Data Integration and Visualisation for Demanding Marine Operations

Hao Wang*, Xu Zhuge*, Girts Strazdins*, Zheng Wei*, Guoyuan Li†, Houxiang Zhang†
* Big Data Lab, Faculty of Engineering and Natural Sciences,
Norwegian Unversity of Science and Technology in Aalesund, Norway.
† Mechatronics Lab, Faculty of Maritime Technology and Operations,
Norwegian Unversity of Science and Technology in Aalesund, Norway.

***Abstract*—Marine operations face the major challenge of increased complexity and human error still accounts for more than 75% of marine losses. Much more data are being collected for diagnostic and monitoring purposes with sensors on-board a vessel. Danger is that the overwhelming volume of data will cause *information overload problems*. In this paper, we explore data integration and visualisation techniques for maritime operations and present a proof-of-concept prototype for offline data integration and visualisation using real data from our industrial collaborators. This work is an important part of our initial efforts towards a *visual analytics* framework for maritime operations.**

***Index Terms*—Maritime operations, visual analytics, data integration, data visualisation.**

I. Introduction

Norway is ranked as the world's second largest nation in maritime operations. Advanced marine operation is becoming the core activity in the Norwegian maritime industrial cluster, and new areas of marine operations emerge for the so-called "after oil" era. Marine operations face the major challenge of increased complexity in technology and integrated operations involving multiple vessels and autonomous units. More importantly, while shipping safety has improved greatly, human error still accounts for more than 75% of marine losses.

Technologies are being adopted for acquiring monitoring data about how the vehicle and different components are behaving. Recently, with the intention of remote ship monitoring for better services for shipping customers, vessel builders started to adopt new sensor technology by installing different sensors for different components on board a vehicle and transmit data using satellite communications to land-based service centres, e.g., HEalth MOnitoring System (HEMOS) by Rolls-Royce Marine AS.

These systems provide more accurate and timely operational data, but they also introduce new danger to the operations: *information overload problem* (IOP) [1] – the captain receives so many alert messages that s/he may easily overlook important/vital ones. Therefore, it is urgently needed to develop and implement a new framework to integrate and visualize the monitoring data in an informative way to assist the on-board operations and onshore analysis.

The new paradigm *Big Data Analytics* (BDA) is quickly rising. It is highlighted in a statement from the United Nations that "the world is experiencing a data revolution" [2]. As a key step in BDA, data integration combines data from various sources and provides a unified view of these data, including data fusion, data purification, and data validation. The monitoring data on a vessel comes from many sources, in different formats and frequencies. i.e., the data are multifaceted [3]. For example, low frequency data include heading, speed, GPS and high frequency data include like vibration, torque. In addition, a major challenge for vessel data integration is the poor quality of the raw sensor data. Data cleaning aims to improve data quality using for example statistics, integrity constraints. However, it is generally very difficult to guarantee the accuracy of the data cleaning process without verifying it via experts or external sources [4]. Therefore, close collaboration with domain experts is essential for cleaning the monitoring data from marine vehicles.

Another major challenge is the visualization and user-interaction of the monitoring data. Marine operations are complicated and the monitoring data can be used in many different, even unforeseeable, ways. The user-interaction functionality in existing data management and visualising tools are either very limited or involved with complex query commands, which is not applicable for marine operations. Therefore, easy and intuitive data visualisation and interaction for both 1) on-board captains and crew members and 2) onshore online support teams and offline analysts are essential.

Visual Analytics (VA) combines automatic analysis techniques (the algorithmic modelling approach mentioned above) with interactive data visualisations. The seminal papers [1], [5] have shown that visual analytics enables a virtuous cycle of user interaction, parameter refinement for algorithmic analysis methods so as to achieve rapid correction and improvement of human's knowledge and decisions.

