which we compute a density field with the given weight  $w_i$  and kernel radius  $r_i$ . From here there are two routes. We may aggregate the density fields in a single density field via the solid lines. Then the single density field is rendered to an image with a color map and visualized together with its illumination in a density map. The other route, which also includes the dashed lines, first renders each density field using the corresponding color map  $c_i$  and then composes them and applies the illumination of the aggregate density field resulting in a density map.

### 5.2.1 Subsets and Parameters

A vessel density field as defined in Chapter 3 is computed based on a subset of the data defined by a filter. Typically, a filter selects all data elements where the value of a specified attribute is within a given range. For example, filtering on the time attribute may result in a subset representing all objects in the night. In Section 5.2.5 we explain techniques for creating filters using multiple attributes.

For each subset a density field is computed with its own user-defined parameters: weight  $w$  and kernel radius  $r$ . The values of these parameters can be set for each subset with an interactive bar chart, with each bar shown with the hue of the color map  $c$  of the density field. Figure 5.2 shows such a bar chart for the kernel radius of five subsets used in Figure 5.11 where each subset represents a ship type. The density fields are updated after changing the bar chart.

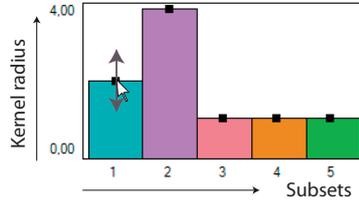


Figure 5.2: Interactive bar chart to set a density parameter for subsets.

Once we have a set of density fields we can follow two routes: Either we forward the density fields to the rendering stage and compose the images as explained in Section 5.2.4, or we can aggregate the density fields into a single density field.

### 5.2.2 Density Aggregation

For density aggregation we choose the following variations: addition, absolute difference, and anomaly detection (see Fig. 5.3). For a cell  $Q$  density aggregations  $\mathcal{D}$  are defined in terms of densities  $D_i$ :

$$\mathcal{D}_{\text{addition}}(Q) = \sum_{i=1}^N D_i(Q), \tag{5.1}$$

$$\mathcal{D}_{\text{difference}}(Q) = |D_2(Q) - D_1(Q)|, \text{ and} \tag{5.2}$$

$$\mathcal{D}_{\text{anomaly}}(Q) = \max(0, D_2(Q) - D_1(Q)). \tag{5.3}$$

In Equation (5.1) we define an addition for multiple density fields. Note that by choosing the density field weights  $w_i = 1/N$ , we obtain an average density. With  $\mathcal{D}_{\text{addition}}$  we obtain the vessel density map, described in Chapter 3, if we use two density fields containing both all data and if we convolve one with a large and one with a small kernel radius. The absolute difference in Equation (5.2) and the anomaly detection in Equation (5.3) are defined for two densities. In aggregation  $\mathcal{D}_{\text{difference}}$ , we simply search for a symmetrical difference between the two density fields. The aggregation  $\mathcal{D}_{\text{anomaly}}$  is typically used to find deviations between a sparse density field  $D_2$  and a dense density field  $D_1$ , where  $D_2$  contains current movements and  $D_1$  represents normal movements. In Figure 5.3 the gray trajectory is only gray, or is anomalous, at the places where the other trajectory is not located. Section 5.4.1 shows anomalies with real-world data.

Aggregating densities is a known technique in geographical information system for deriving features in combined raster maps, which are defined with map algebra [144].

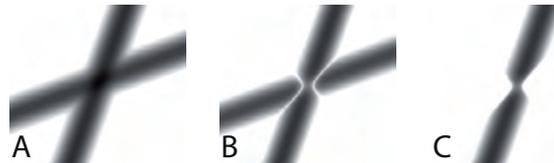


Figure 5.3: Density aggregation: (A) Weighted addition, (B) absolute difference, and (C) anomaly.

In the basic map algebra cells are cell-wise combined with an expression, but there are also map algebras considering spatio-temporal data [101] or non-raster data [30].

### 5.2.3 Rendering

In the visualization part of the architecture in Figure 5.1 the density fields are rendered to images by mapping density values to colors via a color map  $c$ . A single hue color map is used for each density field. The user is supported by a default setting of clearly distinguishable hues to obtain a perceptually balanced map [106], as used in Figure 5.11. The default setting is a pastel rainbow color map generated with PaletteView [160] and sampled at equal intervals for a given number of colors. The chosen color is associated to the highest density value of the field, and a complete color map is created by interpolating the saturation towards white, see the lower right corner of Figure 5.10. Note that this approach limits the number of density fields, since for more than approximately ten colors, the density fields become hard to distinguish.

If only one density field is visualized a multi-hue color map is used, since hue is not needed to distinguish between density fields. We choose to use a yellow-to-red color map with more contrast between density levels (see Fig. 5.8A and 5.10A).

### 5.2.4 Image Composition

Not only density fields but also the color mapped images of density fields can be aggregated. To obtain a density map, we aggregate the color mapped images with an operation called image composition and, finally, apply Phong shading on a height field given by the density aggregation field (see Fig. 5.1).

We have six types of image compositions (see Fig. 5.4): single-field  $\mathcal{J}_{\text{single}}$ , multi-field  $\mathcal{J}_{\text{multi}}$ , opacity-blend  $\mathcal{J}_{\text{opacity}}$ , max-blend  $\mathcal{J}_{\text{max}}$ , block  $\mathcal{J}_{\text{block}}$ , and pie  $\mathcal{J}_{\text{pie}}$ . The output of  $\mathcal{J}_{\text{single}}$  is one of the color mapped density fields and for  $\mathcal{J}_{\text{multi}}$  it is the color mapped aggregated density field obtained by the selected density aggregation. Since  $\mathcal{J}_{\text{single}}$  and  $\mathcal{J}_{\text{multi}}$  have a single density field as input they can use a multi-hue color map. In the density maps shown in Figure 5.10A  $\mathcal{J}_{\text{single}}$  is used, in Figure 5.8A  $\mathcal{J}_{\text{multi}}$  is used.

The  $\mathcal{J}_{\text{opacity}}$  composition is a weighted sum of the colors of the rendered density fields in RGB-space with the opacity as weight. With  $\mathcal{J}_{\text{max}}$  we show the color of the density field with the highest value, as used in Figure 5.8B.

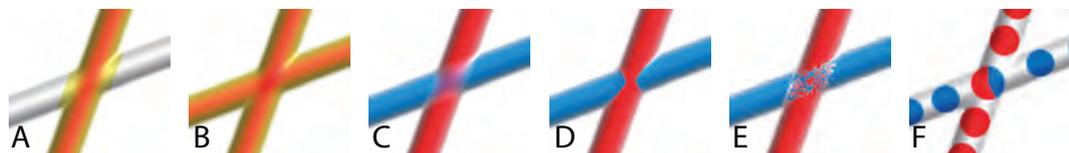


Figure 5.4: Image composition: (A) Single-field, (B) multi-field, (C) opacity-blend, (D) max-blend, (E) block, and (F) pie.

From the latter two image compositions it is hard to see which colors are used in a certain neighbourhood, therefore we introduce  $J_{\text{block}}$ , which is inspired by attribute blocks [103]. In this image composition, a new image is created covering the same area as the input images, but with a much lower resolution to avoid dithering effects. The color of the pixels in this new image are set as follows: If a pixel overlaps with one or more density values larger than zero, one of these density fields is randomly chosen and the corresponding color is used. In the randomization we use a probability mass function, which is generated by the proportions of the density values per field in this block. Hagh *et al.* [69] have evaluated this composition, called color weaving, and show that it is more effective than color blending ( $J_{\text{opacity}}$ ) for two to four colors, while for upto six colors the error rates significantly increase. In Section 5.4.1,  $J_{\text{block}}$  is used to reveal various ship types in a neighbourhood.

For blocks it is hard to see the proportions of overlapping density values. We solve this by introducing  $J_{\text{pie}}$ , which shows the proportions by means of icons in a new image. We choose pie charts because of their symmetrical shape, resulting in a balanced picture. The area of the pie charts are proportional to the combined total density at the center location and the wedge angles are proportional to the ratio between the individual and total density field values in the area of the pie chart. The charts are placed iteratively at non-overlapping locations with the highest total density values. The number of charts, the spacing, and the scale can be interactively set by the analyst (see Fig. 5.5). When using pie charts it is possible to quantify the proportions between density fields, however, it only shows a coarse sampling and it is therefore not suitable for detailed analysis. In Section 5.4.2 we have used  $J_{\text{pie}}$  to analyze visit purposes of pedestrians.

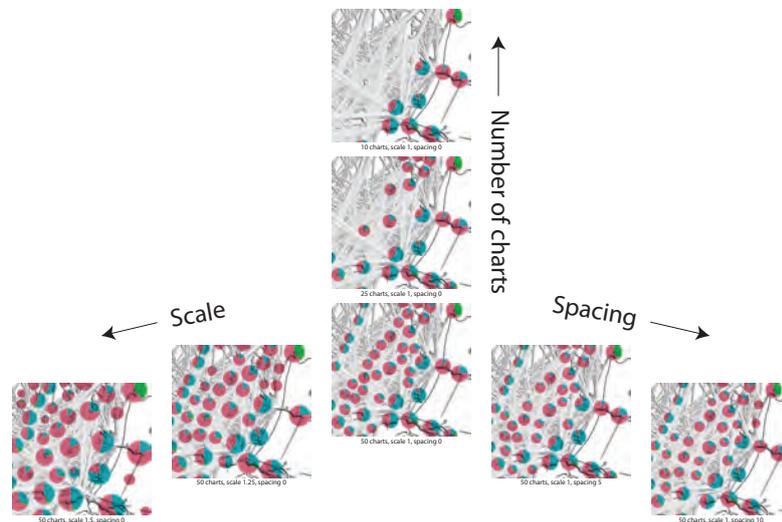


Figure 5.5: The influence of the number of charts (10, 25, and 50 charts, with scale 1 and 0 spacing), scale (scale 1, 1.25, and 1.5, for 50 charts with 0 spacing), and spacing (spacing 0, 5, and 10 for 50 charts at scale 1) for  $J_{\text{pie}}$  image composition.

### 5.2.5 Multivariate Filters

So far, filters for single attributes have been considered. The usability of our method greatly increases by defining interactive filters based on multiple attributes. To this end, we have developed an interactive widget, called a Distribution Map (DM), as shown in Figure 5.6 with various pairs of attributes used in vessel tracking.

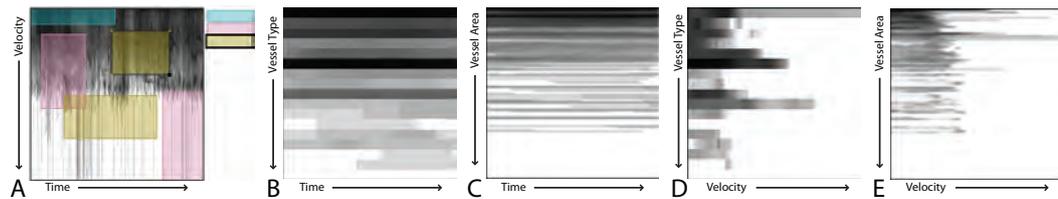


Figure 5.6: *Distribution maps for several pairs of vessel attributes.*

A DM consists of a 2D-plane with two axes corresponding to two chosen attributes and shows in gray-scale the distribution of the values of these attributes as they appear in the trajectories. Guided by this distribution, the user, while defining a filter, selects combinations of attribute intervals by drawing a set of rectangles, colored with the same hue as the color map  $c$  for rendering the density field. For example the pink rectangles in Figure 5.6A represent medium-speed moving vessels in the morning together with fast moving vessels in the evening. Adaptations of the rectangles result in an update of the density field of the corresponding subset.

The distributions are generated by drawing, with additive blending, bright gray polylines as trajectories for the two attributes given on the axes. With this technique we mimic counting how long an attribute pair occurs in the trajectories.

## 5.3 Implementation

In this section we give a method for parallel computation of density fields on the GPU-based on the non-parallel version in Algorithm 1 on page 34. As in that algorithm we assume that the trajectory points are given in geographic coordinates and a cylindrical equal-area map-projection  $G$  [135] is used to transform these in map coordinates. Given the density fields the visualization of the final density map is straightforward, also realized on the GPU, and omitted here.

### 5.3.1 Parallelization

A method for parallelization of density field computations either has an image-space or an object-space approach. In an image-space approach the parallelization is done over computations per pixel, where in an object-space approach it is done over computations per object. Here we chose an object-space approach where we simply traverse the objects, in this case the trajectory segments. Additional advantages are that adding

---

**Algorithm 2** ParallelComputeDensityField()

---

```

CopyTrajectoriesToVideoMemory()
 $D \leftarrow \mathbf{0}_{H,B}$  {Density D becomes zero matrix}
for  $o \in \text{Objects}$  do
  for tuples  $\alpha_i$  and  $\alpha_{i+1}$  in trajectory  $\tau_o$  do in parallel
     $\mathbf{p}_0 \leftarrow \mathbf{p}_o(t_i); \mathbf{p}_1 \leftarrow \mathbf{p}_o(t_{i+1})$ 
     $OBB \leftarrow$  oriented bounding box at distance  $K$  to  $\widehat{\mathbf{p}_0\mathbf{p}_1}$ 
    for  $(u, v) \in OBB$  do in parallel
      {Get geographical coordinate of the cell center}
       $\mathbf{q} \leftarrow G^{-1}(u, v)$ 
      {Assign density based on Equation (3.2) in Chapter 3 on page 31}
       $D(u, v) \leftarrow D(u, v) + C_{\widehat{\mathbf{p}_0\mathbf{p}_1}}(\mathbf{q})$ 
return  $D$ ;

```

---

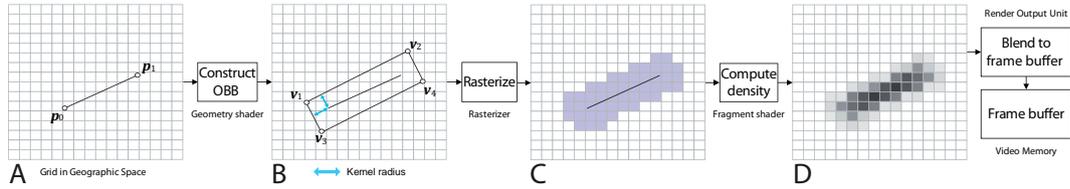


Figure 5.7: The pipeline for computing density fields on a GPU: (A) The trajectory segment  $\mathbf{p}_0\mathbf{p}_1$  given in geographic coordinates. (B) The segment mapped to map coordinates using the cylindrical equal-area projection and its OBB  $\mathbf{v}_1\mathbf{v}_2\mathbf{v}_3\mathbf{v}_4$  at distance  $r$ . (C) The fragments inside OBB. (D) The density field computed for each fragment in OBB, which is added to the density field.

and removing trajectory segments is trivial and that the regular graphics pipeline elements, such as the rasterizer and the render output unit, can be used without explicit synchronization to obtain high throughput.

Our approach is given in Algorithm 2 and illustrated in Figure 5.7. We first construct a texture according to the subdivision of our geographic space into uniform cells with equal area. For each vessel  $o$  we render its trajectories to this texture that ultimately represents the density field. For each trajectory we handle all line segments between consecutive tuples  $\alpha_i$  and  $\alpha_{i+1}$ . Line segments are processed in parallel by geometry shaders that construct, given a line segment  $\mathbf{p}_0\mathbf{p}_1$ , an oriented bounding box (OBB) for the points within distance  $r$ , the kernel radius, from the rhumb line  $\widehat{\mathbf{p}_0\mathbf{p}_1}$  [2] and return four vertices defining an OBB (see Fig. 5.7B). Using the rasterizer the OBB is then subdivided into fragments (see Fig. 5.7C), representing the cells to whose density the object  $o$  contributes. For each fragment  $(u, v)$  a fragment shader is run that computes the geographical coordinates  $\mathbf{q}$  of the fragment (see Fig. 5.7D) using the inverse of the cylindrical equal-area map-projection  $G$ . Finally, the density contribution  $C_{\widehat{\mathbf{p}_0\mathbf{p}_1}}(\mathbf{q})$  is additively blended into density field  $D$  by the render output unit.

Some issues are not handled in Algorithm 2. First, for reasons of accuracy almost

stationary objects are handled different in the sense that their density is computed by drawing a texture mapped fat point with radius  $r$ . Second, the standard formula for computing the great circle distance, which is inaccurate for points at small distances, is replaced by an approximation that is stable for such points. Third, numerical integration of Equation (3.2) on page 31 uses Simpson’s rule. And finally, kernel evaluation is a texture look up, which allows to us to use arbitrary finite-support kernels.

### 5.3.2 Performance

The pictures and performance tests in this chapter have been generated on an Intel Core i7 CPU at 2.8 GHz with 6 GB of RAM memory and an NVidia GeForce GTX 285 with 1 GB of video memory. The computation time of visualizing a single density field depends on the size of the density field, the kernel radius, the number of steps in the integral approximations, and the size of the data set. For a typical data set, containing data from a single day in the North Sea with 1390 objects and 306,521 data points, a density field of 1250x1020 cells, representing an area of 125 km by 102 km, is computed and displayed on a screen of 900x900 pixels in, on average, 0.77 seconds for a kernel radius of 2 km. This does not give full interactive rate, but this can be accomplished by diminishing the accuracy of the density map *during interaction*. There are several ways to do so. First of all reducing in the above example the density map to 125x102 cells and using Catmull-Rom spline interpolation for intermediate values reduces the computation time to 0.13 seconds. The cell size is then still half the kernel radius and the overall appearance of the image remains intact. Also the number of numerical integration steps can be reduced from the default 10 steps to 5 steps to gain up to a factor of two in performance while still obtaining acceptable results.

The interactivity could suffer when computing multiple density fields for a single density map. However, in practice, this is not problematic. If for the density map the data has been split in several subsets, one for each density field, the total computation time is approximately the computation time of a density field using the complete data set. If the density map requires multiple density fields using the full data set, the interaction is mostly such that only one density field needs recomputation and in this case only the initial computation of the density map is relatively slow.

## 5.4 Use Cases

Density maps are intended for expert users exploring distributions of attributes along trajectories. In the exploration, the user interacts with the distribution maps, though the framework is extensive in fine tuning for optimizing details. One of the possible outcomes of the exploration is a density map, which can be used as a static overlay in addition to the regular operational maps containing live traffic for supporting monitoring tasks. To show the expressiveness of our framework, we have defined various use cases towards open problems in the analysis of moving objects with a density approach. The use cases are illustrated with data from vessels and pedestrians.

### 5.4.1 Vessel Traffic

Historical AIS data [83] from vessels can be used for planning of the spatial usage of coastal areas, for instance by validating whether or not all ships have obeyed the rules. The data set contains both route-bounded vessels (e.g., tankers, cargo ships) and non-route-bounded vessels (e.g., pleasure craft, tugs), hinting that our method works for both constrained (e.g., cars, trains) and unconstrained trajectories (e.g., animals, pedestrians).

#### Temporal Aggregation

In a vessel density map (see Chap. 3), the order in which movements take place is lost, since all trajectories are convolved with the same smoothing kernel. With the new density maps, we can vary the kernel radius and kernel weight over time. Figure 5.8A shows a single day of vessel traffic in front of Rotterdam harbour with four subsets of six hours. During the day, the kernel radius is decreasing, while the weights are increasing, and by using density aggregation  $\mathcal{D}_{\text{addition}}$  various moments in time are distinguishable. The

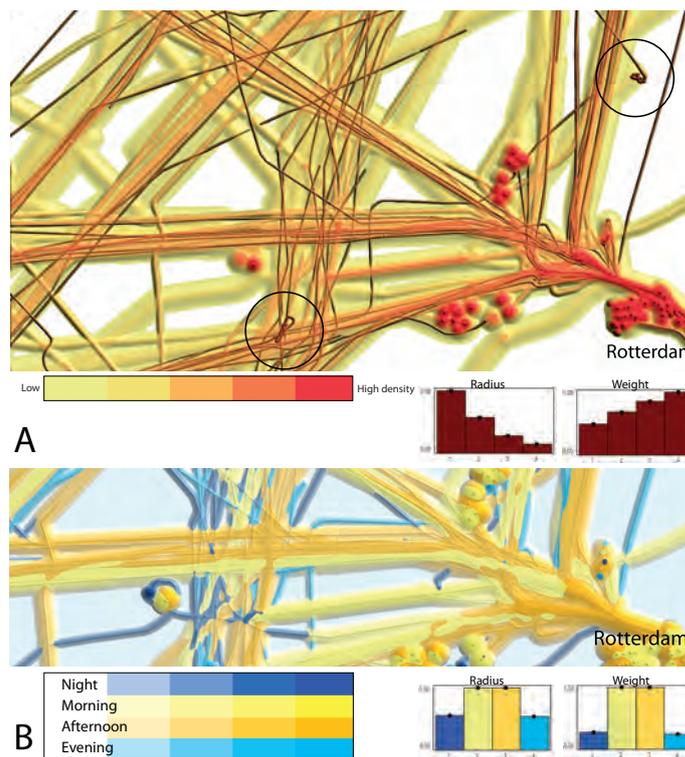


Figure 5.8: An aggregation of four density maps each covering six hours of a day starting at mid-night, using (A) addition as density aggregation and (B) max-blending as image composition.

resulting density map consist of a rendering in both color and illumination of a single aggregated density field using  $\mathcal{J}_{\text{multi}}$  image composition.

In Figure 5.8A we see that the subset in the evening, shown with small and dark trajectories are highlighted, while the others serve as a context. Noticeable movement patterns in these trajectories in the evening that were not visible before are encircled. In the circles we clearly distinguish relatively high density along narrow tracks indicating slow moving night ships. Notice that we can also show variations over time using image composition  $\mathcal{J}_{\text{max}}$  as displayed in Figure 5.8B.

### Anomaly Detection

A density field of a large amount of vessels represents the nautical history in an area, indicating, for instance, which movements are usual. By comparing other trajectories with this density field, we can determine abnormal behaviour in areas where the density field values are low. Density maps can be used to show these anomalously behaving vessels. Figure 5.9 shows the result of the density aggregation  $\mathcal{D}_{\text{anomaly}}$  of two density fields: one with six days of data between Amsterdam and Scheveningen and a density field with the traffic from the last two hours, which can be animated by varying the current time. For the latter one, the kernel radius is decreasing backwards in time, i.e., the head is the most current position. The resulting density field is shown with  $\mathcal{J}_{\text{single}}$  image composition and displays potential anomalies in color from white (none)

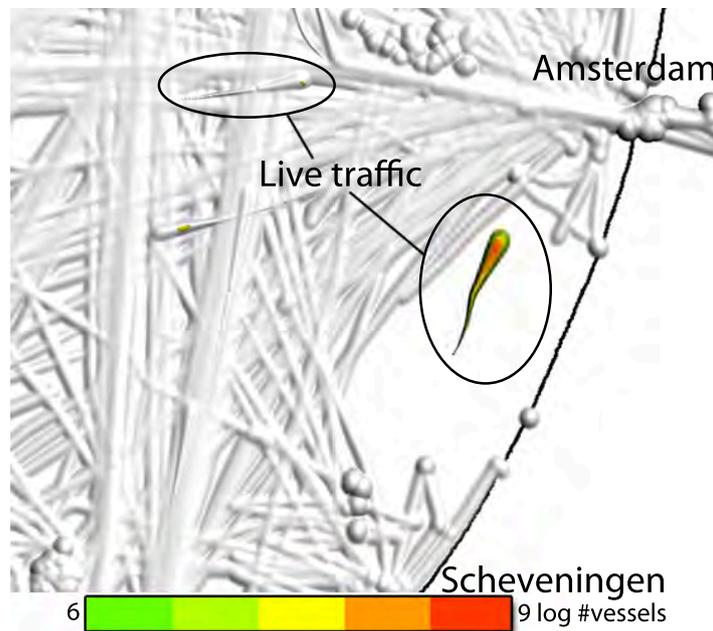


Figure 5.9: A vessel sailing between Amsterdam and Scheveningen is marked as anomalous, since normally no vessels sail in this area.

and green (low) to red (high) in the shaded context of all data. This example shows how density-based anomaly detection can be used in a real-time system.

### Stopping Areas

A vessel density map highlights anchoring zones as groups of dark dot-like parts of trajectories, as shown in the rectangles in Figure 5.10A. However, similar dark features popup for other slow movements, which are not in anchoring zones. In this use case, we try to find explanations for what happens in these areas.

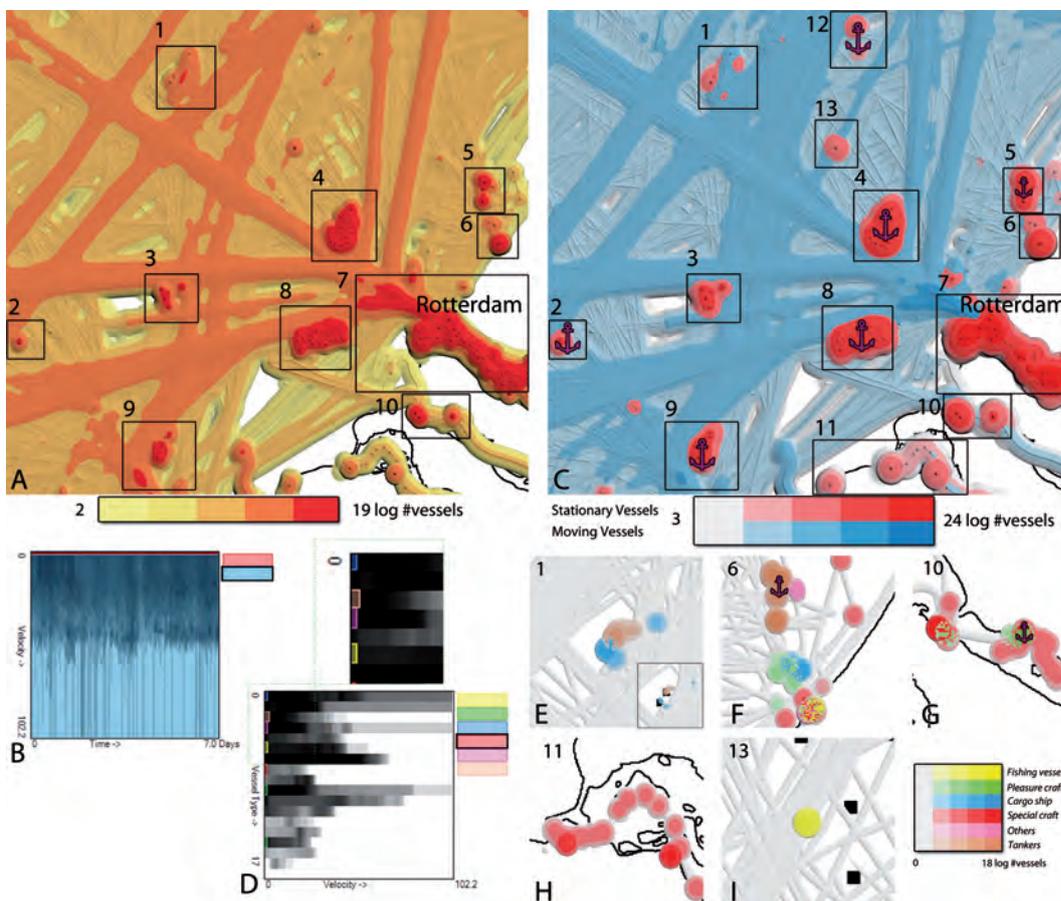


Figure 5.10: *Classifying behaviour of slow moving vessels in front of Rotterdam harbour during one week. In (A) a vessel density map is shown and in (C) slow moving areas are isolated using the DM in (B) and anchor zones are marked with an anchor. After defining different types of slow moving vessels with the DM in (D) we can figure out what happened in the zoomed in areas (E...I), which correspond with the numbers in (C).*

First, we isolate the areas with slow movements by defining two subsets using the DM shown in Figure 5.10B; a red subset for stationary vessels and a blue one for moving vessels. Using image composition  $\mathcal{J}_{\max}$  we show the colors assigned to the subsets in the density map of Figure 5.10C. The official anchoring zones are marked with an anchor. Area number 7 is Rotterdam harbour. The other areas with slow movement need additional inspection.

The vessel type and the way it moves often explains what happens. Therefore, we change the time axis of the DM to velocity and change the velocity axis to vessel type (see Fig. 5.10D) and define six subsets with slow moving vessels of different types. The colors of the subsets in the DM correspond with the legend on the bottom right of Figure 5.10. To find the various subsets available in these small areas we use  $\mathcal{J}_{\text{block}}$  for image composition. In Figure 5.10E we see a cargo ship and a nearby tanker. When decreasing the kernel radius black squares indicating oil platforms become visible (see inset) and actually explain the slow moving vessels. The surroundings of the harbour of Scheveningen with its popular beach are shown in 5.10F. Some typical vessels close to the coast are visible via a manually defined color map: fishing vessels in yellow, pleasure craft in green, cargo ships in blue, special craft, e.g., a rescue vessel in red, and other crafts in pink. Noticeable in this area is a cargo ship relatively close to the beach, since such ships are not expected close to the coast. In Figure 5.10G, there are two hotspots with small vessel types; the one on the right is the harbour of the city Hellevoetsluis, while the left one is near the Haringvlietdam. The existence of a lock explains the concentration of waiting ships on the bottom left-hand side of the Haringvlietdam and also explains why only small vessels sail in this neighbourhood. By clicking on vessels we find in Figure 5.10H that the special crafts are a dredger at work (left) and a law enforcement vessel (right). In the course of its duty the latter stops multiple times. Finally, Figure 5.10I shows a fishing vessel stopping in a sea lane.

### Risk Assessment

Some vessels are more dangerous than others. In this use case, we create a density map showing the possible risks of various vessel types. In the DM we put the vessel type and the area on the axes and define the following subsets, with 'large' being a larger area than 9000 m<sup>2</sup> and 'small' the rest: Large cargo vessels (blue), large tankers (purple), small passenger ships (pink), small high speed craft (orange) and other type of small ships (green). The color map is given by the default color map mentioned in Section 5.2.4. The color mapped densities are composed with  $\mathcal{J}_{\max}$  to show the most dangerous types in a certain area. To illustrate possible risks of dangerous vessels, we have identified three classes of danger and we have assigned a large kernel with high weight for more dangerous vessel types. In this risk density map we can see that the shoreline is not put in danger, and that dangerous tankers take the left route, while less dangerous cargo vessels take the route closer to the shore.

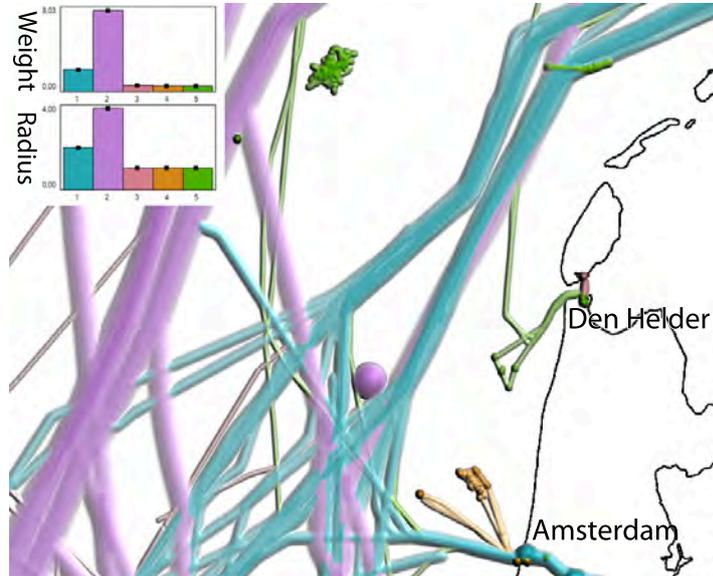


Figure 5.11: A risk map with various vessel types: large cargo vessels (blue), large tankers (purple), small passenger ships (pink), small high speed craft (orange), and other type of small ships (green).

## 5.4.2 Pedestrians in Urban Planning

Multivariate trajectories appear in other domains as well. In this use case we analyze pedestrian movements as an urban planner [149]. We take a look at a data set of pedestrian movements in the city center of Delft over a period of four days. The temporal resolution of this data is similar to the vessels, a sample every couple of seconds. However, the paths are entangled, due to the spatial noise of the small GPS devices, and hence an aggregated view tends to provide more insight than a detailed view. The pedestrians have filled out a questionnaire explaining the purpose of their visit: shopping, tourism, leisure, or something else. We would like to find areas of interest that attract visitors with a certain purpose and see if they appear in the expected areas. To do so, we divide our data set into four density fields, one for each purpose, with equal weight and kernel radius. Like in the previous use case, the colors are chosen with the automatically generated default settings. We combine the density fields using the image compositions for single-hue color maps (see Fig. 5.12A-E), but notice that these result in unreadable images, since in every location many density fields have non-zero values. With the  $J_{\text{opacity}}$  (see Fig. 5.12A) only areas with a single density field defined are readable, while the other places give undistinguishable colors. With  $J_{\text{max}}$  (see Fig. 5.12B) the picture is clearer and the most dominant purpose is visible, however “shopping” in blue is too dominant to make the analysis useful. When using  $J_{\text{block}}$  (see Fig. 5.12CD) we can better distinguish the colors in the large blocks, since the weak distribution of the cones in the human eye prevents to see the color of small patches. The difference between a uniform distribution (Fig. 5.12C) and a distribution proportional

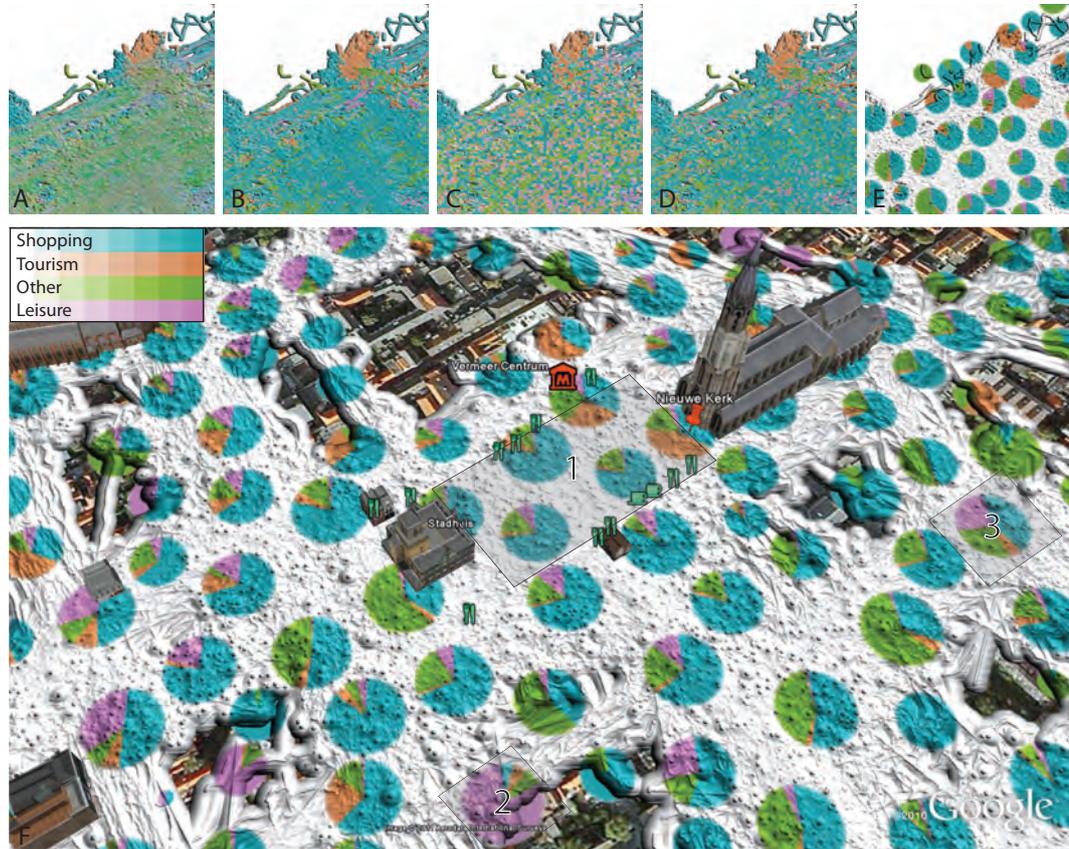


Figure 5.12: Pedestrians around the market square in Delft divided over four visit purpose categories: shopping (blue), tourism (orange), leisure (purple), and other (green). We combine the four density fields using the following image compositions: (A) opacity-blend, (B) max-blend, (C) block with uniform probability, (D) block with probability proportional to density value, and (E) pie. (F) A density map with icons in Google Earth with several areas of interest marked.

to the density values (Fig. 5.12D) for the randomization becomes clear; in the latter the more dominant density fields are better visible. The proportions between density fields are visible, but with  $J_{pie}$  (see Fig. 5.12E) the proportions are better quantifiable, however the orange trend in the middle is less visible than in Figure 5.12A upto D.

In Figure 5.12F a density map with icons is integrated into Google Earth [61], a popular GIS. We see a number of areas of interest where the distribution of purposes is biased towards one or more particular purposes. Using Google Earth as an exploration tool, we can find explanations for these biases. For example, at the market square in area 1 we see an increase in “tourist” activity, near the church and an art exhibition center (Vermeer Centrum), while the “other” category becomes more dominant around

the city hall (Stadhuis). Due to the presence of several diners and stores, the “shopping” and “leisure” categories are also present. In area 2 and 3, we can see the “leisure” category becoming more dominant, explained by the presence of a gallery and a number of bars and cafes. Overall, using our method in Google Earth, we find a relationship between an area and the visit purpose of pedestrians by using additional data sources available in the tool. This may aid an urban planner in making decisions.

## 5.5 Conclusion and Future Work

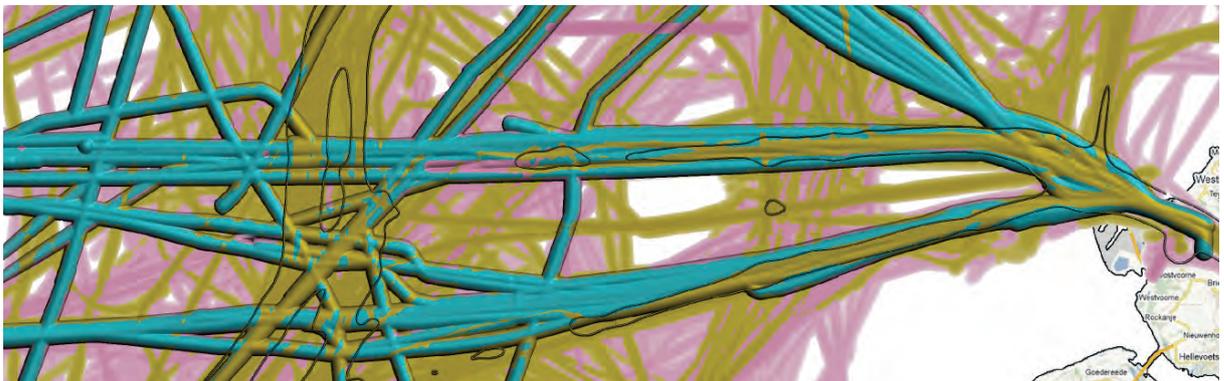
We have presented a method to explore multivariate trajectories with density maps. To support massive real-world trajectory data sets, we have significantly improved the density field computation time with respect to the one in Chapter 3 by means of a parallelized implementation for high-end graphics hardware. As a result, we are able to quickly combine multiple density fields allowing us to interactively enrich density maps. The combination of density fields takes place in either density aggregation or the combination of images of density fields, which can be interactively defined by the user. Density aggregation is typically useful for quantitative analysis, while image composition enables to make a distinction between subsets. The set of aggregations tends to be fairly complete, though the framework can be extended easily with new aggregations. Applying our method to vessel trajectories allows us to reveal *what* takes place *when*, find anomalously behaving vessels, drill down on attributes to solve ambiguities in density fields, and conduct risk assessments. Besides, pedestrian movements are analyzed to match the purpose of visiting Delft.

In future research, we will further investigate the parameters in our method, the interaction to set them, and possibilities for new visual cues to render these parameters. A user study should answer how effective density maps are for analysis tasks as an extension of the work presented in Chapter 4. Furthermore, direction is an important feature of trajectory data, but has not been addressed yet in density maps in an appropriate way. Finally, we are interested if the method can support analysis of moving objects in other domains, by exploring different data sets.

*We would like to thank Stefan van der Spek for collecting and providing the pedestrian movements data set.*

# 6

## Composite Density Maps



*We consider moving objects as multivariate time-series. By visually analyzing the attributes, patterns may appear that explain why certain movements have occurred. Density maps as proposed in Chapter 5 reveal these patterns by means of aggregations of filtered subsets of trajectories. Since filtering is often not sufficient to express domain knowledge, we propose in this chapter to use expressions instead. We present a flexible architecture for density maps to enable custom, versatile exploration using multiple density fields. The flexibility comes from a script, depicted as a block diagram, which defines an advanced computation of a density field. We define six different types of blocks to create, compose, and enhance trajectories or density fields. Blocks are customized by means of expressions that allow the analyst to model domain knowledge. The versatility of our architecture is demonstrated with several maritime use cases developed with domain experts.*

## 6.1 Introduction

Object behaviour is examined by analyzing object states in multivariate trajectories as they move during a certain period. Visual inspection of attributes along these trajectories may reveal patterns that, for instance, may explain why movements have occurred.

Aggregation is a suitable technique for analyzing massive trajectory data sets [10]. A density map as defined in Chapter 5 is a visual aggregation method, inspired by kernel density estimation [133], that generalizes characteristics of multivariate trajectories. Many relevant analyses, such as anomaly detection and risk analysis, can be conducted with these density maps, but their expressibility is limited. If a fixed architecture is used, density maps only consist of predefined compositions of density fields of subsets of trajectories. However, realistic analyses often require more expressibility.

In Chapter 2.4 we defined two roles: analyst and operator. An analyst is a domain expert who defines, possibly complex, parameterized computations of density maps in order to find certain features. An operator, e.g., a coast guard surveillance operator, visually inspects a computed density map for occurrences of the features and interactively changes the parameters of its computation to facilitate the inspection.

To allow the analysts to express their interest in terms of density fields, the density requires a more complex definition. First of all, attribute values could be taken into account, for instance as a weight. Secondly, additional aggregations may be required, such as an average attribute value. Finally, local structures may need to be emphasized, for instance by highlighting areas where movements are vivid, such as meeting areas. Operators should be able to use the density map defined by the analyst and should be able to vary some basic parameters.

To increase the expressibility, we propose to extend the density field computation with a kernel with varying radius, in a similar fashion as Brunson [26], and a user-definable expression for the density that may take other density fields into account.

---

*This chapter is based the paper “Composite Density Maps for Multivariate Trajectories” [128] presented at the IEEE Information Visualization Conference organized in Providence, Rhode Island, United States in 2011.*

Furthermore, the analyst can combine multiple density field computations to compute a complex density field. With such density fields we can perform more analyses: First of all, we can compute non-spatial deviations from normal behaviour, such as finding relatively slow objects moving against the main traffic flow. In fact, this is a contextual computation, where the environment is given by the data. Furthermore, we can show attribute variations in the density by relating attributes to the kernel radius. Finally, by defining the desired result in a composition of simple computations (e.g., bottom of Fig. 6.6), the analyst keeps an overview while building a complex density field. The operator can use the computation defined by the analyst and vary parameters with automatically generated user interface elements. These improvements together enable the analysts and operators to investigate real-world situations.

The remainder of this chapter is organized as follows. Section 6.2 explains our approach to compute, so called, composite density maps, and in Section 6.3 we elaborate on the implementation. In Section 6.4 we apply our method to real-world use cases and in the final section we draw conclusions and suggest some future work.

## 6.2 Composite Density Maps

We adapt the fixed architecture for density maps (see Fig. 5.1 on page 59) as defined in Chapter 5, and propose a flexible architecture for the density model as shown in Figure 6.1. The new architecture only differs from the old one in the density model. In the data part, the filtering is moved to the density model. The visualization part remains, where rendered density fields are combined to a colored and illuminated height field.

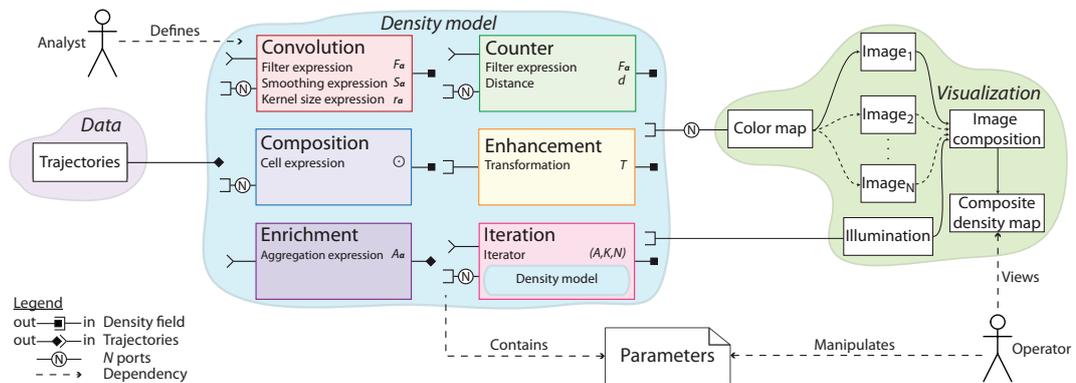


Figure 6.1: The architecture for composite density maps consisting of three parts: data, density model, and visualization. The analyst composes the blocks to compute a set of density fields, while the operator uses the defined model by manipulating its parameters. Each block takes  $N$  density fields and, optionally, trajectories as input to compute a new density field, with the expressions listed in the corresponding block. These density fields are rendered as a colored image and an illuminated height field, which together give a composite density map.

The new density model allows an analyst to define a density computation with a set of composable blocks. An analyst may include parameters in the density computation, which allows an operator to use the definition of computation and change these parameters to tune the visualization. Each individual block creates a new density field and may take density fields generated by other blocks as input. Some blocks also have trajectories as input, which may be filtered with a user-definable expression  $F_\alpha$ , expressed in the attributes of a trajectory  $\alpha$ . The filter  $F_\alpha$  results in a filter attribute  $f_\alpha(t)$ , which is 1 for states  $\alpha(t)$  that satisfy  $F_\alpha$  and 0 otherwise. We define six types of building blocks: convolution, composition, counter, enhancement, enrichment, and iteration, which are discussed in the following sections.

### 6.2.1 Convolution Block

A convolution block takes density fields  $D_i$  and trajectories as input, and is parameterized by expressions for filter  $F_\alpha$ , for smoothing  $S_\alpha$ , and for kernel radius  $r_\alpha$ . The density fields  $D_i$  are used to supply extra attributes  $D_i(t)$  for a trajectory. A density attribute  $D_i(t)$  is obtained by bilinearly sampling the density field  $D_i$  at position  $\mathbf{p}(t)$ . With the expressions and density attributes, the contribution  $C_o$  of an object  $o$  with trajectory  $\alpha$  to the density given in Equation (3.2) on page 31 is generalized as follows

$$C_o(\mathbf{q}) = \frac{1}{T} \int_0^T f_\alpha(t) \cdot S_\alpha(k_{r_\alpha(t)}(\mathbf{q} - \mathbf{p}(t)), t) dt, \quad (6.1)$$

with  $f_\alpha(t)$  the filter attribute and  $\mathbf{p}(t)$  the position in state  $\alpha(t)$ . The smoothing expression  $S_\alpha(\rho, t)$  may consist of the density contribution  $\rho$  given by the kernel  $k$  and the attributes in  $\alpha(t)$ , including the density attributes. The kernel radius expression  $r_\alpha(t)$  may consist of the attributes in  $\alpha(t)$ .

We illustrate the convolution block with a time-based semantic depth of field [91], where recent movements of tankers are considered more of interest than older movements. We can compute such a field allowing only tankers in the filter  $F_\alpha$  and vary both the smoothing  $S_\alpha$  and the kernel radius  $r_\alpha$  along the trajectory with these expressions:

$$S_\alpha(\rho, t) = \rho (t/T)^2, \quad (6.2)$$

$$r_\alpha(t) = 3 - 2.9t/T. \quad (6.3)$$

The non-linear decay of the density and the decreasing kernel radius  $r_\alpha$  in time emphasize the recent movements with a sharp and small kernel as shown in Figure 6.2B. Compared with a normal density map in Figure 6.2A, we can now for instance see the order in which anchor zones are occupied based on the size of the dot.

### 6.2.2 Composition Block

A composition block takes one or more density fields as input and produces a single density field as output. The output is generated based on a user-definable expression  $\odot$  that composes the input density fields per cell. Examples of  $\odot$  are the density aggregation operators of Chapter 5: weighted addition and difference.