The city Aalesund is located at the heart of the northwestern Møre cluster of maritime industry. The SFI[1] Centre for marine operations has been set up in Norwegian University of Science and Technology in Aalesund in 2015 to support innovation for demanding marine operations. In the framework of our Innovation Norway project "GCE Blue Maritime Big Data" [6], we obtained the monitoring data from HEMOS by Rolls-Royce Marine AS covering three years' operation by one

Corresponding addr.: Hao Wang, hawa@hials.no, Norwegian Unversity of Science and Technology, Aalesund, Postboks 1517, 6025 Aalesund, Norway

[1]SFI is a prestigious research program in Norway to build up a Centre for *Research-based Innovation* for a certain area

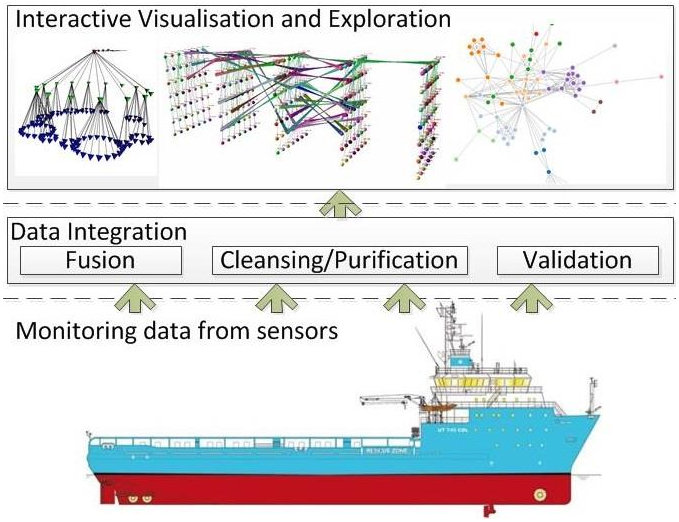

Figure 1. Integration and visualisation of monitoring data for demanding marine operations

vessel including high frequency machinery and low frequency vessel behaviour monitoring data.

To address the challenges in maritime operations, we aims to develop a visual analytics framework for maritime operations. This paper and papers [7], [8] represent our initial results in data integration and visualisation, efficient pattern identification, and prediction respectively, towards such a VA framework. The overview of data integration and visualisation of this paper is depicted in Figure 1. Monitoring data is collected from various sensors on board a vehicle, the data will go through the integration layer before it is visualised with intuitive interactive data exploration support.

We have built proof-of-concept prototypes which integrate and visualise the monitoring data along with a 3D animation of the vessel motions. More importantly, the prototypes allow easy data exploration w.r.t. spatiotemporal features, data correlations, and etc. The prototypes have received positive feedback from our industrial partners and is undergoing further improvements.

The remainder of the paper is structured as follows: Section II reviews existing literature on visual analytics with a focus on data visualisation. In Section III, we discuss in details the challenges we faced in integrating the real monitoring data and potential solutions and strategies. In Section IV, we discuss severel aspects of interactive data visualisation for the monitoring data. Section V presents our prototypes with discussions and Section VI concludes the paper and present some future directions.

II. Background

McKinsey Global Institute [9] has the following definition:

> "Big Data" refers to datasets whose size is beyond the ability of typical database software tools to capture, store, manage, and analyze.

However, there has always been "too much" data to analyse, why only in very recent years people claim that we are now in the "new era of big data"? Dean [10] gives a good observation on this phenomenon:

> The large data volume does not solely classify this as the big data era... What sets the current time apart as the big data era is that companies, governments, and nonprofit organizations have experienced a shift in behavior. In this era, they want to start using all the data that it is possible for them to collect, for a current or future unknown purpose, to improve their business.

Here in the Big Data Lab in NTNU Aalesund (BDL), we aim to develop the so-called *visual analytics* (VA) framework, that combines automated analysis techniques with interactive data visualisations, for maritime applications.

Visual analytics is a multidisciplinary field that includes the following focus areas [11]:

- *Analytical reasoning techniques* that enable users to obtain deep insights that directly support assessment, planning, and decision making;
- *Visual representations and interaction techniques* that take advantage of the human eye's broad bandwidth pathway into the mind to allow users to see, explore, and understand large amounts of information at once;
- *Data representations and transformations* that convert all types of conflicting and dynamic data in ways that support visualization and analysis;
- *Techniques to support production, presentation, and dissemination* of the results of an analysis to communicate information in the appropriate context to a variety of audiences.