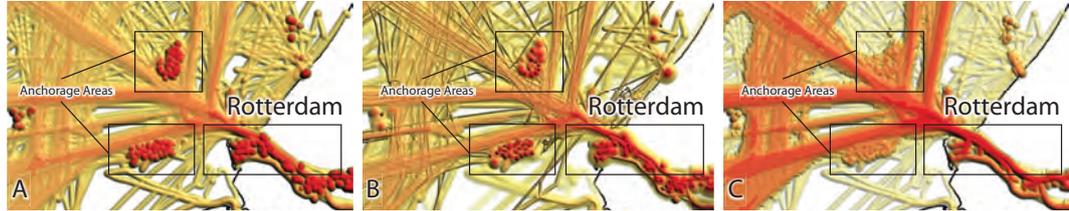


Figure 6.2: Composite density maps of vessels during a week, using (A) a normal density map, (B) a varying kernel radius emphasizing new movements with a small kernel using Equation (6.2), and (C) a varying kernel weight emphasizing sea lanes using Equation (6.4).

### 6.2.3 Counter Block

A counter block returns the number of occurrences of objects that have been active within a given distance from a cell. This building block is useful to quantitatively segment busy areas. For instance, with a count field  $D_c$  generated by a counter block, a density field enhancing traffic lanes can be constructed using a convolution block. If we define traffic lanes as areas with more than 8 moving ships, we can use a constant kernel radius  $r$  and a smoothing expression  $S_\alpha$  defined by

$$S_\alpha(\rho, t) = \begin{cases} \rho v(t) & \text{if } D_c(t) > 8; \\ \rho & \text{otherwise,} \end{cases} \quad (6.4)$$

where  $D_c(t)$  is the number of objects and  $v(t)$  is the velocity in state  $\alpha(t)$ . Figure 6.2C shows enhanced traffic lanes while busy anchorage areas are reduced, since high values occur in areas with many vessels, however, by using the velocity we reduce the contribution for anchoring areas in comparison with the relatively fast lanes.

### 6.2.4 Enhancement Block

An enhancement block encapsulates a set of basic image processing operations, such as Gaussian blur, discretization, and edge detection as shown in Figure 6.3. For these operations, density fields are interpreted as a gray-scale image. Gaussian blur is used to remove distracting noise patterns in density fields. Discretization is used to segment the density field in areas with similar values using a step function. For example, by setting the number of steps to 2, a density field can be thresholded, and turns into a binary density field that can be used as a mask. Edge detection is implemented with a Sobel filter and is used to create contours after discretization.

### 6.2.5 Enrichment Block

An enrichment block adds attributes to trajectories. The state  $\alpha(t)$  is extended with attributes containing aggregations of existing attribute values, defined by an aggregation expression  $A_\alpha$ , such as minimum, maximum, or average. The input of the block is a set of trajectories and the output is the same set of trajectories with a new attribute.

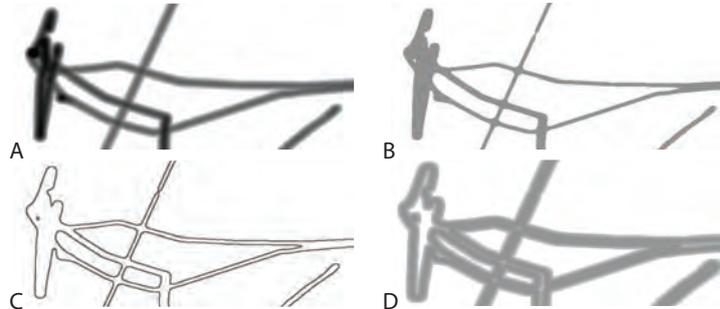


Figure 6.3: Enhancements applied to a density field: (A) a normal density field, (B) the density field discretized using two steps, (C) contours generated from the discretization, and (D) the contours after Gaussian blur.

### 6.2.6 Iteration Block

An iteration block computes a density model in a number of steps. It is parameterized with an attribute  $A$ , an attribute value range  $[0, K]$ , and a number of steps  $N$ . The block computes a density model  $N$  times for  $N$  subsets of the data given by consecutive intervals of attribute  $A$  with length  $K/N$ . Figure 6.4 shows that the output density field is incrementally built by adding a filtered density field for each interval. With an iteration block it is possible to force an order in which density computations are performed. This order is required, for instance, to compute interactions between objects (see Section 6.4.6).



Figure 6.4: Intermediate density fields for 1, 2, and 3 consecutive iterations over (A) time intervals and (B) individual objects.

## 6.3 Implementation

We use a Graphics Processing Unit (GPU) to compute density fields fast. A density field is stored in a texture, where each texel represents a cell, and is computed with shader programs. The blocks and their relations are defined in a script language. To avoid long code listings, these scripts are depicted as block diagrams in this chapter, such as at the bottom of Figure 6.6. For each block, shader code is generated on-the-fly

by instantiating building block-type specific templates with expressions defined in the script. The CPU controls the flow of the computations and sends both the intermediate density fields as textures and the appropriate shader programs to the GPU. Unbound variables in the script are treated as parameters of the density model and can be easily adapted and animated with automatically generated user interface widgets (see Fig. 6.5) similar to the `Manipulate` function in Mathematica [167]. This allows the analyst to include useful parameters in the computation used by the operator.



Figure 6.5: A parameter `minSpeed` can be varied with a widget.

### 6.3.1 Performance

The performance loss of a single density field computation due to the generalization is negligible compared to the one in Chapter 5. A single density field is computed in a split second by a machine equipped with an Intel Core i7 CPU at 2.8 GHz, 6 GB of RAM memory, and an NVidia GeForce GTX 285 with 1 GB of video memory, for given data size (400,000 points), resolution (1000x1000 cells), and kernel radius (2 km). However, experienced users may create complex density fields that may require tens or more intermediate density fields. These computations in total may result in wall times of several minutes. For these complex computations, the memory size of the graphics hardware is a limiting factor for the number of density fields.

## 6.4 Maritime Use Cases

We have constructed use cases together with two Dutch maritime partners, who investigate vessel traffic with different perspectives: One wants to find anomalous behaving ships, while the other is interested in maritime safety. The presented use cases triggered the partners to experiment further with our tooling for their own analyses.

Riveiro *et al.* [121] describe the sense-making process for vessel tracking systems, rephrased with our definition of analysts and operators, as follows: The analyst first performs a foraging loop where the data is explored, filtered, and exploited by extracting information or noticing patterns. Using a visualization and domain knowledge, the operator then gathers information and develops insight into the data set by manipulating the visualization using interaction and parameter settings. Finally, the operator may take action based on newly developed insights or gathered information. We present use cases to show how our method can play a role within this process, whether it is in the process of defining a visualization of normal behaviour, finding anomalies using a rule-based or a data-based approach, or providing interaction to drill down.

These use cases are demonstrated with a vessel trajectory data set in an area in the North Sea near the Dutch coast of approximately 400 by 400 kilometer. The data set

contains 425,591 states covering 2,268 vessels over a period of one week. The trajectories have the following attributes in state  $\alpha^o(t)$ : Vessel type  $s$ , mass  $m$ , course  $c(t)$ , actual orientation  $h(t)$ , velocity  $v(t)$ , and acceleration  $a(t)$ . The course and orientation of a vessel are angles relative to some reference direction.

### 6.4.1 Detailed Harbours

A density field may be an aggregation of a large amount of movement data, and as a result details might be lost, especially around the busier areas, such as harbours and anchorage areas as shown in Figure 6.2A. To emphasize these areas, we could decrease the kernel radius  $r$ , but this also adds more details to areas where we do not require extra detail. With previous density maps, density fields could only be computed with a fixed kernel radius or, at best, the trajectories could be divided into subsets with each assigned a different kernel radius. In this section we describe multi-resolution approaches focussing on details in harbours in a smoothed context.

A composite density map is generated with a convolution block (see bottom left Fig. 6.6) with an adaptive kernel radius that gives us more insight into what is happening in harbour areas. Since vessels generally sail slower in harbour areas, we accomplish this by using the velocity  $v(t)$  to vary the kernel radius, i.e.,  $r_\alpha(t) = \max(\omega v(t), 0.1)$ , with  $\omega$  a scaling factor. Figure 6.6A shows that a small kernel enhances movements in

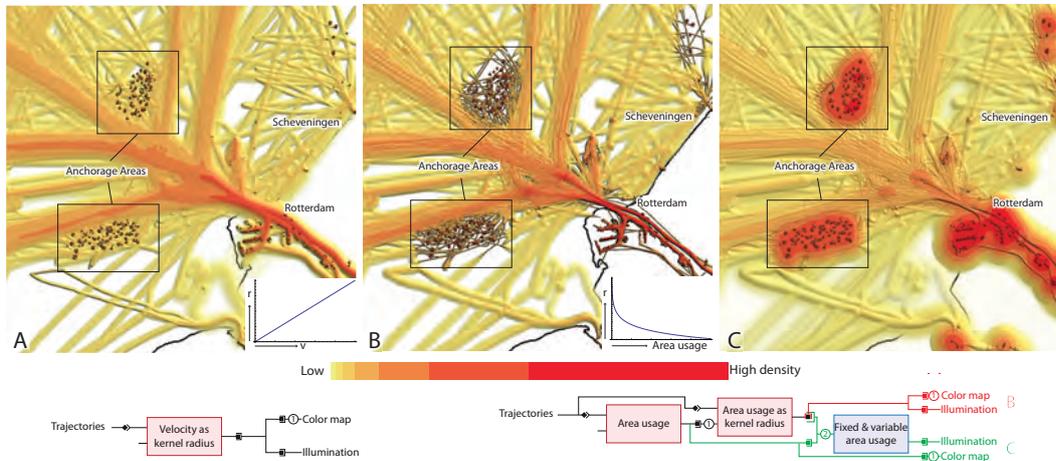


Figure 6.6: *Density maps of Rotterdam harbour using a variable kernel based on (A) the velocity and (B) on traffic density. In (C) we use the density field of (B) as an illuminated height field and show the area usage in color. In (A) more detail is visible in areas with slow moving or stationary vessels, such as inside the harbour and in anchorage areas. In (B) more detail is visible in dense areas such as the harbour mouth, the anchorage areas outside the harbour as well as the sea lanes. The block diagram for (A) is shown on the left, while the block diagram for (B) and (C) is shown on the right, with (B) with black and red links and (C) with black and green links.*

anchor and slow-moving areas while a large kernel blurs fast movers.

One of our partners has expressed interest in a variable kernel based on traffic density such that movements in traffic lanes and other busy areas also have more detail through a smaller kernel radius. This is expressed in a block diagram (see bottom right Fig. 6.6), by first generating a density field  $B$  with a constant kernel radius in a convolution block ‘Area usage’ to get an overview of the general area usage. The result is then fed to a second convolution block ‘Area usage as kernel radius’ with a varying kernel radius based on area usage defined as follows:

$$r_{\alpha}(t) = \max(R - \omega \log(1 + B(t)), 0) \quad (6.5)$$

where  $R$  is a maximum kernel radius,  $B(t)$  the density attribute ‘Area Usage’, and  $\omega$  is a parameter. The second block is connected to both the color map and illumination component, as shown in Figure 6.6B. We are able to infer from the amount of detail where the busier areas are. Figure 6.6C shows an example where the detailed trajectories reveal the actual structure of these areas with slow movers. This is done by separating the varying kernel in the illuminated height field and the area usage in the color map. We obtain a density map that is more smooth and therefore better legible.

### 6.4.2 Drifters

A dangerous situation may arise when a vessel has an engine failure, becomes uncontrollable, and starts drifting. A vessel is said to be drifting when it is moving slowly (i.e.,  $3 \leq v(t) \leq 5$  knot) and its course  $c(t)$  and actual orientation  $h(t)$  have a significant difference (i.e.,  $> 30^\circ$ ) as depicted in Figure 6.7A. These specific values may be tuned by domain experts using the widgets like in Figure 6.5.

To visualize potential drifters, we create a block diagram as shown in Figure 6.7, where a single convolution block is created with  $S_{\alpha}(\rho, t) = \rho |c(t) - h(t)|$  and a filter  $F_{\alpha}$  is used to select only slow movers. The resulting density field is connected to the color component. For contextual information, we also create a density field for the entire set of trajectories and connect it to the illumination component. This reveals a number of potentially dangerous situations in the data set. In Figure 6.7B, we show an overview of the drifters we find with this block diagram and zoom in on several particular cases.

In Figure 6.7C we can see a vessel with drifting patterns around its turns. Closer investigation, by dragging an area with the mouse, which shows more information about these ships in a list, reveals that this is a research vessel which is expected to make sharp turns. In Figure 6.7D, we see potential drifters in the areas shown in red. By clicking in this area, we notice that these trajectories are caused by a single cargo vessel that originated from the North, loitered in the area for approximately two days and then proceeded into the harbour of Antwerp. It is possible that the vessel was too early and had to wait two days before it could head to its destination harbour. This vessel has most likely been drifting on several occasions while it was waiting. The vessel appears to alternate between drifting in a South-West direction and actively moving in a North-East direction. In Figure 6.7E cargo vessels move between ocean platforms, which are visualized as black squares. These vessels are most likely resupplying ocean

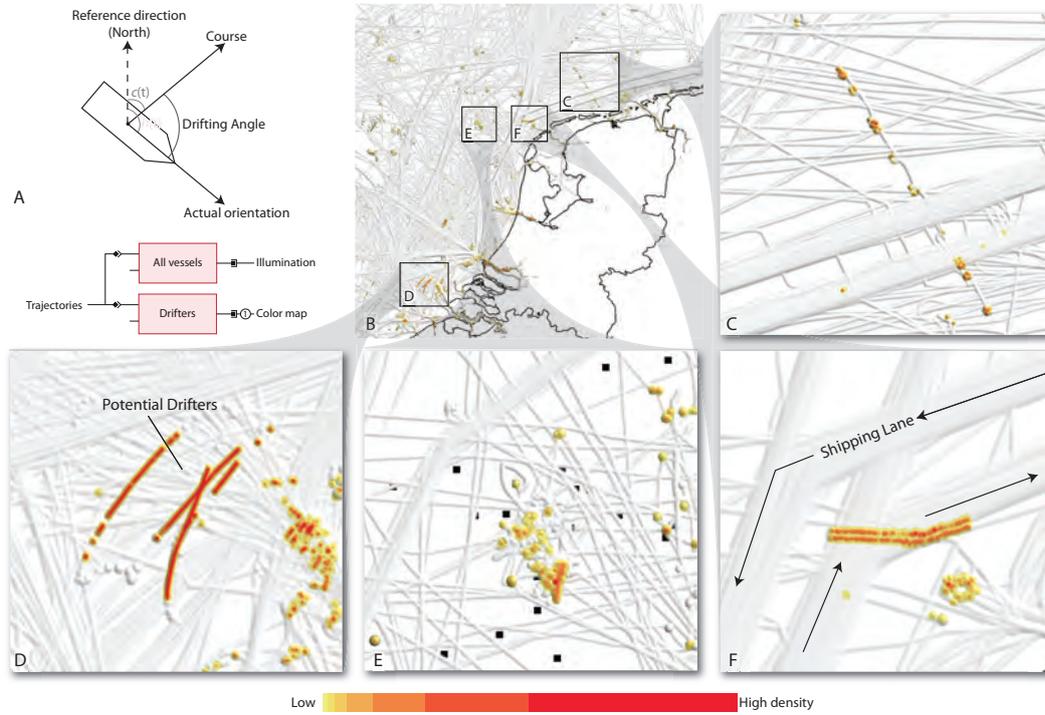


Figure 6.7: (A) The course  $c(t)$  and actual orientation  $h(t)$  of a vessel. (B) An overview of all drifter movements shown in color with all data shown in the illuminated height field. In areas (C...F) we see several close ups of a number of unusual drifter patterns.

platforms and making short stops and sharp turns. In Figure 6.7F we see two parallel drifter trajectories. Since these trajectories occur within a busy area, this may be a dangerous situation, however, again it is a research vessel.

### 6.4.3 Sea Lanes

Traffic Separation Schemes (TSSes) are defined in busy areas to guide large vessels by means of sea lanes. In the North Sea, several TSSes are in effect, but they are not mandatory for all vessels. We separate the sea lanes and other frequently used routes from other movements to get an overview of normal shipping traffic (see Fig. 6.8).

To do this, we partition the data into  $N$  ( $N \geq 8$  to avoid side-effects in crossings) sectors  $\eta$ , each representing a course range. For each sector, we compute a density field and sum the resulting fields into the final field shown in 6.8B. The partial lanes are masked with a thresholded smoothed count field to obtain only busy lanes. We filter out all movements with a course that differs from the average course by at least some constant using an average course field. The 'Average course' field is computed

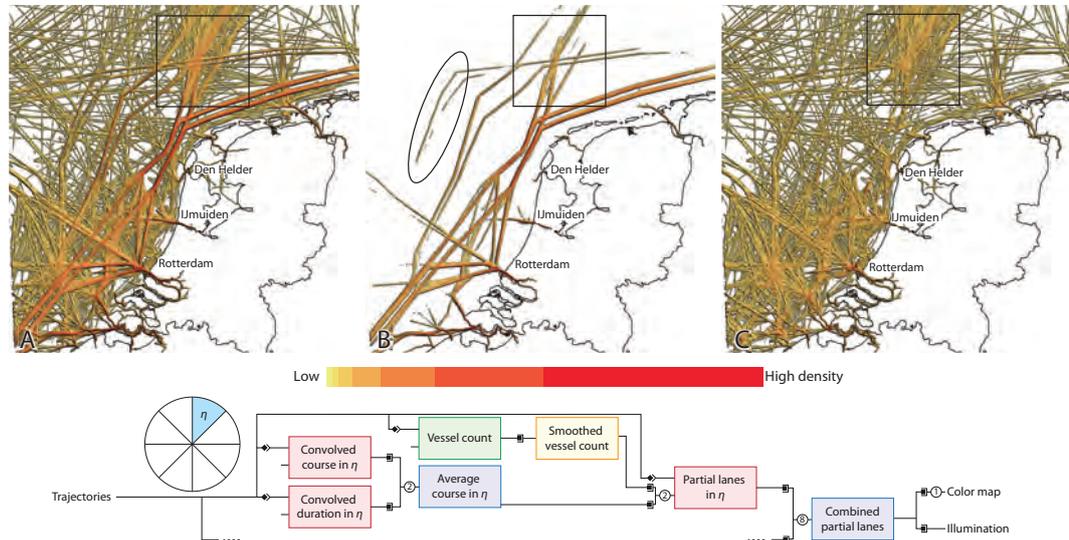


Figure 6.8: A density map (A) containing all movements, (B) containing only movements over sea lanes with a maximum of 10 vessels and a minimum course deviation of  $10^\circ$ . (C) The complement of (B), showing traffic outside of the lanes. The block diagram is shown at the bottom.

using the cell-wise division of two density fields: a density field ‘Convolved course’ using the smoothing function  $S_\alpha(\rho, t) = \rho c(t)$  and a density field called ‘Convolved duration’ with  $S_\alpha(\rho, t) = \max(\rho, \epsilon)$ , for a small  $\epsilon > 0$  to avoid divisions by zero.

The sea lanes appear to disintegrate in the marked areas in Figure 6.8B. This happens where vessel directions start diverging or where there simply is not enough data to extract a reliable route as shown in the encircled area for a little used sea lane.

#### 6.4.4 Traffic Direction

The TSSes separate vessel traffic heading in opposite directions, but vessels may ignore these schemes and move in a potentially dangerous direction. We use the average course maps generated in the previous use case to see whether there are vessels moving against the main traffic flow or whether there are areas containing major traffic flows in both directions (see Fig. 6.9). To do this, another convolution block ‘Partial risk map’ is added to each sector; it filters out all movements that have a course difference less than some threshold compared to the average course in the cell. The results of these blocks are then summed into a single density field that shows the risk of collision in each area given some threshold. In Figure 6.9 we show an example with a minimum deviation of  $175^\circ$  (Fig. 6.9A) and an example of a minimum deviation of  $125^\circ$  (Fig. 6.9B). The resulting density map can be interpreted as a risk map that shows the distribution of possible collisions. From the average course maps of the previous use case we have only used those with many vessels, since the others show low to no risk of collision.

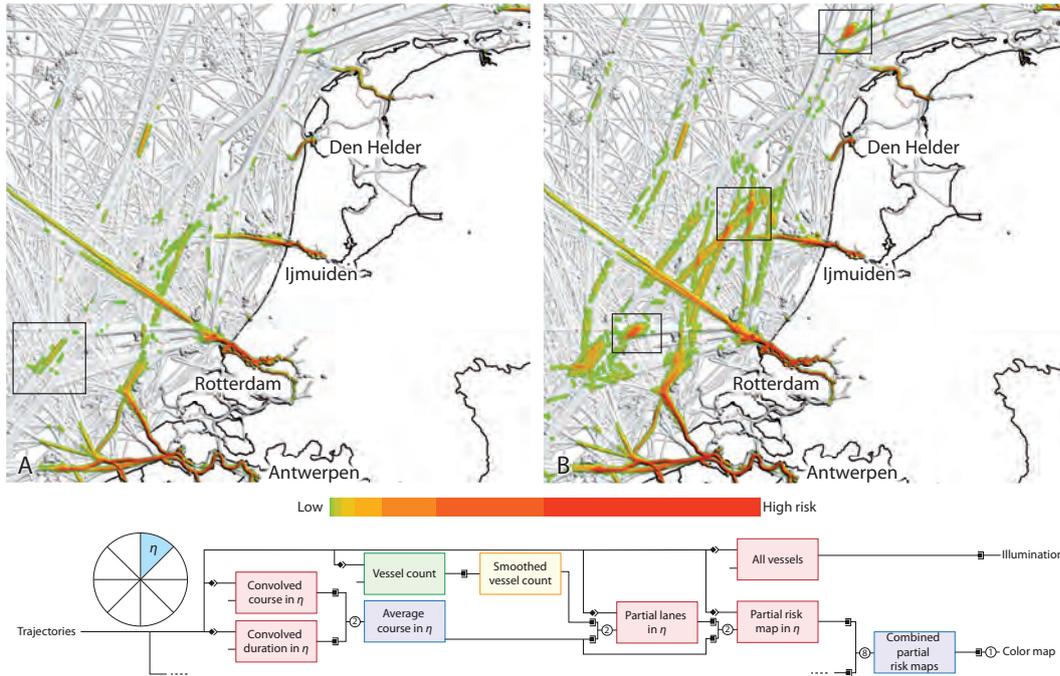


Figure 6.9: Collision risk maps shown with a green to red color map with all data shown in the illuminated height field as context. The maps use a minimum deviation of (A) 175° and (B) 125°. In the marked area in (A) we see a single research vessel moving in the opposite direction of a sea lane. In the marked areas in (B) we see sea lanes intersect with an angle larger than 125°.

In the example with a minimum deviation of 175°, we immediately see hotspots in and near harbours. Since harbours typically have narrow access corridors, vessels are likely to move in opposite directions close to each other, which in turn increases the likelihood of collisions. Note that the kernel radius defines this ‘closeness’ relation. We can also see a small green pattern in a busy sea lane to the West, which is not visible in a standard density map at all. This means a vessel has been moving in the opposite direction within a sea lane, which is potentially dangerous. Closer investigation reveals that this is a vessel moving back and forth over a relatively small area to collect hydrographic data. In the example with a minimum deviation of 125°, we can see a high risk of collision in and near harbours and an increased risk in intersections between sea lanes where the intersecting angle is at least 125°.

### 6.4.5 Risk Analysis

A common task in movement analysis is risk assessment. Each vessel is assigned a risk, which is defined as the probability of an accident multiplied by the consequences of the accident, as described by Koldenhof *et al.* [90]. Both the accident probability and

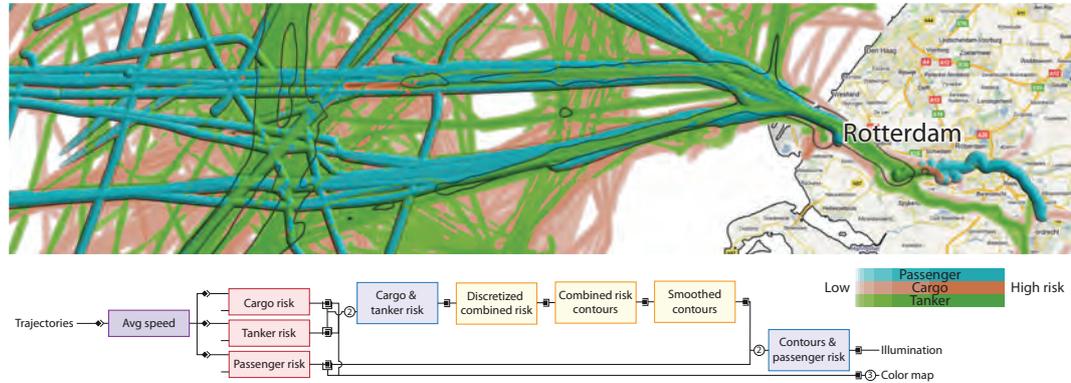


Figure 6.10: An accident risk map of passenger vessels (turquoise), cargo vessels (orange), and tanker vessels (green) in front of Rotterdam harbour. The highest density at a location determines the color of the cell. The areas that need extra attention are highlighted in a separate illuminated height field containing the passenger risk map and contours as ridges. The contours are derived from a combination of tanker and cargo risks marking where they both have a high risk.

consequences are derived from vessel attributes such as size, age, type, cargo, and environmental attributes, e.g., depth or distance to the shore. The risk of each individual ship is then aggregated into a risk map.

This risk model can be implemented as a block diagram (see bottom Fig. 6.10). First we enrich the trajectories with the average speed  $v_{\text{avg}}$  per vessel, which is used in the filter  $F_{\alpha}$  to exclude slow movers from the analysis. We model the risk  $\kappa$  for a single vessel  $o$  with the attribute values in state  $\alpha^o(t)$  as  $\kappa(t) = \omega m v^2(t)$ , where  $\omega$  is a risk factor based on the vessel type  $s$  and the remainder of the risk is based on the kinetic energy function  $E_k = \frac{1}{2} m v^2$ , with mass  $m$  and velocity  $v$ , as a simplified model for the impact of a collision. The risk factor  $\omega$  is high for passenger ships due to potential loss of life due to an accident,  $\omega$  has a medium value for tanker and cargo vessels carrying potentially hazardous materials, and  $\omega$  is low for normal tanker and cargo vessels.

Using this model, we create a density field with  $S_{\alpha}(\rho, t) = \rho \kappa(t)$  for each vessel type (passenger, cargo, and tanker) and visualize these using three distinct color maps (see Fig. 6.10). The colored density fields are combined using the max-blend image composition of Chapter 5, showing the color of the highest density field. The areas with a high risk for both tankers and cargo vessels are invisible, therefore we create contours around them. These contours combined with the passenger ship risk map are connected to the illumination component, which indicates the areas that need attention due to high risks. They are created by discretizing the combined cargo and tanker risk map into a binary field using an enhancement block. The binary fields is then fed into another enhancement block with a Sobel filter, resulting in crisp edges. We smooth the contours with a Gaussian using another enhancement block.

Looking at Figure 6.10, we notice that cargo vessels and tankers sail in less well-defined routes than passenger ships. Despite this high spread, the combined high risk is mainly in the same routes as passenger ships in the sea lanes between West and East,

and, in a sea lane between North and South where they move towards the mandatory sea lanes for merchant vessels. By comparing the contours and the illuminated height field of the passenger ships, this density map allows us to distinguish between areas with a high risk caused by loss of life or caused by financial or environmental damage.

### 6.4.6 Vessel Interactions

Due to the high amount of traffic and the limited manoeuvrability of vessels, the collision risk is relatively high. We define that a vessel  $o'$  interacts with another vessel  $o$  at time  $t$  if  $o'$  is within interaction distance  $d_o(t)$  of vessel  $o$  as shown in Figure 6.11A, where these interactions are then convolved. We propose a method, using our architecture, to visualize such interactions with a flexible definition for  $d_o(t)$ .

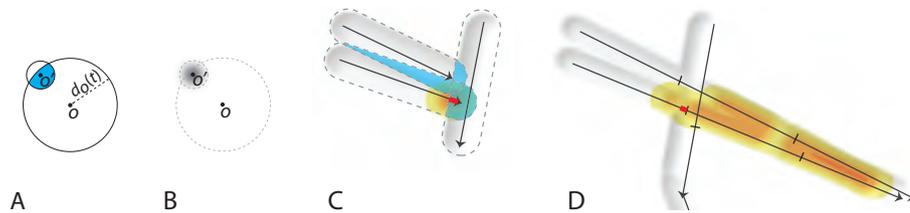


Figure 6.11: (A) Interaction distance  $d_o(t)$  of a vessel  $o$  at time  $t$ . (B) Vessel  $o'$  is within  $d_o(t)$  of vessel  $o$  and is therefore convolved. The vessel interaction procedure applied to (C) a single and (D) three time intervals. Areas where multiple trajectories overlap are shown in blue and the movements within these areas are shown in red and convolved.

Since both velocity and mass reduce the manoeuvrability of a vessel, we express the interaction distance  $d_o(t)$  in a similar fashion as in Section 6.4.5, i.e.,  $d_o(t) = \omega m v^2(t)$ , with  $\omega$  a scaling factor. For kernel radius  $r_\alpha(t) = d_o(t)$ , we obtain a density map as shown in Figure 6.12A: the larger and faster a vessel is, the larger its radius is.

Since continuously sampling each point in time is infeasible, we perform our computations in discrete steps. Therefore, we use the iteration block to find vessels that are within interaction distance of another vessel for separate time intervals. In each iteration step, we use a count block to find overlapping movements. This result is used as an extra attribute for a convolution block in which only movements for which the result of the count block is two or more are convolved as displayed in Figure 6.11C and 6.11D. Using a two minute time interval we obtain the density map as shown in Figure 6.12B.

With this method we can see stretched out patterns marked with arrows in sea lanes where vessels overtake each other. Although these vessels pass each other close by, this is not considered dangerous. Also, tugs and pilot vessels cause large patterns in and near the harbour mouths when they are tugging or guiding other vessels. In both cases, the interacting vessels move in a similar direction. Dangerous interactions are found by subtracting the density fields for these parallel movements from a density field containing all movements.

The data is partitioned into  $N$  sectors  $\eta$ , each representing a range of  $\gamma^\circ$  courses (see Fig. 6.12). For each sector  $\eta$ , we count the occurrences of parallel movements

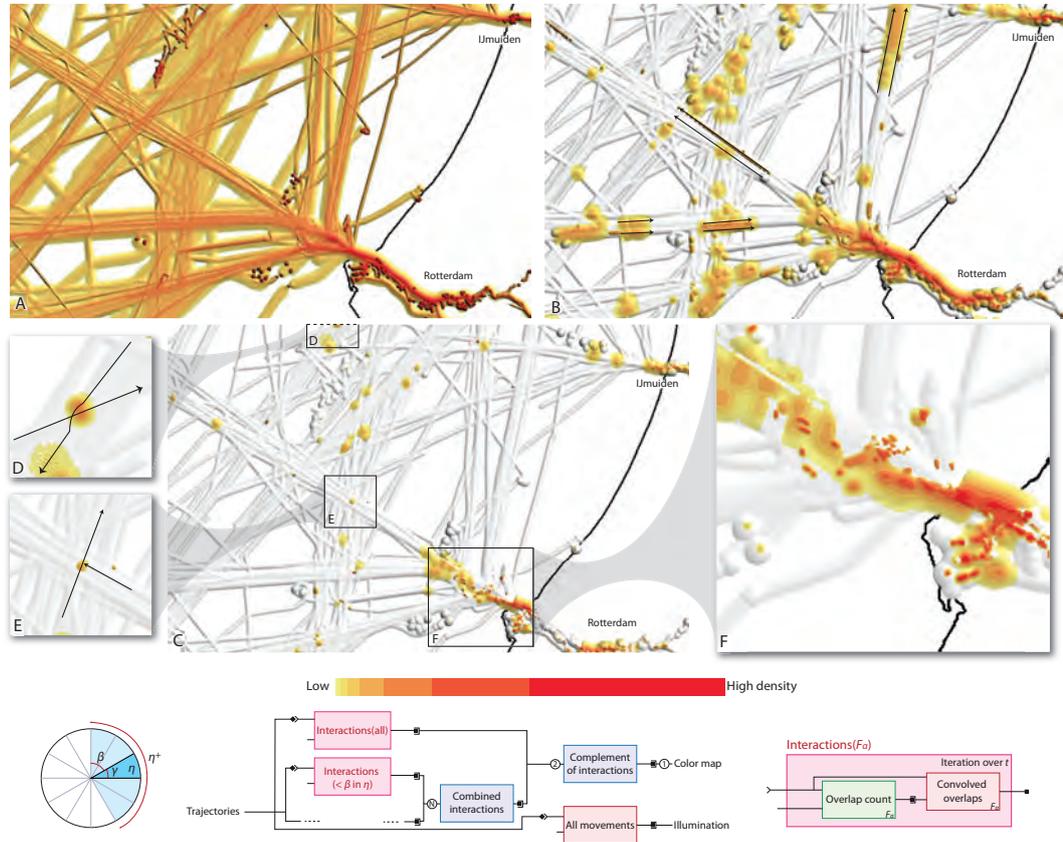


Figure 6.12: Interactions around Rotterdam harbour with  $\beta = 60^\circ$ ,  $\gamma = 30^\circ$ ,  $N = 12$ , and the arrows show vessels overtaking each other. (A) An overview of all vessels using  $r_\alpha(t) = d_o(t)$ , (B) all interactions within a time interval of two minutes in color and an overview of all movements used as illuminated height field, and (C) all vessel interactions with a course difference of at least  $\beta$ . The block diagram for density map (C) is shown at the bottom. Areas (D...F) are enlarged interactions.

within a sector  $\eta$  widened with  $\beta$  on both sides, called  $\eta^+$ . If an object moves with a direction in sector  $\eta$ , then a parallel movement occurs if another object moves with a direction within  $\eta^+$ , and hence deviates less than an angle  $\beta$  with sector  $\eta$ . If the number of parallel movements is two or more then movements within  $\eta$  are convolved.

This results in the density map shown in Figure 6.12C. All areas shown in color now indicate what we have defined as interactions without movement in a similar direction. We take a closer look at a few of them using interactive inspection in our tool. In Figure 6.12D we can see a small dot caused by two vessels passing each other at close range, where the vessel heading in South-Western direction has to slightly change course to avoid the other vessel. In Figure 6.12E we see another small dot caused by

a vessel heading in a North-Western direction having to stop for a vessel heading in a North-Eastern direction. All visible dots in the density map in Figure 6.12C are caused by similar situations, either by vessels passing each other close by without action taken, or by vessels having to perform special manoeuvres to avoid collisions.

In Figure 6.12F we see a number of small dots near the harbour mouth of Rotterdam, which is an area with high pilot vessel activity. Pilot vessels carry trained professionals to guide larger vessels through busy and/or dangerous areas around and in harbours. The pilot vessel latches onto a large vessel and the pilot boards the larger vessel to guide it safely into the harbour. The dots in this area are possibly caused by these pilot vessels latching onto larger vessels.

## 6.5 Conclusion and Future Work

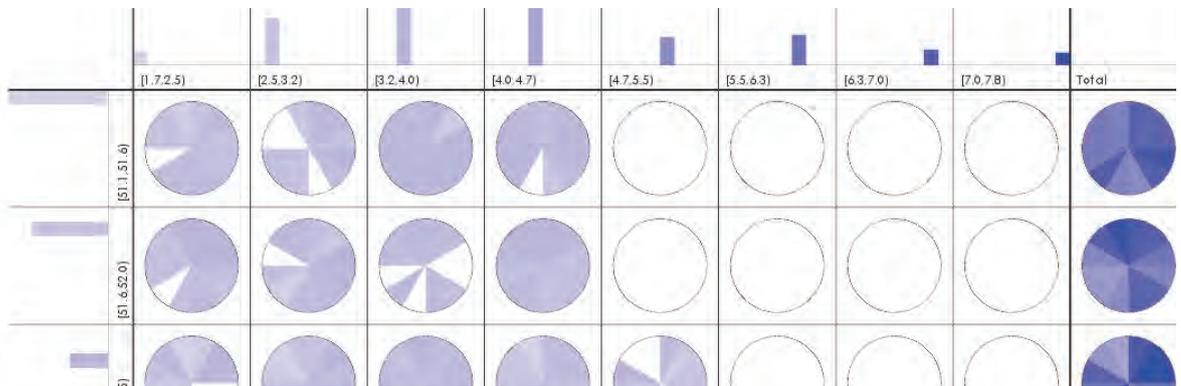
We have introduced a flexible architecture for composite density maps, where a density field is computed by means of a composition of density and trajectory computations. The density field computation itself is extended with a variable kernel radius along a trajectory and user-definable expressions for filtering, smoothing, and composing density fields. This enables to apply domain knowledge as shown with use cases displaying patterns for collision risks, drifters, common routes, and interactions.

To extend this work, we are looking in the following directions. First of all, we would like to include environmental context given by raster maps, such as weather or elevation, and use them as input in the density field computations. We would like to extend the user controls shown in Figure 6.5 with additional graphics to support finding the optimal setting. To increase the performance of the density field computation we may use the one as proposed by Lampe *et al.* [95], where the trajectories are convolved with a constant velocity instead of a constant acceleration. However, it is unclear what the speedup will be when using our multivariate approach. Furthermore, we would like to extend the set of building blocks. User interaction needs to be developed further such that, for example, objects participating in an interesting pattern can be easily identified and traced. And finally, cyclical movements and density fields of non-spatial attributes are of interest.

*COST Action IC0903 MOVE [79] established collaboration between Eindhoven University of Technology, Fraunhofer Institute for Intelligent Analysis and Information Systems, and Maritime Research Institute Netherlands (MARIN). This project is partially supported by the German priority research program on Visual Analytics.*

# 7

## Attribute Exploration



*In this chapter we present an integrated and multi-disciplinary approach for analyzing behaviour of moving objects. The results are combined research of four partners from the Poseidon project [48], which aims to develop new methods for Maritime Safety and Security (MSS) systems to monitor vessel traffic in coastal areas. Our architecture enables an operator to visually test hypotheses about vessels with time-dependent sensor data and on-demand external knowledge. The system includes the following components: abstraction and simulation of trajectory sensor data, fusion of multiple heterogeneous data sources, reasoning, and visual analysis of the combined data sources.*

## 7.1 Introduction

Today’s tracking devices lack the capabilities to capture semantically-rich data related to the patterns we are searching for. Often, only the path of the trajectory is captured, which makes finding patterns complex. Different useful patterns may be found by using additional data, which is obtained by extra sensors, manual labeling, or results from computational methods as suggested by Andrienko and Andrienko [14]. We focus on visually detecting patterns for testing hypotheses about the behaviour of moving objects using multiple heterogeneous data sources containing derived data and existing knowledge about the objects and their context; one of the visual analytics challenges as identified by Thomas and Kielman [142]. For instance, in aviation, an analyst may want to test whether civil airplanes landed on a military airport, which requires sensor data from the airplanes and a geographical context to determine the type of airport.

Our approach is demonstrated in the maritime domain for monitoring vessels in coastal areas and is a prototype for a part of a so called Maritime Safety and Security (MSS) system. Figure 7.1 displays the architecture containing an overview of the tools and interfaces. Three types of low-level data sources are aggregated in our knowledge base (see Sec. 7.3): vessel trajectories obtained by means of the Automatic Identification System (AIS) or simulated with Presto [86] (see Sec. 7.2.1) and web data from various sources about vessels and their (geographical) context. The trajectories are converted into meaningful Simple Event Model (SEM) instances (see Sec. 7.3.1) after having been preprocessed with Piecewise Linear Segmentation (see Sec. 7.2.2) to generalize over the massive amount of sensor data and obtain a smaller, but semantically equivalent data set that can be processed more efficiently. The web data are added to these SEM events as instances in Resource Description Framework (RDF) and stored in an SWI-Prolog-based RDF store (see Sec. 7.3.2). Prolog is ideally suited for reasoning

---

*This chapter is on joint work of the following members of the Poseidon team: Willem Robert van Hage and Véronique Malaisé (Web & Media Group, VU University Amsterdam), Gerben de Vries (Informatics Institute, University of Amsterdam), Jeroen Janssens (Tilburg Centre for Creative Computing, Tilburg University), and Niels Willems (Dept. Mathematics and Computer Science at Eindhoven University of Technology), which resulted in the paper “An Integrated Approach for Visual Analysis of a Multisource Moving Objects Knowledge Base” [165] presented at the AGILE Workshop Geospatial Visual Analytics: Focus on Time in Guimarães, Portugal in 2010 and the extended version [164] with the same title published in the International Journal of Geographical Information Science.*

over logical knowledge facts, such as RDF, because it is a declarative logic programming language. Deduction rules can be used to define behaviour on top of the movement events and by reasoning we build up knowledge, for instance by adding new attributes to a trajectory. The visual interface is an interactive trajectory contingency table enabling a user to find patterns, such as correlations between attributes, in the knowledge base (see Sec. 7.4). The trajectories generated with Presto allow us to evaluate the performance of the system with scenarios containing ground truth (see Sec. 7.5). The last two sections elaborate on the system as a whole and suggest for future works.

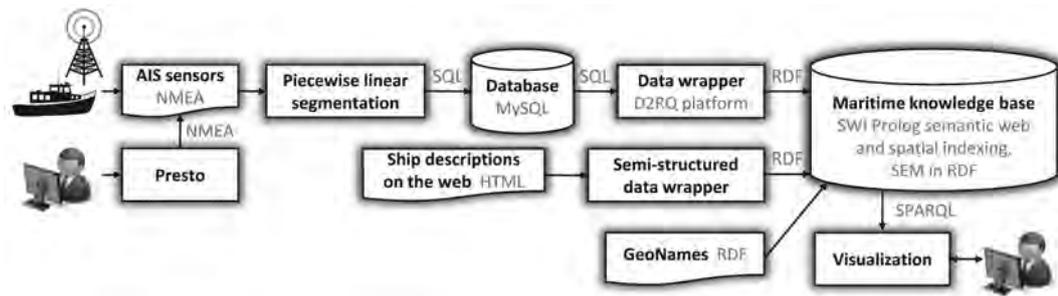


Figure 7.1: The architecture of our approach for the analysis of a multi-source moving object knowledge base as a flowchart. Straight boxes are processes, curved bottom boxes are documents, and cylinders are databases. Interfaces and platforms are shown in light grey.

## 7.2 Related Work

There are some integrated frameworks for analyzing spatio-temporal data sets. However, these frameworks focus on one aspect at a time: heterogeneous data sources [34], visual patterns [168], or mining patterns [23]. We combine these three aspects.

Sánchez *et al.* [127] reason about moving objects in a knowledge base, but its reasoning serves the construction of the actual trajectories, and is not involved in any further analysis. Yan *et al.* [172] proposed a trajectory ontology that allows for querying and reasoning with homogeneous trajectory data, but it is limited by not incorporating other data sources. A knowledge base has been used in only a few visual analytic systems, and none of these focus on spatio-temporal data. Xiao *et al.* [171] visually analyze network traffic with a knowledge representation. The HARVEST system [62] is based on the principle to synthesize knowledge for progressive analysis.

Semantic analysis work comparable to our analysis of moving objects has been reported by Orellana and Renso [111]. This work mainly focuses on describing collective behaviour in OWL, whereas we describe an entire integrated workflow from the level of sensor data to the level of behaviour semantics. Also, we choose to use Prolog as our reasoning tool as opposed to an OWL reasoner. A methodology for modeling faceted *who, what, where, when* data with the CIDOC-CRM event model has been proposed by Hiebel *et al.* [73]. The ontology we use, SEM, is much smaller than CIDOC-CRM.

SEM provides a mapping of all its constructs to CIDOC-CRM. A survey of semantic approaches for the integration of geospatial data has been undertaken by Buccella *et al.* [28]. Hornsby and Joshi [76] combine spatial and temporal semantics of objects that could be applied to the Place and Time facets of SEM Events.

### 7.2.1 Creating Artificial Vessel Trajectories with Presto

An MSS system aims at guiding an operator in finding maritime anomalies, such as vessel traffic violations, illegal fishing activities, and drug smuggling. In real-world data, maritime anomalies rarely occur, therefore trajectories with ground truth are needed to evaluate whether arbitrary anomalies can be detected properly. To this end, the Poseidon partner at Tilburg University has developed an application, called Presto, which enables maritime domain-experts to easily create anomalies, including the ones mentioned by Janssens *et al.* [86]. In contrast to existing simulation applications, such as VR-Forces [156], which impose restrictive behaviour models, our application gives an expert full control over the trajectories.

Figure 7.2 shows the concepts and user interface elements of Presto. The main concept in Presto is a scenario, which can contain one or more trajectories. Each trajectory is defined by several waypoints, which is a location with additional parameters: veloc-

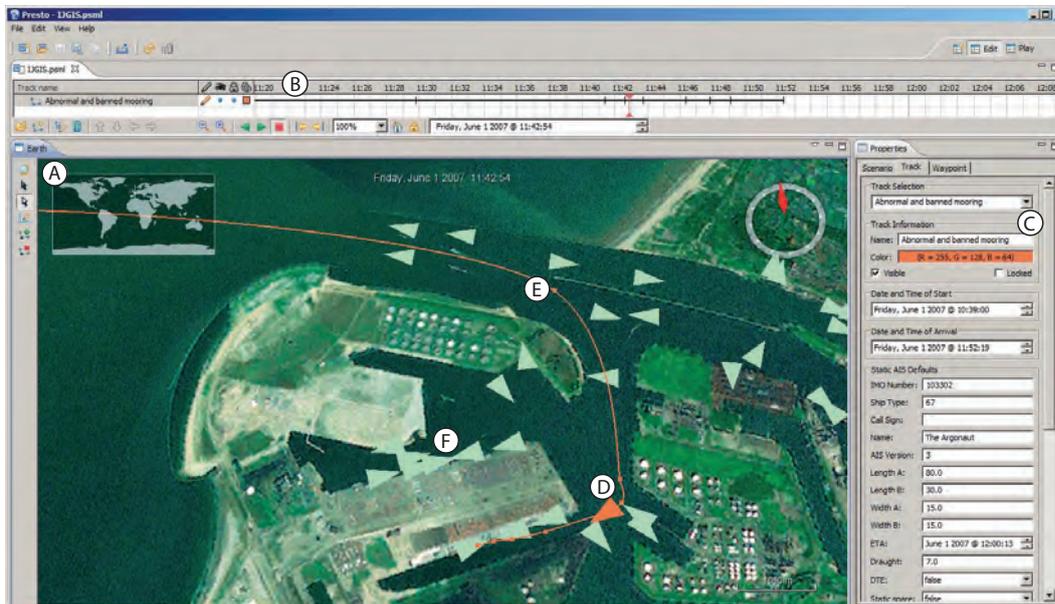


Figure 7.2: The screenshot illustrates the user interface elements and concepts of Presto: (A) the world map, (B) the timeline showing the trajectory in the scenario, (C) the property editor showing the trajectory properties, (D) the current location of the artificial vessel displayed as an orange triangle along the created trajectory, (E) one waypoint of the artificial trajectory as a dot, and (F) the background data at the current moment displayed with a green triangle for each vessel.

ity, time, and curvature. The user interface of Presto consists of the following main elements: a world map (Fig. 7.2A) based on NASA World Wind [107] to position the trajectories and their waypoints, the timeline (Fig. 7.2B) to navigate through time, and the property editor (Fig. 7.2C) to edit specific parameters (e.g., for a scenario: name, author, and description; for a trajectory: name, call sign, and flag; and for a waypoint: location, time, velocity, and curvature). We can load existing, real AIS data as so-called background data (Fig. 7.2F), which serves as a reference when creating new artificial trajectories, so that, for instance, collisions can be avoided or enforced.

A trajectory is generated from the waypoints using the following iterative procedure. Given a location, speed, and time, as provided by the current waypoint, the vessel starts sailing towards the next waypoint for a user-defined time step, after which the location, speed, and bearing of the vessel are updated. This is repeated until the vessel reaches the next waypoint. When the vessel passes the waypoint within the time step, all variables are adjusted such that the vessel continues exactly at the waypoint. Rather than using a linear interpolation for the position between two waypoints, which result in unnaturally sharp turns, we employ Bézier interpolation with control points placed at a location defined by the curvature variable. A lower curvature value causes the control points to be placed further away from the waypoints, resulting in a smoother and larger turn. The speed between waypoints is interpolated using exponential easing. The generated data is converted to data according to the AIS protocol and fused with real-world AIS data, such that the artificial data cannot be distinguished up front from the original data by the system in the subsequent processing.