Keim et al., in their seminal paper [1], explored the definition, process, and challenges of VA. They stated that the goal of VA is the creation of tools and techniques to enable people to:

- *synthesise* information and *derive* insight from data;
- *detect* the expected and *discover* the unexpected;
- *provide* timely and understandable assessments;
- *communicate* assessment effectively for action.

Shneiderman [12] proposed a well-known information seeking mantra: "Overview first, zoom/filter, details on demand". However, when the data are too large or complicated, more iterations with the human analyst will be necessary. Therefore, Keim et al. [1] extended the mantra to be: "Analyze first, show the important, zoom/filter, analyze further, details on demands" and introduced the seminal VA framework, depicted in Figure 2.

The use of visualisation can steer the analytical process [14]: the human analyst can interactively choose parts of data or change parameters of automatic analyses methods, the results of which can be further visualised and analysed with other methods.

One key element of VA, compared to the currently dominating approach of automatic (algorithmic) analysis, is the recognition of the importance of *information visualisation* in the human understanding and analysing process. Fekete et al. [5] has rigorously explored the value and benefits of information visualisation. Even as fully-automated algorithmic

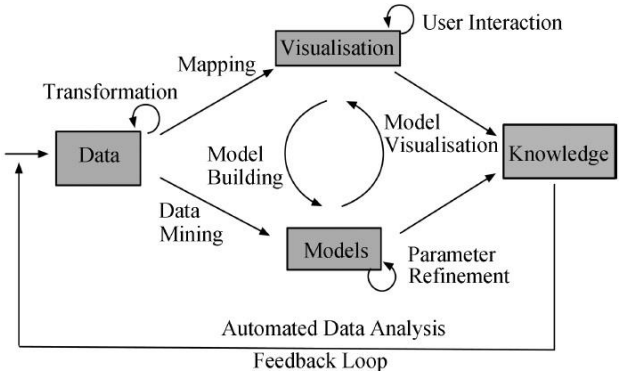

Figure 2. Visual Analytics Framework [13, Fig.1]

methods can quickly identify useful information and provide more accurate prediction, they lack the ability to interact with human and deliver effectively the knowledge. The combination of visualisation and automated algorithmic methods enables a virtuous cycle of user interaction, parameter refinement for algorithmic models so as to achieve rapid correction and improvement of human's knowledge and decisions.

Visual analytics is still relatively new for the maritime community. Maria Riveiro's Ph.D. thesis (2011) [15] on the detection of anomalous vessel behaviour in traffic and some following-up research on maritime traffic are the closest work we can find. The subject area, requirements, and sources of data of their work are different from our project and visual analytics framework, but we did get plenty of information and inspirations from them.

III. Data Integration

Data collected from ship on-board systems presents a set of challenges. In this section we identify the challenges based on our experience with real ship monitoring data sets. We discuss potential solutions and mitigation strategies.

A. Data in raw formats

Data logging systems used on vessels are not necessarily optimized for analytics and visualization. Rather the goal for these tools is to record as much data as possible with minimal loss of accuracy. As a result, the data is usually stored in formats well suited for fast sequential logging, without any indexing or other analytics capabilities. Typical formats include plain-text files in Comma-Separated-Value (CSV) format, Matlab data files, or custom binary files. These formats have their advantages: relatively small needed storage space, fast load time for applications analyzing the whole data set. However, these formats require linear algorithms for basically every operation: *O(n)*, where n is the number of data samples. These formats are not optimal for interactive visualizations or analytics, where we often need to select a subset of data:

- A specific time window: one month, day, second, and etc;
- Only a few columns (variables) from a table with tens of columns;
- Searching specific values, such as extremes or values in a specific range (e.g. geographical coordinates in a specified window).

These concerns raise a strong signal that data analysts should be ready to convert data to other formats more appropriate for integration, analytics and visualization. Choice of appropriate format may be application-specific. However, there are some common principles which we want to emphasize.