For the use cases in Section 7.5, we have created a trajectory of a passenger ship that goes to an anchorage area for cargo storage in Rotterdam harbour (see Fig. 7.2). After integration with the other six hundred ship trajectories in the knowledge base, our movement pattern recognition rules can be tested. This should lead to a conflict, since a passenger ship should anchor at a passenger terminal.

### 7.2.2 Piecewise Linear Segmentation

Vessels are constrained in their movement and, consequently, they often sail along predictable courses. However, AIS messages are broadcast frequently, hence many subsequent messages do not differ semantically. Therefore, the Poseidon partner at University of Amsterdam compresses these messages into segments of consistent behaviour, which has the advantage that we drastically reduce computation time for the higher level reasoning (Sec. 7.5).

The basis for our trajectory compression is a Piecewise Linear Segmentation (PLS) method, such as studied by Gudmundsson *et al.* [64], and Cao *et al.* [31]. Also in Chapter 3 we proposed a trajectory compression algorithm, which is similar, but reconstructs the position using the velocity in the second step of the algorithm. These algorithms are based on a well-known, efficient compression algorithm for polylines introduced by Douglas and Peucker [42], but have been adapted to take the temporal component of a trajectory into account. Figure 7.3 illustrates a generic version of the PLS-algorithm. This algorithm compresses a sequence of tuples  $\alpha_{\text{start}}, \dots, \alpha_{\text{end}}$  by finding the tuple  $\alpha_{\text{max}}$  that has the largest error  $\epsilon_{\text{max}} = E(\alpha_{\text{max}}, \alpha'_{\text{max}})$  with respect to

the reconstructed tuple  $\alpha'_{\max} = R(\alpha_{\text{start}}, \alpha_{\max}, \alpha_{\text{end}})$ , for some error function  $E$  and reconstruction function  $R$ . If  $\epsilon_{\max}$  is larger than some given error  $\epsilon$ , then we select  $\alpha_{\max}$  and recursively compress the sequences  $\alpha_{\text{start}}, \dots, \alpha_{\max}$  and  $\alpha_{\max}, \dots, \alpha_{\text{end}}$ , otherwise the algorithm stops and accepts  $\alpha_{\text{start}}, \alpha_{\text{end}}$ .

The concepts of *stop* and *move* have been argued to be important semantic movement primitives [137]. A big part of the higher level reasoning in Section 7.5 is based on these concepts. To better preserve stops and moves we have investigated and created a two-stage variant of the PLS-algorithm. In the first stage, tuples  $\alpha_i$  are selected, using the PLS-algorithm, based on the velocity of the vessel. The velocity  $v'_{\max}$  is reconstructed with  $R$  being the linear interpolation between  $v_{\text{start}}$  and  $v_{\text{end}}$  at  $t_{\max}$  and the error  $E$  is the absolute difference between  $v'_{\max}$  and  $v_{\max}$ . Then, in the second stage, the positions between each pair of points selected in the first stage are compressed, with PLS, as a polyline, where  $E$  is the Euclidian distance between  $\mathbf{p}_{\max}$  and  $\mathbf{p}'_{\max}$ , which is reconstructed by  $R$ , the projection of  $\mathbf{p}_{\max}$  on the line  $\mathbf{p}_{\text{start}}\mathbf{p}_{\text{end}}$ .

To test whether the two-stage variant retains more stops at the same compression rate (i.e., the amount of ignored data) than other variants of PLS-based compression techniques, we created a data set of vessel trajectories with 281 manually labeled stops. We define a vessel to be stopped between two consecutive tuples  $\alpha_i$  and  $\alpha_j$  if:

$$\frac{\|\mathbf{p}_j - \mathbf{p}_i\|}{(t_j - t_i)} \leq \theta, \quad (7.1)$$

where  $\theta$  is a threshold velocity. From experience, we noticed that in the maritime domain  $\theta = 0.05$  knots ( $\approx 0.093\text{kmh}^{-1}$ ) is a good threshold for retaining stops. We give a small excerpt of the experiments that we have conducted. From earlier work on trajectory compression algorithms we selected the two error measures that retained the most stops. We used  $E_\mu$  from Gudmundsson *et al.* [64], for which we selected the best value for  $\mu$  (in terms of stop retention) that we could find, and  $E_t$  from Cao *et al.* [31]. With both these error measures a trajectory is treated as a polyline in 3D, with  $t$  being the third dimension. The  $E_\mu$  measure is an Euclidian distance in this 3D space, where  $\mu$  is the weight of the time dimension. The error measure  $E_t$  focusses purely on the temporal difference between the original point and its 2D interpolation. In Figure 7.4 we have plotted the performance of PLS with error measures  $E_\mu$  and  $E_t$ , and our two-stage variant. For the different algorithms the compression rate versus the stop retention rate is shown, with 100% being all stops retained.

We say that a stop is retained by a compressed trajectory if there are two consecu-

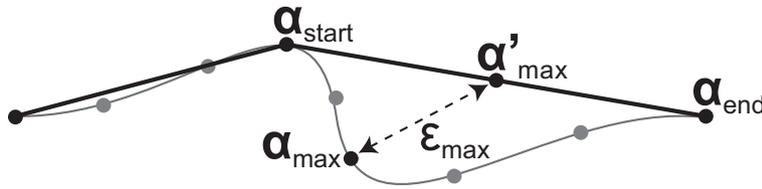


Figure 7.3: A compressed trajectory after one step with the PLS algorithm.

tive tuples for which Equation (7.1) holds and a match occurs with a manually labeled stop. The compression rate is computed on a separate data set of 400 vessel trajectories. We can see that for high compression rates our two-stage approach retains more stops than the single-stage approaches, which is what we are after in this application. This is due to the fact that the first stage of compression focusses purely on the velocity of the vessel, which determines whether a vessel is stopped or not. Note that the high compression rates for our two-stage approach, i.e., 98% instead of 93% in  $E_\mu$  for retaining 95% of the stops, allows us to triplicate the amount of input data for the same amount of output data, which is beneficial for large data sets. With Figure 7.4 we select appropriate settings for our two-stage algorithm, i.e.,  $\epsilon_p = 50m$  and  $\epsilon_v = 2.5$  knots.

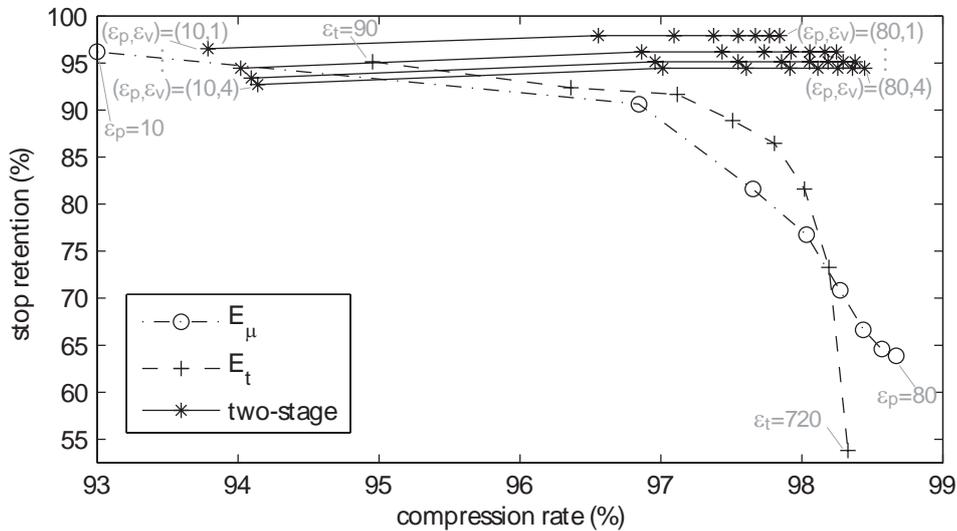


Figure 7.4: Results of stop retention experiment. For  $E_\mu$  we used  $\epsilon_p = 10, 20, 30, 40, 50, 60, 70, 80m$  and for  $E_t$  we used  $\epsilon_t = 90, 180, 270, 360, 450, 540, 630, 720s$ . For our two-stage variant we used the same  $\epsilon_p$  and conduct the experiment for velocity parameters  $\epsilon_v = 1, 2, 3, 4$  knots, resulting in a graph for each velocity with four knots the lowest graph.

The result of the two-stage PLS-algorithm is stored in a MySQL-database, where each record is a segment that describes a piece of constant movement. Let  $T$  be a trajectory, then  $T^C$  is its compressed variant resulting from our two-stage algorithm. For each consecutive pair  $\alpha_i, \alpha_j$  in  $T^C$  we create a record  $\langle uri, \mathbf{p}_i, \mathbf{p}_j, t_i, t_j, v_i, v_j, c_i, c_j \rangle$ . The  $uri$  is a Uniform Resource Identifier (URI), which uniquely identifies each segment in our knowledge base, and is based on the ship’s unique identifier, the Maritime Mobile Service Identity (MMSI) number, and the start time  $t_i$ . Furthermore, the segment contains a start position  $\mathbf{p}_i$  and end position  $\mathbf{p}_j$ , a start time  $t_i$  and end time  $t_j$ , a start speed  $v_i$  and end speed  $v_j$ , and a start course  $c_i$  and end course  $c_j$ . The segments constitute the first level of abstraction for meaningful movements, which are represented using the Simple Event Model (see Sec. 7.3.1) and can be agglomerated (and combined with other resources) into complex behaviour definitions.

The movement primitives *stop* and *move* are added by attaching a property with value `ex:stopped` to each segment with an average speed below the threshold  $\theta$  and the value `ex:move` to all other segments. By looking at the acceleration, computed by  $(v_j - v_i)/(t_j - t_i)$ , we also assign the values `ex:speeding_up` and `ex:slowing_down` to the `sem:eventType` (see Sec. 7.3.1) property of a segment. Additionally, the set of movement primitives can be extended with more semantic classes, like `ex:fast_move`, or other features, such as direction, i.e., `ex:to_North`. While direction can be obtained by thresholding, the `ex:fast_move` needs reasoning, since the notion of fast is related to ship type and weather conditions.

## 7.3 Semantic Web Technology

Only a limited number of movement patterns and useful vessel behaviour can be detected on the sole basis of the segments. Therefore, the Poseidon partner at VU University Amsterdam proposes to combine knowledge about the vessel's position, this position's characteristics (e.g., geographic type), movement characteristics, and the vessel's type: the same movement pattern may have different implications across object types. This allows for a richer behaviour definition and enables hypothesis testing about single vessel movement. For instance, if a vessel shuttles between two locations, it may be a dredger or a ferry. Without further knowledge, it is not possible to disambiguate between these two. But, web-based information can fill in some missing data, and help enrich the event description further. If we know that a vessel is a dredger, then we can determine that one of the locations is a dredging place. Equally, if one of the locations is a dredging place, then the vessel is a dredger. In order to reason about information coming from different sources, integration is needed, and therefore we have developed the Simple Event Model.

### 7.3.1 Simple Event Model

The Simple Event Model (SEM) [151] provides a set of classes and properties to define events and their context. The segments are represented as instances of SEM events. SEM's classes are organized in three groups (see Fig. 7.5). SEM Core classes describe the *classic* parts of an event: *What is happening?* (`sem:Event`, e.g., anchoring, fishing), *Who participates?* (`sem:Actor`, e.g., a vessel), *Where?* (`sem:Place`, e.g., in Rotterdam), and *When?* (`sem:Time`, e.g., during year 2008). SEM Types can be linked to each of the Core instances, which typically come from existing domain-specific vocabularies. For example, `sem:ActorTypes` can be selected from vessel classifications, such as vessel types defined in the AIS protocol [83] or on webpages like VesselTracker [154]. GeoNames can be used for the `sem:Places` and `sem:PlaceTypes`. SEM Constraints express Temporary, Authoritative, and Role constraints on the validity of properties between SEM Classes. Time and Place can have symbolic (URIs) or concrete values, to fit the representation of most data on the web. It is possible to use complex time representation such as the ISO 8601 standard and the TIMEX3 format for representing time interval, dates, and re-occurring time expressions (e.g., 'two times a week'). All the classes and properties of

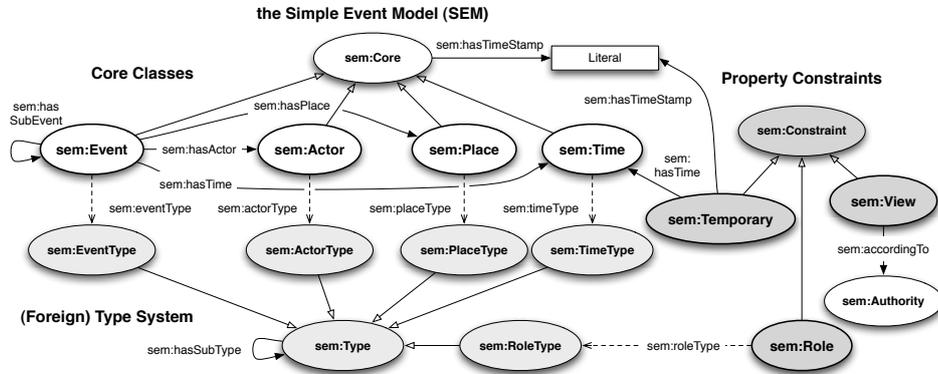


Figure 7.5: Simple Event Model. Ellipses represent classes, arrows represent properties between them and square boxes represent datatypes. Open-ended arrows represent subclass properties, dotted arrows are typing properties and the regular arrows' semantics are given by their labels.

SEM are optional and duplicable: SEM can integrate the partial information provided by different sources as different aspects of an event.

SEM is based on and informally linked to other event models such as the Event Ontology [120], LODE [132], and the F model [131]. The main differences with them are loosened constraints to be more flexible with the unpredictable syntax and inherent messiness of web data and a linking to domain ontologies via `sem:Types` instead of a direct usage of other model's classes, making us more independent from changes in vocabularies we can not control.

### 7.3.2 Moving Object Knowledge Base

The instances of SEM are either segment events (see Sec. 7.2.2), or events extracted and converted from the web. They are simply extracted if they are exposed as RDF (e.g., DBpedia [38] and GeoNames [59]), and converted to RDF if they are exposed as HTML (e.g., VesselTracker [154]). The former are accessed from the MySQL database via the D2RQ platform [24] and the latter two are accessed directly with the SWI-Prolog based triple store [140]. Rules defining ship types and ship behaviour enrich them. The rules are written in SWI-Prolog and the triple store is a knowledge base that enables us to link the different parts of an event represented in SEM with semantic web and spatial reasoning. In this way, rich movement patterns, vessel behaviour, and hypotheses can be defined and checked against trajectory data. Various web sources provide us with interesting information, but they can have different names for the same concepts. We align them to a single local identifier using the SKOS [102] vocabulary's `skos:(exact|close|broad|narrow|related)Match` properties. The current knowledge base consists of two areas in front of the Dutch coast: a circular area of 5,000km<sup>2</sup> around Rotterdam harbour and a circular area of 1,250km<sup>2</sup> around the Dutch island Texel. In total about 600 vessels are tracked for a single week, resulting in 92,000 segments with about 30 attributes, given by AIS or looked up from the web. We have annotated these

areas with 225 new GeoNames features, such as harbours to be able to reason with geographic locations. In Figure 7.8 we give an example of how different attributes of a single SEM event that were collected from the web can be used together, in a Prolog predicate that defines the notion of a *ship arriving at an appropriate harbour*. Figure 7.9 shows another example of a rule: *a ship banned from European waters close to a European harbour*. The vessel behaviour for both of these examples cannot be derived from the trajectory itself, and would not be obvious to a human operator, but they are nonetheless important events to be brought to his attention while monitoring vessels. In Section 7.5 we explain these use cases in more detail.

In this set up, SEM is the central piece for aggregating knowledge and resources at different levels of abstraction into event representations. These events can be used in complex, multi-step rules defining ship behaviour. The behaviour examples that have been described here are taken from the MSS domain, but they can be related to more generic movement patterns [41]. For example, the successive segments can be classified as a Spatio-Temporal sequence: ‘an ordered subsequence of locations with their timestamps’. Daily movements of migrating birds belong to the same generic pattern description. Complex events are multidimensional data, aggregating low-level tracks and semantic information. They can be visually explored by humans, to find computationally expensive patterns relevant to a certain domain or situation and to check detected anomalies. A visualization tool is needed to take advantage of the richness of the knowledge and present it in a useful way to a human user.

## 7.4 Trajectory Contingency Tables

The Poseidon partner at Eindhoven University of Technology proposes an interactive Trajectory Contingency Table (TCT), which allows expert users to browse the knowledge base. They want to discover spatial patterns for relations between attributes, such as the spatial changes over *time* for different *vessel types*. We do not know up front which attributes are available in the knowledge base, therefore the visualization needs to be generic. The TCT can deal with the most common types of attributes, such as time, geographical locations, numbers, and derived predicates, like *ship arriving at an appropriate harbour* from the previous section. With the TCT we can show temporal and spatial patterns, or correlation between attributes in terms of these patterns.

The visualization retrieves trajectories by querying the knowledge base with SPARQL queries [170], which convert segments to trajectories. First, we determine for tuples  $\alpha_i$  which attributes are available, by searching for owl:DatatypeProperty properties. The data type of the attribute is specified by the accompanying rdfs:range property and is encoded with predefined XML schema types, such as xsd:integer and xsd:string. Then, for each vessel, a trajectory  $\alpha_0, \dots, \alpha_{N-1}$  is reconstructed by gathering all values of the queried attributes of the vessel’s Actor and the Events related.

Pearson [113] introduced a contingency table where two categorical or binned attributes are put on the axes of a table by assigning (groups of) outcomes to a row (or column). In a cell, the number of data items are shown that contain the outcomes of the corresponding row and column for the attributes assigned to the axes. Further-

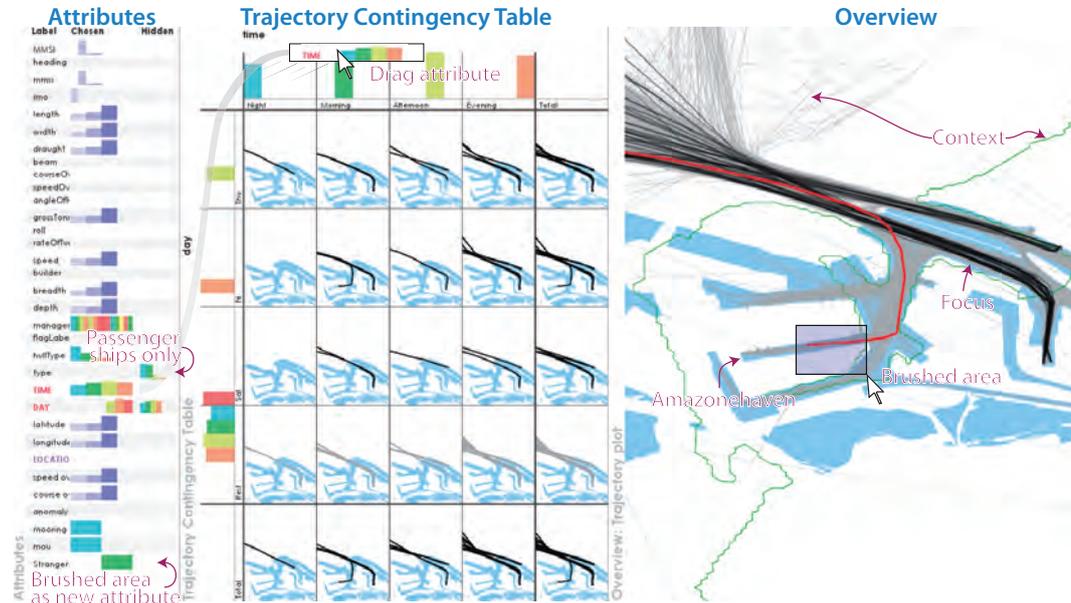


Figure 7.6: A Trajectory Contingency Table (TCT) with an Attribute view (left) and an Overview (right). All vessel trajectories in the knowledge base in the neighbourhood of Rotterdam harbour during a single week are shown. In the Attributes view, attributes are divided in bins and listed with two histograms: chosen bins and hidden bins. Trajectories are annotated with a vessel type attribute obtained from the web, and only the trajectories of passenger ships are shown in the TCT and highlighted in the overview by choosing this bin. The TCT is displayed with time and day attributes on the axes, with only Thursday until Saturday selected from the day attribute. Each cell contains a map with parts of trajectories that satisfy the accompanying row and column labels. In the Overview, chosen trajectories (focus) are highlighted in dark gray on top of the context containing all data as thin, light gray lines and a map with a contour of The Netherlands in green and a solid shape for the harbours in the port of Rotterdam. By brushing we select areas to define new attributes. From the visualization we notice a strong pattern towards two mooring areas, given by the dark trajectories in the overview. A single trajectory on Friday morning shows possible anomalous behaviour, by mooring in the 'Amazonehaven'. By means of reasoning with contextual information in Section 7.5.1, we can determine more precisely whether or not this is an anomalous behaving passenger ship.

more, each axis has an additional row that shows for each column the sum of the items. The data corresponding to a cell can be visualized with a small multiple. One of the first small multiple displays appeared in Los Angeles Times (July 22, 1979) [145] for air pollution. Becker *et al.* [21] generalized the contingency table to a Trellis display, which is an arrangement of small multiples of basic graphics. Brunson [27] showed a variant with maps as a scatterplot of locations and Carr *et al.* [32] applied this technique with choropleth maps. Among others, Stolte *et al.* [138] extended Trellis displays to Tableaux, to display various renderings in cells, including graphs, scatterplots, and maps, and is often used for relatively small data sets, while our TCT is capable of inter-

actively visualizing large data sets, i.e., the complete knowledge base.

In the TCT we display a map with the parts of trajectories that satisfy the constraints of the cell (see Fig. 7.6) given by the corresponding row and column. For instance, by putting *time* and *speed* on the axes, we see how the different speed-related areas, such as stop places, change over time. Since the input data is large and consists of many attributes, the TCT is extended with two views for improving the navigation: Attributes and Overview (see Fig. 7.6). In the Attributes view on the left, all attributes are listed and divided in bins, which are shown as a histogram. The bins are selectable and filter the whole data set. The exploration in specific domains may be optimized by manually adjusted bins for the attributes. By clicking on a bin, the whole data set is filtered on that bin. All other bins move to the histogram with hidden bins, to provide an overview of the current selection in a single column. The attributes can be dragged from the attributes view to the axes of the TCT to quickly investigate different settings. Alternatively from the trajectory plots in the cell, we can drag a time attribute to the cells to show a temporal distribution by means of clocks as shown on page 89. We can show maps again by dragging a location attribute to the cells. In the overview on the right of the TCT, the selection of data made by choosing bins (focus) is highlighted on top of all data (context). In the overview it is possible to brush parts of the focused trajectories to define areas, which can be used to define a new boolean attribute stating whether or not a trajectory is contained in the area. This new attribute can be used in the TCT, for further investigation.

## 7.5 Use Cases

### 7.5.1 Abnormal Mooring

AIS messages contain some identifying information about the broadcasting ship: name, call sign, MMSI, and a number assigned by the International Maritime Organization (IMO); they are kept when building the segments. We use the latter two to automatically query web sites, such as VesselTracker [154] and the Ship Index of the World Shipping Register [169], for additional ship information, which we convert with a script semi-automatically to RDF. One piece of such information is a precise (set of) type(s) for the ship, such as Oil Tanker, Ro-Ro (i.e., ferry), Refrigerated Fish Carrier. We are interested in passenger ships and how they behave during the week, which is shown in the TCT by means of a spatio-temporal distribution. We notice a strange movement on Friday morning. This is an inserted trajectory of a passenger ship simulated with Presto that anchors at the general cargo harbour ‘Amazonehaven’, whereas passenger ships should stop at passenger terminals. To verify whether this movement is anomalous, we reason further with contextual information from other sources.

The Port of Rotterdam Authority has published maps [117] with a classification of their different harbours (e.g., liquid bulk, food), which we have converted to Google Earth KML (see Fig. 7.7). We used the SWI-Prolog Space package to convert the KML placemarks with polygon shapes into RDF with GeorSS polygon literals. The SWI-Prolog Semantic web package can then be used to enrich events related to this data.

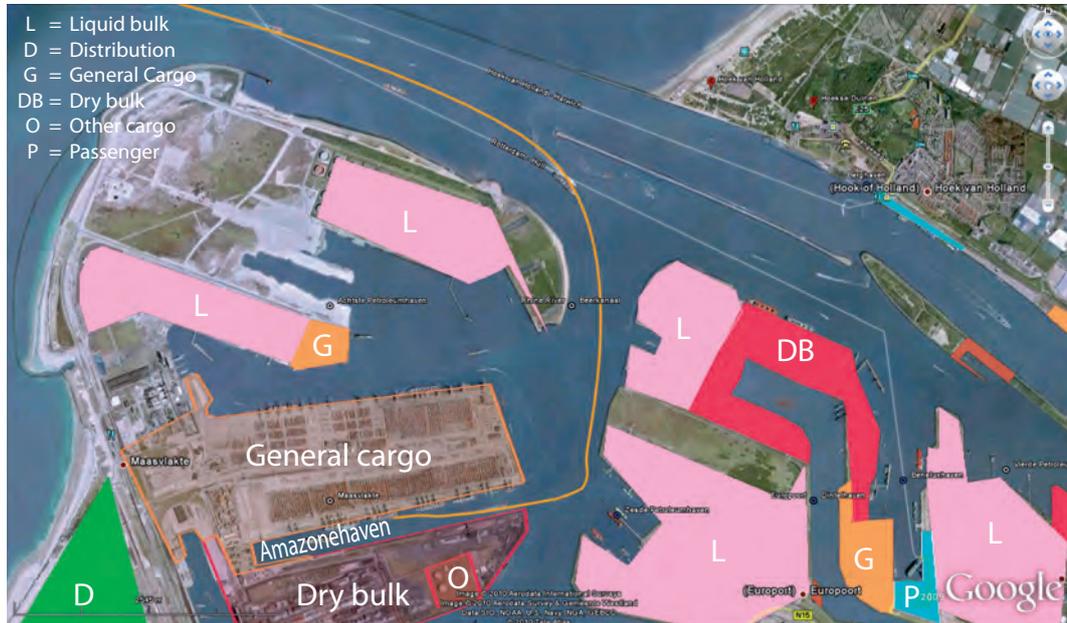


Figure 7.7: Map of Rotterdam harbour with a trajectory of a passenger ship created with Presto, GeoNames features as placemarks, and polygons with harbour types.

To define *abnormal* mooring, we created a Prolog rule that works in three steps. The first step describes the event using the SEM API [150]. The API provides a generic interface to create and query events, abstracting from the level of their RDF implementation details. This API, and our choice of implementation for the spatial indexing, represent the main difference between our platform and that of AllegroGraph [1]: the conversion between the most popular spatial and temporal formats and a unified representation is achieved automatically by our API, while the graph manipulation of SEM events is simplified for the users. Lines 1 to 9 of Figure 7.8 show the test predicate that queries for the types of the ships participating in an event as actors (line 7), and checks out whether they match with the type of the harbour (line 5), which is the place of the same event (line 3): stop (line 2). Step two checks out the matches between the types: they match if they are listed as pairs with the predicate `apt_ship_harbour_type` (line 12-15). This predicate defines the *normal* pairs: a Crude Oil Tanker should anchor at a liquid bulk anchorage, a Passenger Ship should anchor at a passenger terminal. Internally, the SEM API uses the SKOS alignments to find additional types for the ship and equivalences between them. The third step adds new triples to the knowledge base: `ex:abnormal_mooring` as a type for the instances of ship that satisfy the predicate `abnormal_mooring` (line 1).

This rule uses GeoNames for the URIs of geographic places, their types (harbour, populated place, etc.) and their geographic coordinates (line 5). GeoNames consists of user contributed content, and we extended it with extensive information for the

```

1 abnormal_mooring(Event) :-
2     event(Event, wordnet:'synset-stop-verb-1'),
3     event_place(Event, Place, _),
4     space_within_range(Place, Harbour, 0.2),
5     rdf(Harbour, geo:featureCode, geo:'H.HBR'),
6     rdfs_individual_of(Harbour, HarbourType),
7     event_actor(Event, Ship, ShipType),
8     % harbour type is not appropriate for the ship?
9     \+apt_ship_harbour_type(ShipType, HarbourType).
10
11 % liquid bulk harbours are appropriate for tanker, etc.
12 apt_ship_harbour_type(freebase:'en.tanker',
13     ex:ptype_LiquidBulkHarbour).
14 apt_ship_harbour_type(freebase:'en.passenger_ship',
15     ex:ptype_PassengerHarbour).
16 % ... further ship-type and harbour-type mappings
17
18 % add the property to the knowledge base
19 :- findall(Event, abnormal_mooring(Event),
20     AbnormalMoorings),
21     forall(member(AM, AbnormalMoorings),
22         rdf_assert(AM, ex:mooringEventType,
23             ex:etype_abnormal_mooring)).

```

Figure 7.8: Illustration of a Prolog test predicate for the abnormal mooring of a ship. This code makes use of the SEM Prolog API. The predicate is used to add an extra property to segments where an abnormal mooring takes place. This property can then be visualized.

Dutch coast and the port of Rotterdam, including unique identifiers for all the harbours in Rotterdam. The rule also uses ship trajectories (segments, they are the instances queried when instantiating the `sem:Event`, line 1), ship types from the web (Freebase, line 12-15), harbour types from the Port of Rotterdam website, spatial reasoning (`space_within_range` predicate, line 4), and RDF reasoning (`findall` predicate, line 19). The last predicate annotates segments with `ex:abnormal_mooring` type, which can be used for visualization.

The reasoning for the abnormal mooring of our simulated passenger ship is then as follows: The ship is of type `ex:atype_passenger_vessel`, which is mapped to the concept `freebase:en.passenger_ship` with the `skos:closeMatch` property, and the `sem:Place` of the `sem:Event` of type `wordnet:stop-verb-1` is near to the ‘Amazonehaven’, which has as a type `ex:ptype_PassengerHarbour`. The rule classified this stop as an abnormal mooring.

## 7.5.2 Banned Vessels

The previous rule is an example of an anomaly detection by means of a classification of abnormal behaviour for trajectories, given a domain-specific definition. Anomaly detection is crucial in an MSS system, as it enables a user to focus only on the subset of trajectories that are not behaving according to what is defined as normal patterns. Another example of such an abnormal pattern, combining ship movement trajectory,

```

1 banned_ship_observation(Event, Ship) :-
2     % ship observed
3     event_actor(Event, Ship, freebase:'en.ship'),
4     % ship has IMO number
5     rdf(Ship, ais:imo, literal(IMO_Number)),
6     % a banned ship
7     rdfs_individual_of(BannedShip, parismou:banned),
8     % same IMO number
9     rdf(BannedShip, ais:imo, literal(IMO_Number)),
10    event_place(Event, Place, _),
11    space_within_range(Place, Harbour, 0.2),
12    % observed near a harbour
13    rdf(Harbour, geo:featureCode, geo:'H.HBR'),
14    % has GeoNames ID for Europe as a parent feature
15    rdf_reachable(Harbour, geo:parentFeature,
16                 geoinst:'6255148/').
17
18 :- findall(Event, banned_ship_observation(Event, _),
19          BannedShipObservations),
20    forall(member(BSO, BannedShipObservations),
21           rdf_assert(BSO, ex:mouEventType,
22                     ex:etype_banned)).

```

Figure 7.9: Illustration of a Prolog test predicate for detecting ship banned by the Paris MOU near a European harbour. The predicate is used to add an extra property to segments of trajectories where a banned ship is observed in violation of this ban. This property can then be visualized.

ship type and web-based information is the detection of ships currently undergoing a ban from the European waters. The Paris Memorandum of Understanding on Port State Control (ParisMOU) [112] maintains a list of IMO identifiers of ships currently or historically under the EU ban. Fetching the data and converting it to RDF enables our system (see Fig. 7.9) to detect ship trajectories corresponding to banned ships that would come near a given harbour, in a similar fashion as Figure 7.8. Using type hierarchies defined in GeoNames, we can extend this rule to a harbour from any country for which the parent feature is Europe (Fig. 7.9, line 13), and add a property that is valid on the European Union level.

## 7.6 System Design

Our architecture is one of the possible configurations for composing a system that enables a user to detect patterns in moving objects at three different levels: data, knowledge, and graphical interface. The interfaces of the architecture given in Figure 7.1 contain the preprocessing stage of the trajectories (like NMEA), which is domain specific, but can be replaced by other systems. The system’s rationale is that if we convert data from different sources to a format compatible with an ontology in RDF, the data can be matched on the semantic level with other data and knowledge sources, which enhances a visual analytics system without further adaptation. Testing hypotheses based

on attributes not available in the knowledge base is possible by adding data on demand from the web or by reasoning.

Our methodology is illustrated with vessel traffic, but no major domain-specific assumptions have been made, and as a result we expect our approach to be applicable in other domains. However, the rules will be domain dependent, as our examples show. The various approaches and tools combined in the methodology each have certain limitations. Presto can only be used for generating trajectories on a geographical map; Trajectories for mice running in a cage for lab experiments or a cursor trail on a computer screen should be generated in other coordinate systems. In PLS, topological changes, such as self-intersections, may occur while generalizing a trajectory, hence the trajectory is required to be smooth. However, applying PLS is not required. In the knowledge base, when using external vocabularies, it may be possible to run into inconsistencies and trust issues. In the TCT, we may need to adopt the bins to get more sophisticated results in specific cases. Nevertheless, our architecture shows an interesting combination of geographic and semantic reasoning, coupled with a multi-dimensional visualization tool to address requirements for analyzing moving objects.

## 7.7 Conclusion and Future Work

We have presented an integrated approach for analyzing moving object data. The approach includes trajectory generation using *Presto*, trajectory compression using *Piecewise Linear Segmentation*, trajectory modeling with the *Simple Event Model*, which is the data model of our knowledge base. The trajectories’s multiple dimensions, including the additional knowledge created by *reasoning*, can be visualized in a *Trajectory Contingency Table*. This approach allows us to visually detect patterns in trajectory data by means of exploring the trajectory’s attributes and additional attributes obtained by various web sources and reasoning. New sources can be added easily which allows us to consult existing knowledge on demand for testing hypotheses containing attributes not available in the current knowledge base.

In future research we will include high-level abstractions of trajectories in SEM: behaviour descriptions defined across domains and movement definition based on other movement data sources than AIS data (e.g., ferry logs from harbour web sites). Furthermore, we would like to efficiently describe and reason with the relative position of moving ships and movements of multiple entities. For interoperability with existing GIS systems we may adopt our interfaces to OGC standards like Web Feature Service. Finally, we will investigate whether our approach can be used in other domains.

*We thank the reviewers, Jack van Wijk and Huub van de Wetering of Eindhoven University of Technology, Guus Schreiber of VU University Amsterdam, Pieter Adriaans and Maarten van Someren of University of Amsterdam, and Eric Postma of Tilburg University for their fruitful comments for improvements of the original articles [165, 164]. Furthermore, we thank Michael Borth of the Embedded Systems Institute and, Hans Hiemstra and Jacek Skowronek of Thales Nederland for their contributions in the early design of the system’s architecture.*

# 8

## Conclusions



*In Chapters 3 to 7 we have presented visualization techniques for improving situational awareness for vessel traffic. In this Chapter we summarize our contributions, discuss if our research satisfies the goals as set in Section 1.6, and suggest future work.*

## 8.1 Achievements

The main contribution of this thesis are a number of visualization techniques that provide insight in large amounts of trajectories with attributes, in particular taken from vessel traffic. In Chapters 3 to 6 density-based summarization techniques are presented. These techniques are evaluated with a number of case studies and, moreover a formal user study is conducted in Chapter 4. Furthermore, in Chapter 7 we presented a visual analytics approach for analyzing trajectories with attributes. In the remainder of this section we discuss these contributions in more detail.

In Chapter 3, we presented vessel density as a technique to visualize a summary of significant maritime movement features, including sea lanes, anchor areas, and slow movers around, for instance, oil platforms. All these features are related to velocity and popup with an adapted version of kernel density estimation where a kernel moves along the path of the trajectory with the measured velocity. For different kernel radii various movement features are visible. This inspired us to develop a multi-scale visualization technique to show all these types of features simultaneously. Vessel density enables operators to visually explain what movements have occurred related to speed, for instance, why that ships stop in anchor zones. Since the visualization is a static raster map, our technique can be easily integrated in existing geographical information systems that display live movements to provide an overview of average behaviour. For instance, vessel tracking systems enable an operator to validate the live vessel movements with historical data. Maritime movement features have often a matching variant in other domains. For example, an anchor zone could be a drinking or sleeping place in animal research. Therefore, vessel density is not only a useful tool for vessel traffic, but also for data from other domains.

The vessel density approach is evaluated in Chapter 4 with a formal user study. In this study, vessel density is compared to a dot animation and a space-time cube on performance for maritime tasks: counting stoppers and lanes, and detecting fast movers. The generated data matches with the tasks and is varied in size and for visual clutter to mimic properties of real-world data. The study shows that vessel density is significantly better for finding stoppers and is not significantly worse for the lanes and fast movers tasks with respect to the other visualizations. This result shows that vessel density is indeed a step forward for maritime tasks.

Apart the results of the evaluation, we have also showed an alternative method for the visualization community to analyze the effect of multiple independent test parameters. This is done by generating all the combinations of independent variables. For each setting homogeneous subsets are computed for the dependent variable, in our case the visualization type. All homogeneous subsets are graphically depicted that we are able to draw a conclusion based the effects of all independent variables.

In Chapter 5 vessel density is extended with more attributes and turns into a density map. A density field is created for a filtered subset, which shows areas with common attribute values. By creating multiple density fields in this way, we can see relations between these areas based on the common attribute values. Kernel settings and colors of each density field are customizable to make them distinguishable. Overlap of density fields is tackled with various blending modes and glyphs. The interpretability of density maps heavily depends on the nature of the attributes. If attribute values vary a lot for one trajectory, the areas become scattered, which makes it hard to find clear patterns. Furthermore, in areas with many different attribute values, the many density fields overlap, which makes it difficult to draw a conclusion.

Instead of attribute filters we allow analysts to use expressions for describing their patterns of interest in a composite density map as described in Chapter 6. Furthermore, a network of density field computations can be configured. The combination of these two enables an analyst to define advanced parameterized patterns within trajectories using their attributes. Once a composite density map is defined, an operator can use it in daily operations and explore them with an automatically generated user interface for the parameters. This two-layer approach is a step towards bridging the gap between the complexity of trajectory analysis and required simplicity for daily operational usage.

The density approaches as introduced in Chapters 3 to 6 turn out to be useful for the maritime domain as shown via maritime case studies. Note that these case studies cover a range of tasks for both the operator in daily surveillance as well as the analyst investigating long-term data. This indicates that density is a valuable approach with possibly a wide application range for the maritime domain. Furthermore, since many different maritime tasks are supported and these tasks often have matching tasks in other domains, the density approach seems to be useful for trajectories in other domains. In particular with composite density maps tasks in other domains can be tailored using the user-defined expressions to the specific needs of the task.

Chapter 7 shows an example of a visual analytics framework that was built around the concept of multivariate trajectories, which is the basis of this thesis. The semantic web infrastructure allows to enrich the trajectories by using multiple, semantically aligned data sources, such as semi-structured World Wide Web data or semantic web endpoints. Additional attributes can be computed using any kind of data, such as for instance contextual data. In this chapter we visualize the trajectories with an exploration tool, allowing us to test hypotheses by finding correlations between attributes. In our tool we take a visual approach for testing hypotheses, since the spatial distribution is important, and their related domain knowledge cannot be easily captured in correlation models. Since the semantic web allows to enrich multivariate trajectories, this technology is compatible to be used in combination with the density approaches as presented in Chapters 5 and 6 as well.

## 8.2 Research Contributions

In this section we elaborate to what extent we have met our research goals for the development of effective methods to visualize movement data. The research goals we set ourselves in Section 1.6 are as follows:

I. *Movement visualizations should **summarize the relation between the location and time.***

In all three density approaches we have addressed the relation between spatial distribution and time, each with a different interpretation: absolute time, relative time, and duration. With density maps in Chapter 5 we can show absolute time by defining filters for various consecutive moments in time, allowing us for instance to distinguish between movements in the morning and movements in the evening. Composite density maps in Chapter 6 address relative time with an iteration block using the time attribute, which computes relations between trajectories close by in both time and space. Finally, vessel density in Chapter 3, and all other presented density approaches, show the duration of the presence of objects at a certain location in a density field, which reveals movement features related to the velocity.

II. *New movement visualizations should be **evaluated for the performance in comparison with existing methods.***

In Chapter 4 we have conducted a user study to test the performance of our basic density approach as described in Chapter 3. To keep the size of a user study within manageable proportions we had to focus our study on maritime movement features and vessel density as a visualization technique.

III. *Movement visualizations should also **display distributions of other attributes than location and time given by the data.***

The (composite) density maps in Chapter 5 and 6 take a multivariate approach for showing distributions of trajectories. We demonstrate our approach with numerical attributes, allowing for simple filters and arithmetic computations, which leads to interesting insights in the vessel trajectories. In Chapter 7 we demonstrate a different visualization approach for showing attributes by means of small multiples as well. The distributions in this approach show up both in the axes from the attributes and the spatial distribution in the small multiple itself.

IV. *Movement visualizations should **display distributions of derived attributes.***

In Chapters 6 and 7 we allow to add derived attributes to the trajectories, which can be taken into account in the visualization. With composite density maps we can for instance add aggregated values by means of an enrichment block. In the knowledge base approach we can compute new attributes by means of reasoning, for instance by defining a classifier that tells if a ship moored at the appropriate harbour. Furthermore, users can add new attributes by means of predicates indicating whether or not a trajectory crossed a user-defined area.

V. *Movement visualizations should enable the user to **explore correlations between different attributes.***

By putting two partitioned attributes on the axes of a table we find correlations by looking at the distribution of the trajectories in the cells. This approach appears in two chapters. In Chapter 7 we were explicitly searching for patterns in a table view. A row or column appeared for each of the bins of the attributes. By showing a small rendering of the trajectory we could see spatial patterns. In case the trajectories were clustered in particular cells, correlations could appear. For instance, large vessels move far away from the coast with a moderate speed. The same principle also appeared in Chapter 5 for the distribution map widget. By drawing the semi-transparent trajectories using only the two chosen attributes in a 2D plane we can see a distribution that guides the analyst to choose filters of interest. This is, for instance, needed to define slow, medium, and fast vessels, which have varying threshold values for each vessel type.

The research goals help to answer the research question as stated in Section 1.5:

*How can analysts be supported to understand large amounts of trajectories with multiple attributes by using interactive, visual representations?*

We have made a considerable step forward to analyze multivariate trajectories for the following reasons. First of all, to deal with large amounts of trajectories the software not only needs to be fast, but also the graphical representation should be robust for large amounts of data. We have accomplished this by means of the density visualizations that aggregate arbitrary large amounts of data and are implemented using graphics hardware. Consequently, the results are interactive, allowing for exploration. Finally, the attributes of the trajectories are displayed with a number of visualization techniques to show various distributions.

In a broader view, considering situational awareness in vessel traffic, we have to distinguish between monitoring live traffic by an operator and the investigation of historical data by an analyst. Our research mainly focussed on the analyst’s work by visualizing large amounts of vessel traffic simultaneously. For live traffic, we have considered this a special case of historical data analysis, simply by considering a short period of time. For dedicated live traffic monitoring there is future work to be done. For instance, with composite density maps we have a powerful tool that needs further investigation on how it can be optimally configured to support live traffic monitoring. All in all, the patterns that result from our visualization techniques reveal insights that can be gained for large amounts of multivariate trajectories, resulting in improved situational awareness.

### 8.3 Valorization

Nowadays, research it is not only valued by the development and evaluation of new techniques. It is more and more of importance to show that new developments have made or can have an impact in society. Our research is not limited to the publication of scientific papers as enumerated in Appendix “Publications” on page 127.

Our research is conducted in cooperation with Dutch industry, in particular in cooperation with Thales Nederland located in Hengelo and Noldus Information Technology located in Wageningen. Furthermore, we have closely cooperated with other key research institutes, such as the Embedded Systems Institute in Eindhoven, the Dutch maritime research institute MARIN in Wageningen, Fraunhofer Institute IAIS in Sankt-Augustin, Germany, and the urban design group of TU Delft. These cooperations directed the research and inspired the use cases shown in our papers.

The cooperation with Thales as partner in the Poseidon project was pleasant and fruitful for a number of reasons. First of all, we are inspired by their ideas of history learning. Our view on this is shown in the anomaly detection use case of density maps [129]. Secondly, since the density method was developed, it has been a mascotte for our Poseidon project. Thirdly, Thales has always shown a great interest in evaluating our prototypes and investigate if they could use it for real-world analysis. And finally, Thales showed the density approach at a defense exhibition and there are even short-term plans to put the density approach in commercial products.

Also outside the scientific world our density approach has been attracting attention. After the vessel density publication [162] it has been shown at the website of New Scientist [109] and had follow ups at major visualization websites like Visual Complexity [98], Infoethetics [105], and many personal blogs. The density approach has also been published in the national press, including Kijk (issue 2, 2010), Intermediair (issue 40, 2009), De Ingenieur (issue 14, 2009), and De Technologiekrant (issue 19, 2009) with a total count of close to 400,000 copies.

Last but not least, the academic community has followed up our papers with interesting alternative approaches, which we discussed in Chapter 2. All in all, we think that our research is a good basis for product development based on the methods presented as well as for new techniques in future research.

## 8.4 Next Challenges

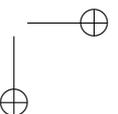
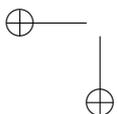
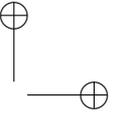
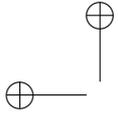
For each of the individual visualization approaches we have ended the chapter with suggestions for specific future work. In this chapter we step back and suggest more general future work related to the research presented in this thesis.

For all our visualization techniques there is a need to investigate further up to which point our techniques are useful for other domains. At this moment there are only a limited number of data sets with many attributes. We expect that it is a matter of a few years until data sets with multivariate trajectories become available, because GPS trackers have become increasingly popular due to the decreasing price and size. This trend has led to an increase of large data sets with trajectory paths, and we expect that the next generation trackers will contain more additional data. Current smart phones already contain GPS and compass devices, and, moreover, internet connectivity, which are the basic needs to capture multivariate trajectories.

One of the open visualization challenges is how to visualize movements, i.e., the relation between space and time. After more than half a decade of research there is still not a clear answer how we should draw the movement of one or more objects in

such a way that we instantly see what has happened. Of course the space-time cube is a possible solution, but the interpretation of two objects along curved paths is already hard to interpret in terms of their actual movement in the spatial plane. In short, the space-time cube has issues regarding scalability and occlusion as mentioned in Chapter 4. Also animation is not optimal for visualizing movement, because we have to wait till the end of the animation to draw a conclusion. Density is also not the answer, since data is aggregated over time. We are curious what the answer will be.

Another visualization challenge that needs further research is to show the difference between the actual movement and the reconstructed movement using the captured data points: *uncertainty*. Despite that we may not be able to show movements perfectly, as mentioned in the previous paragraph, we should show which errors occur in the reconstruction of the movement. Next to reconstruction errors, it is likely that at some point the tracking devices are broken, manipulated, or inaccurate. This can be by mistake, or on purpose to hide movements that are not supposed to be analyzed, such as a meeting between two ships where illegal goods could be exchanged. In this case there will be a large difference between the distance between the last two captured points and the largest detour one could take given the begin and end times. There are already a number of computational models for uncertainty of moving objects, like Brownian Bridges [75], but these models are not always satisfactory for multivariate data. The topic of uncertainty visualization for moving objects will be investigated further in the successor of the Poseidon project [48], called Metis [49].