B. Relational databases versus NoSQL storage

One research direction in the last decade has been so called NoSQL data storage - data storage models not relying on relational databases and tables. While NoSQL includes a large variety of databases, the term is used to describe massively scalable solutions that can run on clusters of thousands of machines, such as MapReduce [16], BigTable [17] and Dynamo [18]. Another niche for NoSQL data storage is so called document stores, such as MongoDB [19], efficient in scenarios with data having dynamic and not predefined structure. We argue that NoSQL databases are useful in two scenarios: (1) for real-time analysis of live data streams where record pre-processing and indexing is not possible; and (2) for storage of data without a specified schema. Yet for marine operation data, especially for post-analysis and visualization, traditional relational databases are more appropriate due to several reasons:

1) Relational databases are well established and the implementations are well tested by both: open source community and commercial vendors;
2) SQL query language is very flexible and provides rich feature set. Moreover, it is better known among programmers and data analysts, therefore requiring less time to learn. NoSQL may require learning specific query languages and even a different mindset. However, tools exist (e.g. Pig latin [20], HiveQL [21]) providing SQL-like language on top of MapReduce NoSQL engines;
3) Unless the processing algorithms are really scaled to hundreds or thousands of machines, there is no real performance benefit. SQL engines have efficient query optimizers based on relational algebra;
4) Vessel monitoring data comes from many different sources, including on-board sensors. While the data types and value represent a large range, the structure of data is very well defined, at least at the individual vessel and operation level. Therefore constructing a schema for such data sets is straight forward.
5) Considering limited bandwidth of vessel communication channels during maritime operations, it is not practically feasible to transfer all the data to offshore data centers in real-time. Also, there is no room for huge data centers on board vessels due to strive for energy-efficiency (and hence space-efficiency). Therefore, typically the on-board systems log as much data as possible in raw format for post-processing, while uploading only a limited and pre-processed data set for near-real-time analysis onshore.
6) Parallel processing algorithms, such as MapReduce, divide data in sections which can be processed inde-

pendently. However, maritime operations are described with spatio-temporal multivariate data. Although each sample can be seen as independent (e.g., calculation of min/max values can be done in parallel), in many cases we are interested to work with derivatives, analyze and visualize the sequence of samples. Therefore, possibilities of parallel processing are reduced.

Based on the above arguments we conclude that relational databases with SQL interface are powerful tools satisfying wide range of data integration and visualization needs for marine operation analysis.

C. Black box

When analysts receive the data, to a great extent it is a *black box*. We may have different assumptions about the data. Yet considering the challenging conditions during maritime operations and complexity of the systems, different unpredicted results can be found in recorded data. We have to validate all our assumptions and get statistics to understand our data.

For example, are there overlaps of data? If file names include date and time, do they represent time of first or last record in the file, or something else? Are there gaps in data and how large? Although these questions may sound trivial, one can discover surprising facts while answering them. For example, our experience shows that some data logging systems may have overlaps in data samples. That may be due to internal buffering and data flushing procedures of the logging systems.

Several things are important in this regard. First, domain expert advice is very valuable. Especially, if these experts are close to the particular vessels and know the data logging systems. Second, the data integration system should include a generic framework for automated testing of data. It is similar to *unit testing* in software development: we define assumptions (or assertions) about the data and the system should be able to automatically detect and report violations. For example, the system can detect data overlaps, wrong timestamps, huge gaps, and etc.

D. Overlaps and gaps in data

As mentioned in III-C, the data can bring different surprises. One typical surprise is gaps in logged data. That can happen due to different reasons: lack of storage space on board, system errors or operational policies. Although less likely, data overlaps can also happen, especially when data from different sub-systems is merged and the sub-systems are not aware of the global state of the system.

Four things should be done in this regard. Most importantly: to detect the gaps and overlaps. The data integration system should do that automatically. Second, The statistics and rare incidents should be reported to analysts. Third, automatic elimination of duplicates should be done. This step is trivial in SQL, showing yet another advantage of relational databases. Forth, extrapolation of data may be useful to fill in short gaps. In this case every record should be marked: either original data, or calculated value. In this way we can filter necessary samples in queries, and exclude extrapolated values if necessary.