## Bibliography

- [1] AASMAN, J. Geospatial and moving objects with RDF and allegrograph. In *Proceedings of the International Semantic Web Conference* (2009).
- [2] ALEXANDER, J. Loxodromes: A rhumb way to go. *Mathematics Magazine* 77, 5 (2004), 349–356.
- [3] AMERICAN ASSOCIATION OF PORT AUTHORITIES. World port ranking, 2009. Accessed: May 2011, Available from: <http://aapa.files.cms-plus.com/PDFs/WORLD%20PORT%20RANKINGS%202009.pdf>.
- [4] ANDERSON, T. Comparison of spatial methods for measuring road accident ‘hotspots’: A case study of London. *Journal of Maps* (2007), 55–69.
- [5] ANDERSSON, M., GUDMUNDSSON, J., LAUBE, P., AND WOLLE, T. Reporting leaders and followers among trajectories of moving point objects. *GeoInformatica* 12 (2008), 497–528.
- [6] ANDRIENKO, G., A. N., AND SAVINOV, A. Choropleth maps: Classification revisited. In *Proceedings of International Cartographic Conference* (2001), pp. 1209–1219.
- [7] ANDRIENKO, G., AND ANDRIENKO, N. Spatio-temporal aggregation for visual analysis of movements. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology* (2008), pp. 51–58.
- [8] ANDRIENKO, G., AND ANDRIENKO, N. A visual analytics approach to exploration of large amounts of movement data. In *Proceedings of the International Conference on Visual Information Systems: Web-Based Visual Information Search and Management* (2008), pp. 1–4.
- [9] ANDRIENKO, G., ANDRIENKO, N., BREMM, S., SCHRECK, T., VON LANDESBERGER, T., BAK, P., AND KEIM, D. Space-in-time and time-in-space self-organizing maps for exploring spatiotemporal patterns. *Computer Graphics Forum* 29, 3 (2010), 913–922.
- [10] ANDRIENKO, G., ANDRIENKO, N., DEMŠAR, U., DRANSCH, D., DYKES, J., FABRIKANT, S. I., JERN, M., KRAAK, M.-J., SCHUMANN, H., AND TOMINSKI, C. Space, time and visual analytics. *International Journal of Geographical Information Science* 24, 10 (2010), 1577–1600.

- [11] ANDRIENKO, G., ANDRIENKO, N., DYKES, J., FABRIKANT, S. I., AND WACHOWICZ, M. Geovisualization of dynamics, movement and change: Key issues and developing approaches in visualization research. *Information Visualization* 7, 3–4 (2008), 173–180.
- [12] ANDRIENKO, G., ANDRIENKO, N., MLADENOV, M., MOCK, M., AND PÖLITZ, C. Discovering bits of place histories from people’s activity traces. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology* (2010), pp. 59–66.
- [13] ANDRIENKO, N., AND ANDRIENKO, G. *Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach*. Springer-Verlag, 2006.
- [14] ANDRIENKO, N., AND ANDRIENKO, G. Designing visual analytics methods for massive collections of movement data. *Cartographica* 42, 2 (2007), 117–138.
- [15] ANDRIENKO, N., AND ANDRIENKO, G. A general framework for using aggregation in visual exploration of movement data. *The Cartographic Journal* 47, 1 (2010), 22–40.
- [16] ANDRIENKO, N., AND ANDRIENKO, G. Spatial generalization and aggregation of massive movement data. *IEEE Transactions on Visualization and Computer Graphics* 17 (2011), 205–219.
- [17] ANDRIENKO, N., ANDRIENKO, G., PELEKIS, N., AND SPACCAPIETRA, S. *Basic Concepts of Movement Data*. Springer-Verlag Berlin Heidelberg, 2008, ch. 1, pp. 15–38.
- [18] ASTRA PAGING LTD. VT Explorer, 2011. Accessed: July 2011, Available from: <http://www.vtexplorer.com>.
- [19] AZUMA, R., NEELY, H., DAILY, M., AND GEISS, R. Visualization tools for free flight air-traffic management. *IEEE Computer Graphics and Applications* 20 (2000), 32–36.
- [20] BAK, P., MANSMANN, F., JANETZKO, H., AND KEIM, D. Spatiotemporal analysis of sensor logs using growth ring maps. *IEEE Transactions on Visualization and Computer Graphics* 15, 6 (2009), 913–920.
- [21] BECKER, R. A., CLEVELAND, W. S., AND SHYU, M. J. The visual design and control of trellis display. *Journal of Computational and Statistical Graphics* 5 (1996), 123–155.
- [22] BERTIN, J. *Semiology of Graphics: Diagrams, Networks, Maps*. ESRI Press, 2011. 2010 edition with English translation by W. J. Berg.
- [23] BERTOLOTTA, M., MARTINO, S., FERRUCCI, F., AND KECHADI, T. Towards a framework for mining and analysing spatio-temporal datasets. *International Journal of Geographical Information Science* 21, 8 (2007), 895–906.

## Bibliography

115

- [24] BIZER, C. The D2RQ platform - Treating non-RDF databases as virtual RDF graphs, 2009. Accessed: February 2010, Available from: <http://www4.wiwiss.fu-berlin.de/bizer/d2rq>.
- [25] BOYANDIN, I., BERTINI, E., BAK, P., AND LALANNE, D. Flowstrates: An approach for visual exploration of temporal origin-destination data. *Computer Graphics Forum* 30, 3 (2011), 971–980.
- [26] BRUNSDON, C. Estimating probability surfaces for geographical point data: An adaptive kernel algorithm. *Computers and Geosciences* 21, 7 (1995), 877–894.
- [27] BRUNSDON, C. The comap: Exploring spatial pattern via conditional distributions. *Computers, Environment and Urban Systems* 25, 1 (2001), 53–68.
- [28] BUCCELLA, A., CECHICH, A., AND FILLOTTRANI, P. Ontology-driven geographic information integration: A survey of current approaches. *Computers & Geosciences* 35, 4 (2009).
- [29] BURROUGH, P. A., AND McDONNELL, R. A. *Spatial Analysis using Continuous Fields*. Oxford University Press, 1998, ch. 8, pp. 200–236.
- [30] CÂMARA, G., PALOMO, D., CARTAXO, R., SOUZA, M., AND OLIVEIRA, R. F. D. Towards a generalized map algebra: principles and data types. In *Proceedings of the Workshop Brasileiro de Geoinformática* (2005).
- [31] CAO, H., WOLFSON, O., AND TRAJCEVSKI, G. Spatio-temporal data reduction with deterministic error bounds. *The International Journal on Very Large Data Bases* 15, 3 (2006), 211–228.
- [32] CARR, D. B., CHEN, J., BELL, B. S., PICKLE, L., AND ZHANG, Y. Interactive linked micromap plots and dynamically conditioned choropleth maps. In *Proceedings of the Annual National Conference on Digital Government Research* (2002), pp. 1–7.
- [33] CASPER. Live aircraft tracking, 2010. Accessed: August 2011, Available from: <http://www.casperflights.com>.
- [34] CHEN, B., HUANG, F., FANG, Y., HUANG, Z., AND LIN, H. An approach for heterogeneous and loosely coupled geospatial data distributed computing. *Computers & Geosciences* 36, 7 (2010).
- [35] CHI, E., RIEDL, J., BARRY, P., AND KONSTAN, J. Principles for information visualization spreadsheets. *IEEE Computer Graphics and Applications* 18, 4 (1998), 30–38.
- [36] CORBETT, J. J., AND WINEBRAKE, J. The impacts of globalization on international maritime transport activity: Past trends and future perspectives. In *Proceedings of Global Forum on Transport and Environment in a Globalising World* (2008).

- [37] CRNOVRSANIN, T., MUELDER, C., CORREA, C., AND MA, K.-L. Proximity-based visualization of movement trace data. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology* (2009), pp. 11–18.
- [38] DBPEDIA. The DBpedia knowledge base, 2010. Accessed: April 2010, Available from: <http://www.DBpedia.org>.
- [39] DEMISSIE, B. *Geo-Visualization of Movements: Moving Objects in Static Maps, Animation and The Space-Time Cube*. VDM, 2010.
- [40] DEMŠAR, U., AND VIRRANTAUŠ, K. Space-time density of trajectories: Exploring spatio-temporal patterns in movement data. *International Journal of Geographical Information Science* 24, 10 (2010), 1527–1542.
- [41] DODGE, S., WEIBEL, R., AND LAUTENSCHÜTZ, A.-K. Towards a taxonomy of movement patterns. *Information Visualization* 7, 3–4 (2008), 240–252.
- [42] DOUGLAS, D. H., AND PEUCKER, T. K. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica* 10, 2 (1973), 112–122.
- [43] DOWNS, J. A. Time-geographic density estimation for moving point objects. In *Proceedings of the International Conference on Geographic Information Science* (2010), pp. 16–26.
- [44] DYKES, J., MAC EACHREN, A. M., AND KRAAK, M.-J., Eds. *Exploring Geovisualization*. Elsevier, 2005.
- [45] DYKES, J. A., AND MOUNTAIN, D. M. Seeking structure in records of spatio-temporal behaviour. *Computational Statistics & Data Analysis* 43, 4 (2003), 581–603.
- [46] ECCLES, R., KAPLER, T., HARPER, R., AND WRIGHT, W. Stories in geotime. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology* (2007), pp. 19–26.
- [47] ELMQVIST, N., DRAGICEVIC, P., AND FEKETE, J.-D. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Transactions on Visualization and Computer Graphics* 14, 6 (2008), 1539–1148.
- [48] EMBEDDED SYSTEMS INSTITUTE. The Poseidon project, 2007. Accessed: February 2010, Available from: <http://www.esi.nl/poseidon>.
- [49] EMBEDDED SYSTEMS INSTITUTE. The Metis project, 2011. Accessed: August 2011, Available from: <http://www.esi.nl/metis>.
- [50] ENDSLEY, M. *Situational Awareness in Aviation Systems*. Lawrence Erlbaum Associates, Inc., 1999, ch. 11, pp. 257–276.

## Bibliography

117

- [51] ENDSLEY, M., AND CONNORS, E. Situation awareness: State of the art. In *IEEE Power and Energy Society General Meeting (2008)*, pp. 1–4.
- [52] FARRELL, J. A., AND BARTH, M. *The Global Positioning System and Inertial Navigation*. McGraw-Hill, 1998.
- [53] FAVILL, J. *Primer of Celestial Navigation*. Cornell Maritime Press, 1944.
- [54] FEIBUSH, E., GAGVANI, N., AND WILLIAMS, D. Visualization for situational awareness. *IEEE Computer Graphics and Applications* 20, 5 (2000), 38–45.
- [55] FISHER, D. Hotmap: Looking at geographic attention. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1184–1191.
- [56] FLIGHT RADAR 24. Live flight tracker, 2010. Accessed: August 2011, Available from: <http://www.flightradar24.com>.
- [57] FORER, P. *Spatial and Temporal Reasoning in Geographic Information Systems*. Oxford University Press, New York, 1998, ch. Geometric Approaches to the Nexus of Time, Space and Microprocess: Implementing a Practical Model for Mundane Socio-spatial Systems, pp. 171–190.
- [58] FRIENDLY, M. Visions and re-visions of Charles Joseph Minard. *Journal of Educational and Behavioral Statistics* 27, 1 (2002), 31–51.
- [59] GEONAMES. The GeoNames geographical database covers all countries and contains over eight million placenames that are available for download free of charge, 2010. Accessed: April 2010, Available from: <http://www.geonames.org>.
- [60] GONZALEZ, C. Does animation in user interfaces improve decision making? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Common Ground (1996)*, pp. 27–34.
- [61] GOOGLE. Earth, 2011. Accessed: August 2011, Available from: <http://www.google.com/earth>.
- [62] GOTZ, D., ZHOU, M. X., AND AGGARWAL, V. Interactive visual synthesis of analytic knowledge. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology (2006)*, pp. 51–58.
- [63] GRUNDY, E., JONES, M. W., LARAMEE, R. S., WILSON, R. P., AND SHEPARD, E. L. Visualisation of sensor data from animal movement. *Computer Graphics Forum* 28, 3 (2009), 815–822.
- [64] GUDMUNDSSON, J., KATAJAINEN, J., MERRICK, D., ONG, C., AND WOLLE, T. Compressing spatio-temporal trajectories. *Computational Geometry* 42, 9 (2009), 825–841.
- [65] GUO, H., WANG, Z., YU, B., ZHAO, H., AND YUAN, X. Tripvista: Triple perspective visual trajectory analytics and its application on microscopic traffic data at a road intersection. In *Proceedings of IEEE Pacific Visualization Symposium (2011)*, pp. 163–170.

- [66] GUO, Z., WARD, M. O., AND RUNDENSTEINER, E. A. Model space visualization for multivariate linear trend discovery. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology* (2009), pp. 75–82.
- [67] HÄGERSTRAND, T. *Innovationsförloppet ur Korologisk Kynpunkt*. Lund, Gleerup, 1952. Translation in English: Allan Pred, Innovation Diffusion as Spatial Process.
- [68] HÄGERSTRAND, T. What about people in regional science? *Papers in Regional Science* 24 (1970), 6–21.
- [69] HAGH-SHENAS, H., KIM, S., INTERRANTE, V., AND HEALEY, C. Weaving versus blending. *IEEE Transactions on Visualization and Computer Graphics* 13 (2007), 1270–1277.
- [70] HALBERDA, J., AND FEIGENSON, L. Counting without individuals: Rapid parallel enumeration of sets implicates preattentive object files. *Journal of Vision* 5, 8 (2005).
- [71] HARATI-MOKHTARI, A., WALL, A., BROOKS, P., AND WANG, J. AIS: Data reliability and human error implications. *The Journal of Navigation* 60, 3 (2007), 373–389.
- [72] HEINRICH, J., AND WEISKOPE, D. Continuous parallel coordinates. *IEEE Transactions on Visualization and Computer Graphics* 15 (2009), 1531–1538.
- [73] HIEBEL, G., HANKE, K., AND HAYEK, I. Methodology for CIDOC CRM based data integration with spatial data. In *Proceedings of the Conference on Computer Applications and Quantitative Methods in Archaeology* (2010).
- [74] HOCHHEISER, H., AND SHNEIDERMAN, B. Dynamic query tools for time series data sets: Timebox widgets for interactive exploration. *Information Visualization* 3, 1 (2004), 1–18.
- [75] HORNE, J. S., GARTON, E. O., KRONE, S. M., AND LEWIS, J. S. Analyzing animal movements using brownian bridges. *Ecology* 88, 9 (2007), 2354–2363.
- [76] HORNSBY, K. S., AND JOSHI, K. Combining ontologies to automatically generate temporal perspectives of geospatial domains. *GeoInformatica* 14, 4 (2010), 481–505.
- [77] HURTER, C., TISSOIRES, B., AND CONVERSY, S. FromDaDy: Spreading aircraft trajectories across views to support iterative queries. *IEEE Transactions on Visualization and Computer Graphics* 15, 6 (2009), 1017–1024.
- [78] HURTER, C., TISSOIRES, B., AND CONVERSY, S. Accumulation as a tool for efficient visualization of geographical and temporal data. In *Proceedings of the AGILE Workshop Geospatial Visual Analytics: Focus on Time* (2010).
- [79] ICT COST ACTION IC0903. Knowledge Discovery from Moving Objects (MOVE), 2010. Accessed: July 2011, Available from: <http://www.move-cost.info>.

## Bibliography

119

- [80] IEEE. Visweek, 2011. Accessed: July 2011, Available from: <http://www.visweek.org>.
- [81] INSELBERG, A. *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications*. Springer, 2009.
- [82] INTERNATIONAL MARITIME ORGANIZATION. Long-range identification and tracking system: Technical documentation. *MSC Circular MSC.1/Circ.1259/Rev.1* (2009).
- [83] INTERNATIONAL TELECOMMUNICATION UNION (ITU). Technical characteristics for an automatic identification system using time division multiple access in the VHF maritime mobile band. *Recommendation ITU-R M.1371-1* (2001).
- [84] JAKOTA CRUISE SYSTEMS GMBH. FleetMon, 2011. Accessed: July 2011, Available from: <http://www.digital-seas.com>.
- [85] JANOOS, F, SINGH, S., IRFANOGLU, O., MACHIRAJU, R., AND PARENT, R. Activity analysis using spatio-temporal trajectory volumes in surveillance applications. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology* (2007), pp. 3–10.
- [86] JANSSENS, J., HIEMSTRA, H., AND POSTMA, E. Creating artificial vessel trajectories with Presto. In *Proceedings of the Benelux Conference on Artificial Intelligence* (2010).
- [87] JIN, X., AND TAI, C.-L. Analytical methods for polynomial weighted convolution surfaces with various kernels. *Computers & Graphics* 26, 3 (2002), 437–447.
- [88] KENNEDY, M., AND KOPP, S. *Understanding Map Projections*. ESRI Press, 2001.
- [89] KJELLIN, A., PETTERSSON, L. W., SEIPEL, S., AND LIND, M. Different levels of 3D: An evaluation of visualized discrete spatiotemporal data in space-time cubes. *Information Visualization* 9, 2 (2010), 152–164.
- [90] KOLDENHOEF, Y., VAN DER TAK, C., AND GLANSDORP, C. Risk awareness: A model to calculate the risk of a ship dynamically. In *Proceedings of the International Scientific and Technical Conference on Maritime Traffic Engineering* (2009).
- [91] KOSARA, R., MIKSCH, S., AND HAUSER, H. Focus+context taken literally. *IEEE Computer Graphics and Applications* 22 (2002), 22–29.
- [92] KRAAK, M.-J. The space-time cube revisited from a geovisualization perspective. In *Proceedings of the International Cartographic Conference* (2003), pp. 1988–1995.
- [93] KRISTENSSON, P., DAHLBACK, N., ANUNDI, D., BJORNSTAD, M., GILLBERG, H., HARALDSSON, J., MARTENSSON, I., NORDVALL, M., AND STAHL, J. An evaluation of space time cube representation of spatiotemporal patterns. *IEEE Transactions on Visualization and Computer Graphics* 15, 4 (2009), 696–702.
- [94] KUSTWACHT NEDERLAND. Jaarverslag kustwacht 2009, 2009.

- [95] LAMPE, O. D., AND HAUSER, H. Interactive visualization of streaming data with kernel density estimation. In *Proceedings of IEEE Pacific Visualization Symposium* (2011), pp. 171–178.
- [96] LI, X., ÇÖLTEKIN, A., AND KRAAK, M.-J. Visual exploration of eye movement data using the space-time-cube. In *Proceedings of the International Conference on Geographic Information Science* (2010), pp. 295–309.
- [97] LIAO, Z., YU, Y., AND CHEN, B. Anomaly detection in GPS data based on visual analytics. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology* (2010), pp. 51–58.
- [98] LIMA, M. Visual complexity, 2009. Accessed: July 2011, Available from: [http://www.visualcomplexity.com/vc/project\\_details.cfm?id=705&index=705&domain=](http://www.visualcomplexity.com/vc/project_details.cfm?id=705&index=705&domain=).
- [99] MACÉACHREN, A. M. *How Maps Work: Representation, Visualization, and Design*. The Guilford Press, 1995.
- [100] MACIEJEWSKI, R., RUDOLPH, S., HAFEN, R., ABUSALAH, A. M., YAKOUT, M., OUZZANI, M., CLEVELAND, W. S., GRANNIS, S. J., AND EBERT, D. S. A visual analytics approach to understanding spatiotemporal hotspots. *IEEE Transactions on Visualization and Computer Graphics* 99, RP (2009), 205–220.
- [101] MENNIS, J., VIGER, R., AND TOMLIN, C. D. Cubic map algebra functions for spatio-temporal analysis. *Cartography and Geographic Information Science* 32 (2005), 17–32.
- [102] MILES, A., AND BECHHOFFER, S. SKOS simple knowledge organization system reference, 2009. Accessed: April 2010, Available from: <http://www.w3.org/TR/skos-reference>.
- [103] MILLER, J. R. Attribute blocks: Visualizing multiple continuously defined attributes. *IEEE Computer Graphics and Applications* 27 (2007), 57–69.
- [104] MODAP. Mobility, Data Mining and Privacy, 2009. Accessed: July 2011, Available from: <http://www.modap.org>.
- [105] MOERE, A. V. Infosthetics, 2010. Accessed: July 2011, Available from: [http://www.infosthetics.com/archives/2010/07/visualization\\_of\\_vessel\\_trajectories.html](http://www.infosthetics.com/archives/2010/07/visualization_of_vessel_trajectories.html).
- [106] MONMONIER, M. *How to Lie with Maps*. The University of Chicago Press, 1991.
- [107] NATIONAL AERONAUTICS AND SPACE ADMINISTRATION (NASA). World wind, 2008. Accessed: February 2010, Available from: <http://worldwind.arc.nasa.gov>.
- [108] NETHERLANDS HYDROGRAPHIC OFFICE AND MINISTRY OF TRANSPORT, PUBLIC WORKS AND WATER MANAGEMENT. Vessel traffic on the north sea, 2004. Accessed: March 2009, Available from: [http://www.noordzeeloket.nl/Images/VesselTraffic\\_tcm14-2878.pdf](http://www.noordzeeloket.nl/Images/VesselTraffic_tcm14-2878.pdf).

## Bibliography

121

- [109] NEW SCIENTIST. Visions of data, 2009. Accessed: July 2011, Available from: <http://www.newscientist.com/gallery/dn17746-visions-of-data/1>.
- [110] OPENSTREETMAP. Accessed: March 2009, Available from: <http://www.openstreetmap.org>.
- [111] ORELLANA, D., AND RENSO, C. Developing an interactions ontology for characterising pedestrian movement behavior. In *Movement-aware Applications for Sustainable Mobility: Technologies and Approaches*. 2010, pp. 62–86.
- [112] PARISMOU. Paris memorandum of understanding on port state control - banned ships, 2010. Accessed: April 2010, Available from: <http://www.parismou.org>.
- [113] PEARSON, K. On the theory of contingency and its relation to association and normal correlation. *Drapers' Company Research Memoirs. Biometric Series I*. (1904).
- [114] PETERS, S., AND KRISP, J. M. Density calculation for moving points. In *Proceedings of AGILE International Conference on Geographic Information Science* (2010).
- [115] PEUQUET, D. J. *Representations of Space and Time*. The Guilford Press, 2002.
- [116] PHONG, B. T. Illumination for computer generated pictures. *Communications of the ACM* 18, 6 (1975), 311–317.
- [117] PORT OF ROTTERDAM AUTHORITY. Port maps - Branches, 2010. Accessed: April 2010, Available from: [http://www.portofrotterdam.com/en/about\\_port/port\\_maps/branches/index.jsp](http://www.portofrotterdam.com/en/about_port/port_maps/branches/index.jsp).
- [118] PORT OF ROTTERDAM AUTHORITY. Port statistics, 2010.
- [119] RADIO TECHNICAL COMMISSION FOR AERONAUTICS INC. Minimum aviation system performance standards for automatic dependent surveillance broadcast (ads-b). *DO-242A* (2002).
- [120] RAIMOND, Y., AND ABDALLAH, S. The event ontology, 2007. Accessed: April 2010, Available from: <http://www.purl.org/NET/c4dm/event.owl>.
- [121] RIVEIRO, M., AND FALKMAN, G. Supporting the analytical reasoning process in maritime anomaly detection: Evaluation and experimental design. In *Proceedings of the International Conference Information Visualization* (2010), pp. 170–178.
- [122] RIVEIRO, M., FALKMAN, G., AND ZIEMKE, T. Visual analytics for the detection of anomalous maritime behavior. In *Proceedings of the International Conference Information Visualization* (2008), pp. 273–279.
- [123] ROBERTSON, G., FERNANDEZ, R., FISHER, D., LEE, B., AND STASKO, J. Effectiveness of animation in trend visualization. *IEEE Transactions on Visualization and Computer Graphics* 14, 6 (2008), 1325–1332.

- [124] ROBINSON, A. H., MORRISON, J. L., MUEHRCKE, P. C., KIMERLING, A. J., AND GUPTILL, S. C. *Elements of Cartography*, sixth ed. John Wiley & Sons, Inc., 1995.
- [125] ROMERO, M., SUMMET, J., STASKO, J., AND ABOWD, G. Viz-A-Vis: Toward visualizing video through computer vision. *IEEE Transactions on Visualization and Computer Graphics 14* (2008), 1261–1268.
- [126] SALAS, R. P., EDELSON, G. D., KLEPPNER, P. S., AND SHAVER, R. S. Data processing apparatus and method for a reformattable multidimensional spreadsheet. *U.S. Patent* (1994).
- [127] SÁNCHEZ, A. M., PATRICIO, M. A., GARCÍA, J., AND MOLINA, J. A context model and reasoning system to improve object tracking in complex scenarios. *Expert Systems with Applications 36*, 8 (2009), 10995–11005.
- [128] SCHEEPENS, R., WILLEMS, N., VAN DE WETERING, H., ANDRIENKO, G., ANDRIENKO, N., AND VAN WIJK, J. J. Composite density maps for multivariate trajectories. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (2011), 2518–2527.
- [129] SCHEEPENS, R., WILLEMS, N., VAN DE WETERING, H., AND VAN WIJK, J. J. Interactive visualization of multivariate trajectory data with density maps. In *Proceedings of IEEE Pacific Visualization Symposium* (2011), pp. 147–154.
- [130] SCHEEPENS, R., WILLEMS, N., VAN DE WETERING, H., AND VAN WIJK, J. J. Interactive density maps for moving objects. *IEEE Computer Graphics and Applications 32*, 1 (2012).
- [131] SCHERB, A., FRANZ, T., SAATHOFF, C., AND STAAB, S. F—A model of events based on the foundational ontology DOLCE+DnS ultralight. In *Proceedings of the Conference on Knowledge Capturing* (2009).
- [132] SHAW, R., TRONCY, R., AND HARDMAN, L. LODÉ: Linking open descriptions of events. In *Proceedings of the Annual Asian Semantic Web Conference* (2009), pp. 153–167.
- [133] SILVERMAN, B. W. *Density Estimation for Statistics and Data Analysis*. No. 26 in Monographs on Statistics and Applied Probability. Chapman & Hall, 1992.
- [134] SLINGSBY, A., DYKES, J., AND WOOD, J. Using treemaps for variable selection in spatio-temporal visualization. *Information Visualization 7*, 3–4 (2008), 210–224.
- [135] SNYDER, J. P. *Flattening the Earth: Two Thousand Years of Map Projections*. University of Chicago Press, 1993.
- [136] SOVON VOGELONDERZOEK NEDERLAND. Satellietzenders bij meeuwen en ganzen, 2011. Accessed: July 2011, Available from: <http://www.sovon.nl/default.asp?id=408>.
- [137] SPACCAPIETRA, S., PARENT, C., DAMIANI, M. L., DE MACÊDO, J. A. F., PORTO, F., AND VANGENOT, C. A conceptual view on trajectories. *Data & Knowledge Engineering 65*, 1 (2008), 126–146.

## Bibliography

123

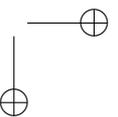
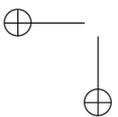
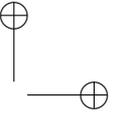
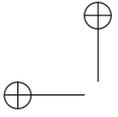
- [138] STOLTE, C., TANG, D., AND HANRAHAN, P. Polaris: A system for query, analysis, and visualization of multidimensional relational databases. *IEEE Transactions on Visualization and Computer Graphics* 8, 1 (2002), 52–65.
- [139] SUNG, K., PEARCE, A., AND WANG, C. Spatial-temporal antialiasing. *IEEE Transactions on Visualization and Computer Graphics* 8, 2 (2002), 144–153.
- [140] SWI-PROLOG. SWI-Prolog for the (semantic) web, 2010. Accessed: February 2010, Available from: <http://www.swi-prolog.org/web>.
- [141] THE MARINE TRAFFIC PROJECT. Live ship map - AIS - vessel traffic and positions, 2011. Accessed: July 2011, Available from: <http://www.marinetraffic.com/ais>.
- [142] THOMAS, J., AND KIELMAN, J. Challenges for visual analytics. *Information Visualization* 8, 4 (2009), 309–314.
- [143] THOMAS, J. J., AND COOK, K. A., Eds. *Illuminating the path*. IEEE Computer Society, 2005.
- [144] TOMLIN, C. D., AND BERRY, J. K. A mathematical structure for cartographic modeling in environmental analysis. In *Proceedings of the Symposium of the American Congress on Surveying and Mapping* (1979), pp. 269–283.
- [145] TUFTE, E. R. *The visual display of quantitative information*. Graphics Press, 1986.
- [146] TVERSKY, B., MORRISON, J. B., AND BETRANCOURT, M. Animation: Can it facilitate? *International Journal of Human-Computer Studies* 57 (2002), 247–262.
- [147] UNITED NATIONS CONFERENCE ON TRADE AND DEVELOPMENT. UnctadStat reports, 2011. Accessed: July 2011, Available from: <http://unctadstat.unctad.org>.
- [148] VAN DER SPEK, S., AND VAN SCHAIK, J. *Urbanism on Track - Application of Tracking Technologies in Urbanism*, vol. 1 of *Research in Urbanism Series*. IOS Press BV, 2008.
- [149] VAN DER SPEK, S., VAN SCHAIK, J., DE BOIS, P., AND DE HAAN, R. Sensing human activity: GPS tracking. *Sensors* 9, 4 (2009), 3033–3055.
- [150] VAN HAGE, W. R. Simple event model prolog API, 2010. Accessed: April 2010, Available from: <http://eculture.cs.vu.nl/git/poseidon/sem.git>.
- [151] VAN HAGE, W. R., MALAISÉ, V., DE VRIES, G., SCHREIBER, G., AND VAN SOMEREN, M. Combining ship trajectories and semantics with the Simple Event Model (SEM). In *Proceedings of the ACM Workshop on Events in Multimedia* (2009), pp. 73–80.
- [152] VAN WIJK, J. J., AND TELEA, A. Enridged contour maps. In *Proceedings of IEEE Visualization* (2001), pp. 69–74.
- [153] VAN WIJK, J. J., AND VAN SELOW, E. R. Cluster and calendar based visualization of time series data. In *Proceedings of the Symposium on Information Visualization* (1999), pp. 4–9.

- [154] VESSEL TRACKER. Maritime business network: Vessels, ports, companies, and people, 2010. Accessed: February 2010, Available from: <http://www.vesseltracker.com>.
- [155] VROTSOU, K., FORSELL, C., AND COOPER, M. 2D and 3D representations for feature recognition in time geographical diary data. *Information Visualization* 9, 4 (2010), 263–276.
- [156] VT MÄK. Vr-forces: The complete simulation toolkit, 2010. Accessed: February 2010, Available from: <http://www.mak.com/products/vrforces.php>.
- [157] WANG, T., PLAISANT, C., SHNEIDERMAN, B., SPRING, N., ROSEMAN, D., MARCHAND, G., MUKHERJEE, V., AND SMITH, M. Temporal summaries: Supporting temporal categorical searching, aggregation and comparison. *IEEE Transactions on Visualization and Computer Graphics* 15, 6 (2009), 1049–1056.
- [158] WARE, C. *Information visualization: Perception for design*. Morgan Kaufmann series in interactive technologies. Morgan Kaufmann, 2004.
- [159] WEAVER, C., FYFE, D., ROBINSON, A., HOLDSWORTH, D., PEUQUET, D., AND MAC EACHREN, A. M. Visual exploration and analysis of historic hotel visits. *Information Visualization* 6, 1 (2007), 89–103.
- [160] WIJFFELAARS, M., VLIEGEN, R., VAN WIJK, J. J., AND VAN DER LINDEN, E.-J. Generating color palettes using intuitive parameters. *Computer Graphics Forum* 27, 8 (2008), 743–750.
- [161] WILLEMS, N., SCHEEPENS, R., VAN DE WETERING, H., AND VAN WIJK, J. J. *Visualization of Vessel Traffic*. 2012. to appear.
- [162] WILLEMS, N., VAN DE WETERING, H., AND VAN WIJK, J. J. Visualization of vessel movements. *Computer Graphics Forum* 28, 3 (2009), 959–966.
- [163] WILLEMS, N., VAN DE WETERING, H., AND VAN WIJK, J. J. Evaluating the visibility of vessel movement features in trajectory visualizations. *Computer Graphics Forum* 30, 3 (2011), 801–810.
- [164] WILLEMS, N., VAN HAGE, W. R., DE VRIES, G., JANSSENS, J. H., AND MALAISÉ, V. An integrated approach for visual analysis of a multisource moving objects knowledge base. *International Journal of Geographical Information Science* 24, 10 (2010), 1543–1558.
- [165] WILLEMS, N., VAN HAGE, W. R., DE VRIES, G., JANSSENS, J. H., AND MALAISÉ, V. An integrated approach for visual analysis of a multisource moving objects knowledge base. In *Proceedings of the AGILE Workshop Geospatial Visual Analytics: Focus on Time* (2010).
- [166] WILLIAMS, S., HECHT, M., PETERSEN, M., STRELITZ, R., MALTRUD, M., AHRENS, J., HLAWITSCHKA, M., AND HAMANN, B. Visualization and analysis of eddies in a global ocean simulation. *Computer Graphics Forum* 30, 3 (2011), 991–1000.

## Bibliography

125

- [167] WOLFRAM RESEARCH INC. Manipulate. Accessed: 3 February 2011, Available from: <http://reference.wolfram.com/mathematica/ref/Manipulate.html>.
- [168] WOOD, J., DYKES, J., SLINGSBY, A., AND CLARKE, K. Interactive visual exploration of a large spatio-temporal dataset: Reflections on a geovisualization mashup. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1176–1183.
- [169] WORLD SHIPPING REGISTER. Ship index, 2010. Accessed: April 2010, Available from: <http://www.e-ships.net>.
- [170] WORLD WIDE WEB CONSORTIUM. SPARQL specification, 2010. Accessed: February 2010, Available from: <http://www.w3.org/TR/rdf-sparql-query>.
- [171] XIAO, L., GERTH, J., AND HANRAHAN, P. Enhancing visual analysis of network traffic using a knowledge representation. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology* (2006).
- [172] YAN, Z., MACEDO, J., PARENT, C., AND SPACCAPIETRA, S. Trajectory ontologies and queries. *Transactions in GIS* 12, Sup. 1 (2008), 75–91.
- [173] ZHAO, J., FORER, P., AND HARVEY, A. S. Activities, ringmaps and geovisualization of large human movement fields. *Information Visualization* 7, 3-4 (2008), 198–209.



## Publications

The major part of this thesis is based on the following publications (presented in chronological order, starting with the most recent publication):

NIELS WILLEMS, ROELAND SCHEEPENS, HUUB VAN DE WETERING, AND JARKE J. VAN WIJK, Visualization of Vessel Traffic. *In Poseidon: Situational Awareness with Systems of Systems*, Pierre van de Laar, Jan Tretmans, and Michael Borth, Eds., Embedded Systems Institute (ESI), 2012, *to appear*, [161].

Overview paper related to all chapters

ROELAND SCHEEPENS, NIELS WILLEMS, HUUB VAN DE WETERING, AND JARKE J. VAN WIJK, Interactive Density Maps for Moving Objects. *In IEEE Computer Graphics and Applications*, 32(1), 2012, [130].

Related chapter: 5

ROELAND SCHEEPENS, NIELS WILLEMS, HUUB VAN DE WETERING, GENNADY ANDRIENKO, NATALIA ANDRIENKO, AND JARKE J. VAN WIJK, Composite Density Maps for Multivariate Trajectories. *In IEEE Transactions on Visualization and Computer Graphics (Proceedings of InfoVis 2011)*, 17(12), pp. 2518-2527, 2011, [128].

Related chapter: 6

NIELS WILLEMS, HUUB VAN DE WETERING, AND JARKE J. VAN WIJK, Evaluating the Visibility of Vessel Movement Features in Trajectory Visualizations. *In Computer Graphics Forum (Proceedings of EuroVis 2011)*, 30(3), pp. 801–810, 2011, [163].

Related chapter: 4

ROELAND SCHEEPENS, NIELS WILLEMS, HUUB VAN DE WETERING, AND JARKE J. VAN WIJK, Interactive Visualization of Multivariate Trajectory Data with Density Maps. *In Proceedings of IEEE Pacific Visualization Symposium*, pp. 147–154, 2011, [129].

Related chapter: 5

NIELS WILLEMS, WILLEM ROBERT VAN HAGE, GERBEN DE VRIES, JEROEN H.M. JANSSENS, AND VÉRONIQUE MALAISÉ, An Integrated Approach for Visual Analysis of a Multisource Moving Objects Knowledge Base. *In International Journal of Geographical Information Science*, 24(10), pp. 1543–1558, 2010, [164].

Related chapter: 7

NIELS WILLEMS, WILLEM ROBERT VAN HAGE, GERBEN DE VRIES, JEROEN H.M. JANSSENS, AND VÉRONIQUE MALAISÉ, An Integrated Approach for Visual Analysis of a Multisource Moving Objects Knowledge Base. *In Proceedings of the AGILE Workshop Geospatial Visual Analytics: Focus on Time*, 2010, [165].

Related chapter: 7

NIELS WILLEMS, HUUB VAN DE WETERING, AND JARKE J. VAN WIJK, Visualization of Vessel Movements. *In Computer Graphics Forum (Proceedings of EuroVis 2009)*, 28(3), pp. 959–966, 2009, [162].

Related chapter: 3

# Summary

## Visualization of Vessel Traffic

Moving objects are captured in multivariate trajectories, often large data with multiple attributes. We focus on vessel traffic as a source of such data. Patterns appearing from visually analyzing attributes are used to explain why certain movements have occurred. In this research, we have developed visualization methods that result in these patterns for large data sets of multivariate trajectories and aim to answer the following research question:

*How can analysts be supported to understand large amounts of trajectories with multiple attributes by using interactive, visual representations?*

We have taken two research directions: one based on density visualization and the other one on visual analytics. Our main contribution is a definition for a density field of trajectories that gives for each point in the field the fraction of the time an object stays in that point. We have used this density in three different ways. The basic approach, called *vessel density*, shows this density at different scales simultaneously and various (maritime) movement features become visible, such as sea lanes (maritime highways) and anchor areas (maritime parking places). The second approach, called *density maps*, allow us to use subsets described by attributes, such as a ship type, to show the various distributions for each individual subset. Density maps allows us to distinguish between various moments in time, to conduct basic anomaly detection, and to explore the various attributes available in the data. The third approach is called *composite density maps*. Based on experience and domain knowledge, it allows users to describe expressions for sophisticated density fields. With this approach it is possible to extract sea lanes, find objects moving in the wrong direction of a sea lane, and conduct risk assessment. Next to these three approaches, we have evaluated vessel density in comparison with other trajectory visualizations. It shows that for some maritime movement features vessel density excels (stops) and for other movement features (fast movers and finding sea lanes) vessel density performs not significantly worse than the other visualizations.

Next to the density approaches, we have created a visual analytics framework with integrated results from the Poseidon project that allows us to generate, simplify, enrich, and reason with trajectories, shown in an interactive matrix visualization with small

renderings in the cells. The key running example shows that a ship enters a harbour where it is not expected to be.

The results of this research can be used to improve situational awareness in, for instance, vessel traffic monitoring systems by including historical data in the analysis of live data. Since movement of objects is of interest in many different disciplines, such as in traffic monitoring systems, urban design, and animal migration, we expect the approaches to be useful for those application domains as well.

# Samenvatting

## Visualisatie van Scheepvaartverkeer

Bewegende objecten worden beschreven met meerwaardige trajectoriën, vaak grote hoeveelheden gegevens met meerdere attributen. We richten ons op scheepvaartverkeer als een bron van deze gegevens. Patronen die voortvloeien uit het visueel analyseren van de attributen kunnen worden gebruikt om uit te leggen waarom bepaalde bewegingen hebben plaatsgevonden. In dit onderzoek hebben we visualisatiemethoden ontwikkeld die deze patronen kunnen weergeven voor grote hoeveelheden meerwaardige trajectoriën, met als doel om de volgende onderzoeksvraag te beantwoorden:

*Hoe kunnen analisten worden ondersteund in het begrijpen van grote hoeveelheden trajectoriën met meerdere attributen door gebruik te maken van interactieve en visuele representaties?*

Ons onderzoek heeft twee verschillende takken: de een is gebaseerd op dichtheidsvisualisaties en de andere op visuele analyse. Onze belangrijkste bijdrage is een definitie voor een dichtheidsveld van trajectoriën dat voor elk punt in het veld de fractie van de tijd geeft dat een object in dat punt verblijft. We hebben deze dichtheid op drie verschillende manieren gebruikt. De basis aanpak, genaamd *scheepsdichtheid*, geeft de dichtheid weer op verschillende schalen tegelijk waardoor verschillende (maritieme) bewegingskenmerken zichtbaar worden, zoals zeeroutes (maritieme snelwegen) en ankerplaatsen (maritieme parkeerplaatsen). De tweede aanpak, genaamd *dichtheidskaarten*, laat ons door attributen, bijv. scheepstype, gedefinieerde deelverzamelingen gebruiken om verschillende verdelingen te laten zien voor elke individuele deelverzameling. Dichtheidskaarten geven ons de mogelijkheid om te differentiëren tussen verschillende momenten in tijd, om anomalieën te vinden en om de verschillende attributen te verkennen in de gegevens. De derde aanpak, genaamd *samengestelde dichtheidskaarten*, geeft de gebruiker de mogelijkheid om met behulp van ervaring en domeinkennis expressies te beschrijven voor de definitie van verfijnde dichtheidsvelden. Met deze aanpak is het mogelijk om zeeroutes te extraheren, spookrijders te vinden in een zeeroute en risicoanalyses te doen. Naast deze drie aanpakken hebben we scheepsdichtheid geëvalueerd in vergelijking met andere trajectorievisualizaties. Daaruit blijkt dat voor sommige maritime bewegingskenmerken scheepsdichtheid uitblinkt (stoppers) en voor andere bewegingskenmerken (snelle bewegingen en het vin-

den van zeeroutes) presteert scheepsdichtheid niet significant slechter dan de andere visualisaties.

Naast de dichtheidsaanpakken hebben we een framework gemaakt voor visuele analyse met de geïntegreerde resultaten van het Poseidon project. Dit geeft ons de mogelijkheid geeft voor het genereren, simplificeren, verrijken en redeneren met trajectoriën en welke weergegeven worden in een interactieve matrixvisualisatie met kleine tekeningen in de cellen. Het belangrijkste resultaat laat een schip zien dat in een haven aanmeert waar hij niet verwacht wordt te komen.

De resultaten van dit onderzoek kunnen worden gebruikt om het bewustzijn van een situatie te verbeteren in, bijvoorbeeld, systemen die de scheepvaart onder toezicht houden door deze uit te breiden met historische gegevens in de analyse van de actuele gegevens. Omdat bewegingen van objecten in vele andere disciplines van belang is, zoals verkeerstoelichtsystemen, stadsontwerpen en dierenmigratie, verwachten we dat onze aanpakken ook nuttig kunnen zijn in deze toepassingsgebieden.

## Acknowledgements

The trajectory of my research reaches a milestone with this thesis, and will proceed in a different direction. Movement has not only been topic of research, but also characterized my life during this period, both literally, in space and time, and symbolically. One of the attributes of this trajectory is the group of people that supported me. I would like to say “Thank you!” to all of you and mention some of you in particular.

First of all, I would like to thank my promotor prof. Jarke J. van Wijk, Jack, for giving me the opportunity to learn the skills of a researcher by travelling together through the world of visualization and around the globe. In the rather unforgiving world of science, you always unconditionally stood by my side as a creative, critical, innovative, funny, pleasant, and accessible mentor. Thank you for this great experience, it was an honor!

I also would like to thank my copromotor dr. Huub van de Wetering for his daily supervision. A PhD research is invariably accompanied with those moments of doubts, when the crucial help and the feeling of constant support make all the difference. Indeed, I have felt fortunate that our trajectories intertwined.

Furthermore, I like to thank the other members of the doctoral committee, prof. Menno-Jan Kraak, dr. Gennady Andrienko, prof. Bauke de Vries, prof. Eric Postma, prof. Mark de Berg, and prof. Arjeh Cohen, for their effort in the preparation of my thesis and the enthusiasm to be my opponents in the oral defense.

I am greatly indebted to Roeland Scheepens for his contributions to the papers we wrote together. It was a pleasure to exchange both our expertises about computer graphics and moving object visualization. I wish you all the best with your own promotion in the coming years! I like to thank also all other coauthors, Gennady Andrienko, Natalia Andrienko, Véronique Malaisé, Willem van Hage, Jeroen Janssens, and Gerben de Vries, to challenge me not only in visualization, but also in other disciplines.

The visualization group of Eindhoven University of Technology has always been a warm place to do *my thing*. Discussions about serious and senseless topics challenged me to look at life with a much broader perspective. I especially like to mention Danny Holten, who has been a sparring partner, and it is a pleasure to have you as my first employer. Thanks to all vis-members that were there during my stay, Michel Westenberg, Andrei Jalba, Miekeal Verschoor, Kasper Dinkla, Stef van den Elzen, Robert van Liere, Meivan Cheng, Dennie Reniers, Alex Telea, Hannes Pretorius, Romain Bourqui, Kees Huizing, Yedendra Shrinivasan, Jing Li, Tineke van den Bosch, and Elisabeth Melby.

The Poseidon project led to a close cooperation between academic and industrial

partners. In our daily life we experienced this with many meetings, which has built up a strong team spirit to shine at the demonstration days. I like to thank you all, Daniel Trivellato, Alberto González Sánchez, Davide Ceolin, Wil van der Aalst, Pieter Adriaans, Sandro Etalle, Arjan van Gemund, Eric Postma, Guus Schreiber, Hans-Gerhard Gross, Maarten van Someren, Marc Voorhoeve, Arjan Mooij, Éric Piel, Fabrizio Maggi, Christian Günther, Nicola Zannone, Ildiko Flesch, Fred Spiessens, Erik Luit, Jimmy Troost, Hans Hiemstra, Jacek Skowronek, Maurice Glandrup, Boudewijn Geerink, Wouter Roosenburg, Bob Eskes, Lucas Noldus, Wil van Dommelen, Frans Beenker, Frans Reckers, Michael Borth, Jan Tretmans, Pierre van de Laar, and Joris van den Aker.

Our case studies wouldn't be possible without the input of some members of the MOVE project, thank you, Jos van Doorn, Yvonne Koldenhof, Kees van der Tak, Erwin van Iperen, Stefan van der Spek, Tine van Langelaar, Arzu Çöltekin, and Peter Bak.

Thank you people at SynerScope, in particular Paul Buyink and Jan-Kees Buenen, to let me finish my PhD, before diving into the next challenge!

I am glad that after work I could rely on many good friends, with who I could enjoy life and who kept me on track. First of all, dr. Nedishka, girly, thank you for opening certain doors in your creative world, such as experiencing your Persian culture, showing your new techniques to improve your butt, or spending hours in the shopping streets. It has been a tough time, with ups and downs, but we both made it, I am proud of you! Our ways will split from here, but nevertheless the door will always be open for you.

Nata, you are an extraordinary girl and I am glad to know you. I enjoyed your appearance, remotely, far away, or in person, here or there, as a nice and warm distraction in sometimes crazy times. I like our talks about the different roads to take in life and other interesting subjects. Thank you for being around!

Eric, *kerel*, our challenge to be active in the gym didn't really pay off, despite we are there pretty often. Probably the chill-out sessions after that are probably a reason for that. You are my anchor and support, I appreciate it a lot!

Marco, dude, you still make me puzzled about your interest in complex mathematical shapes, while with simple recipes you can cook dinners that make my day!

I also like to thank all my other friends, not limited to the following list, Linda Elsackers, Hans Groot, Valeriu Savcenca, Kundan Kumar (thanks for finding the right words in this section!), Maaïke Wolfs, John van Dijk, Martijn Leunissen (thanks, proof reading my thesis!), Rob Verhoeven, Sofia Kaliarnta, and Ali Etaati. I like to thank the Casa poker group for the nice evenings in a casino ambience. Furthermore, I like to thank Marcel Roberscheuten for spending many hours milling the vessel density shown on the front cover.

Last but not least, I would like to thank my family in their native language. Pa, ma, zussie, oma's, en de rest van de familie. De laatste tijd had ik vaak het excuus er niet te zijn, omdat het druk was. De reden hiervoor hebben jullie nu in je handen; hier heb ik al die tijd aan gewerkt. Niemand heeft dit ooit gedaan, maar het ziet er tof uit, hè? Bedankt voor alle onvoorwaardelijke steun in goede en slechte tijden. *Hartstikke bedankt allemaal!*

Niels Willems, November 2011

## Curriculum Vitae

Niels Willems was born on the 28th of December 1983 in Moergestel, The Netherlands. He completed his secondary education at the St. Odulphus Lyceum in Tilburg. From 2002, he studied computer science at the department of Mathematics and Computer Science at Eindhoven University of Technology. He received his Master’s degree in 2007. His Master’s thesis is titled “Interactive visualization in large scale comparative genomics”, where he developed a visualization method to analyze similarities between DNA sequences. This research has been conducted in cooperation with Plant Research International in Wageningen.

In November 2007, he started on a BISK-funded PhD project at the same university under the supervision of prof.dr.ir. Jarke J. van Wijk and dr.ir. Huub van de Wetering. For his research, he developed techniques for the visualization of vessel traffic. This research is conducted in the Poseidon project hosted by the Embedded Systems Institute in Eindhoven and in close cooperation with industry represented by Thales Nederland in Hengelo and Noldus IT in Wageningen.

From November 2011, Niels will start working as a software architect at Synerscope in Eindhoven, a spin-off company of the visualization group where he obtained his academic degrees.