E. Different sampling frequencies

Data comes from different sources. Even inside one system there might be several modules with different sampling frequencies. In some scenarios we might want to compare and map values from different modules: e.g., to calculate correlations or detect anomalies. Several steps can address the challenge. First, it is important to have an exact time stamp for every data sample. Depending on the data format it may or may not be included in raw data. Second, clock synchronization of all the modules must be ensured. It is a complex problem in general, simplified solutions may exist in specific cases. Although synchronizing clocks before data logging is preferred, this could be outside the competence of data analysts. Therefore post-processing phase may require synchronization of modules with drifted clocks. Third, extrapolation of recorded data may be needed to calculate missing records and generate samples with the same frequency for all modules.

F. Large amounts of high-frequency data

Some data modules may have high sampling frequencies in kilohertz range, e.g., engine monitoring data. This can result in more than 100 million samples every day, more than 36 billion samples a year. Although SQL servers can store tables with terabytes of data (e.g. Microsoft SQL server [22]), additional attention may be required to ensure efficient data storage and queries.

First, it should be analyzed or empirically tested how the size impacts query speed. Second, upgrading hardware components to a cluster solution could be necessary. Different relational database management systems allow parallel queries and data sharding with appropriate hardware architecture.

Third, usable data resolution should be considered. For example, maybe our applications do not really use data with more than 100Hz frequency? Finding valuable information in the high-frequency data is dedicated topic on its own. E.g., by using Fourier Transformation the system could suggest how much information would be lost by down-sampling.

Fourth, one large table can be split into several smaller ones, e.g., storing one months data in each table. SQL allows to select from unions of tables. Such separation would decrease size of indexes.

And lastly, other storage alternatives should be explored for high-frequency data. An example: storing only hourly index in SQL, which links to plain-text files containing all values for the particular hour.

G. Sensitive data

The vessel data owners are very cautious when it comes to sharing the data. It is important to keep it safe, not exposing any details to third parties. It is typical to have a requirement for the data to stay in the country of origin. This fact limits the possibilities to use cloud-based solutions, such as Amazon AWS, for data storage and analysis. Even using clusters of external research collaborators may require additional agreements and procedures. Typically this leads to

local small clusters built inside each research institution. One technical thing that can be done: obfuscating some parameters of the data to make identification of a specific vessel or fleet manager impossible. Timestamps can be changed by a constant offset, geographic latitude and longitude can be shifted or removed if not relevant. However, such procedures can remove useful information.

H. Different systems and formats

Extracted data formats may differ from vessel to vessel, depending on manufacturer of different sensors and software modules. As a result, semantically the same data is encoded in different syntax. Several key points are important here.

First, the visualization should support generic data model . The goal of the integration system is to provide a conversion layer, which takes input in different raw formats, stores it in a common format for visualization and analysis. In addition, an automated data assessment module can generate statistics about the data: gaps, number of records, mean values, value distributions, and etc.

Second, the data analysts should be in close cooperation with domain experts to establish common standards and practices for data logging and streaming formats. This aspect is especially challenging considering the degree of conservatism in the industry. Introducing new concepts and standards is challenging. However, more and more maritime experts recognize the importance of data analytics for improvement of ship design and operations.

IV. Interactive Data Visualisation

Keim [23] suggested a classification of visualisation techniques in three dimensions:

- Data type to be visualised;
- Visualisation technique;
- Interaction and distortion technique.

We look at each of the three dimensions for the visual analytics of the maritime monitoring data as follows.

A. Data Types

There are 5 facets of data, as rigorously surveyed by Kehrer and Hauser [3]:

- *spatiotemporal* data that represent spatial structures and/or dynamic processes;
- *multivariate* data consisting of different attributes such as temperature or pressure;
- *multimodal* data stemming from different acquisition modalities (data sources);
- *multirun* data stemming from multiple simulation runs that are computed with varied parameter settings;
- *multimodel* data resulting from coupled simulation models that represent physically interacting phenomena or neighboring climate compartments such as ocean and atmosphere.

From our discussions in SectionIII, we can see that the data we obtained from HEMOS are *multifaceted*, they are spatiotemporal, multivariate, and multimodal data. If we also consider weather and oceans data, then they will become multimodel data. In this paper, we only consider the former two facets, leaving the multimodel facet as our next step.

B. Visualisation Techniques

The monitoring data from vessels contain multiple attributes in different spatiotemporal frequencies. The visualisation of multidimensional multivariate data remains a challenge [24], even after several decades of development [25].

For multidimensional data, *coordinated multiple views* [3] enable that different data variables can be shown, explored, and analyzed in multiple linked views placed side by side.The views can be histograms, scatter-plot matrices, parallel coordinates, or function graphs. Data can be selected or *brushed* [26] in a view, the related data items are instantly highlighted in all *linked views*. Views can be enabled or disabled and different parts of data can be filtered, *reduced*, and etc.

Exploring distributions or correlations among different data dimensions is the key to visualize the multivariate data. A lot of methods have been proved to effectively display and understand these correlations, e.g. projection-based methods [27], automatic analysis methods [28], and factor generation methods [29].

Glyphs [30] are a powerful and popular way of visualising multivariate data. Multiple data variables are represented by a glyph using a set of *pre-attentive visual stimuli* such as shape, size, width, color, or intensity. Relations between the data variables can be more readily perceived and compared than other techniques such as parallel coordinates and scatter-plot matrices. In addition, subsets of dimensions can form composite visual features, which enables a richer description of inter-record and intra-record relationships than can be extracted using other techniques.

The monitoring data from vessels are commonly divided in different modalities because of different types of sensors. The members or called nodes under different modalities need to be normalized to make them comparable [27].

In the systematic review of visualisation for temporal data by Aigner et al. [31], three dimensions of time are considered:

- *Linear time* versus *cyclic time*. Linear time assumes a starting point and defines a linear time domain with data elements from past to future. On the other hand, many natural processes are cyclic, for example, the cycle of the seasons;
- *Time points* versus *time intervals*. Discrete time points describe time as abstractions comparable to discrete euclidean points in space. Time points have no duration. In contrast to that, interval time uses an interval-scaled time domain like days, months, or years. In this case, data elements are defined for a duration, delimited by two time points;
- *Ordered time* versus *branching time* versus *time with multiple perspectives*. Ordered time domains consider things that happen one after the other. For branching time, multiple strands of time branch out, which facilitates description and comparison of alternative scenarios. This

type of time supports decision-making processes, where only one alternative will actually happen. Time with multiple perspectives allows more than one point of view at observed facts (for example, eye-witness reports).

The raw data we obtained are linear timed in time points. The cyclic time visualisation can help showing the recurring patterns like the same time point in each 24-hours-period or the same season in each 365-days-period. The branching time visualisation is useful in showing the results from *prescriptive analytics* [32], which uses technologies like simulation, decision support and expert systems to explore different *alternatives* (branches) and provide recommendations on the course of action of a decision maker.

Many application areas can treat the temporal and spatial data just like any other data attributes. However, the monitoring data for vessels are mainly collected from different temporal data sources along with geospatial data sources like GPS coordinates, which play a central role in the analytical tasks. In addition, an important observation is that in the context of maritime operations, visualising the movement of the vessel w.r.t. the time and location axes is often useful for analysts to understand the behaviour and identify patterns and problems, e.g., propeller ventilation [7].

C. Interaction Techniques

Kerren and Schreiber [14] presented a taxonomy of interaction techniques, based on the ones proposed by [33], [34]. We present as follows a customised taxonomy according to the needs of monitoring data for maritime operations.

(A). Data and View Specification

1. Encode/Visualize: Users can choose the visual representation of the data records including graphical features, such as color, shape, etc.
2. Reconfigure: Some interaction techniques allow the user to map specific attributes to graphical entities. An example is the mapping of attributes in a multivariate data set to different axes in a scatter plot.
3. Filter: This technique is of great importance for visual analysis as it allows the user to interactively reduce the data shown in a view. For temporal monitoring variables, dynamic queries by using time-range sliders are commonly used.
4. Sort: Ordering of records according to their values is a fundamental operation in the visual analysis process.
5. Derive: The completion of additional computations based on the primary input data. Thus, the user might integrate results of statistical computations (aggregation, medians, . . .) into the data to be visualized.
6. Adjust: Related to the previous interaction type is the modification of parameters for automatic analyses (incl. simulations). In this way, actions in the data space using (configurable) computational methods instead of visual ones can be performed.

(B). View Manipulation

1. Select: Selection is often used in advance of a filter operation. The aim is to select an individual object or a set of them in order to highlight, manipulate, or filter out them. An example for maritime operation is to display the vessel on a virtual map to highlight its route.
2. Navigate/Explore: This important class of interaction techniques typically modify the level-of-detail. Well-known approaches are focus&context, overview&detail, zooming&panning, or semantic zooming.
3. Coordinate/Connect: Linking a set of views or windows together to enable the user to discover related items. Brushing and linking techniques (e. g., histogram brushing) are used in almost all VA systems.
4. Organize: Large systems often consist of several windows and workspaces that have to be organized on the screen. Adding and removing views can be confusing to the analyst. Some systems help the user to better overview and to preserve his/her mental map by grouping of views or by assigning specific places where they have to appear.

(C). Process and Provenance

1. Record: Methods that store and visualize the interaction history performed by the user help to facilitate the iterative analysis process. Undo or redo operations are the most simple examples for such techniques. More advanced ones allow to log complete user sessions (i.e., storage of all user actions such as zooming, filtering) and can be used to revisit states of the analysis or for collaborative work as colleagues might import a complete analysis trail of someone else.
2. Annotate: Graphical or textual annotations help the analyst to point to elements or regions within the visual representation. These annotations also might be links to other views.
3. Share: Collaboration in VA often occur in practice, but it is still not very well researched. A VA system has to support discussions, dissemination of results, or interactions of several analysts at the same place and the same time (co-located) or at different places and not necessarily at the same time (distributed). Sharing views or publication of visualizations are examples of important requirements for efficient collaboration between many analysts.
4. Guide: The specification of workflows is difficult for VA tasks as the related analytical processes are typically non-linear. A result can thus be reached by various interaction and analysis paths within the system. It would be beneficial if a VA system would support “guided analytics to lead analysts through workflows for common tasks”.

V. Prototypes and Discussion

A. A Data Integration Prototype

We have implemented a proof-of-concept data integration prototype in Python. We have tested it using part of the data we obtained from HEMOS.

The prototype includes several useful functions:

- Converting data samples to a generic format and importing them to a MySQL database.
- Detecting and removing duplicate data samples. The source data was distributed among many small files. The integration script collects them all together and eliminates duplicates.
- Generating statistics of data records: how long are the periods of uninterrupted (consecutive) data, how large are the gaps where we do not have any data, and statistics on overlaps in data. In addition to calculating the numbers, the system can generate histogram plots.
- A script that can select a projection of the data samples in a specified time window, with several resolution options (one second, one minute, ten-minutes). The data is exported in JSON format, as a generic Web API, allowing wide range of applications to use it. Currently, the visualization with D3.js is using this as a data source.

Histogram plots of the intervals in our test-data set are depicted in: the consecutive data periods (Figure 3), gaps (Figure 4) and overlaps (Figure 5). Most of the consecutive periods with data are shorter than 90 minutes, while majority of gaps are less than 30 hours long. The length of data intervals is more spread while the gaps are very strictly limited: only three outliers are longer than 30 hours.

We can also see interesting facts about overlaps. First of all, it is very important to see that there are overlaps. When our research group initially received the test-data set, it was not expected to have any overlaps in the data. The performed analysis suggests that there is a large number of overlaps in the range up to 1 minute (with several outliers 3 minutes long). Interestingly, the number of overlaps increases if we focus closer to one minute range: there are clearly much more overlaps with length from 30 to 60 seconds, compared to overlaps of 30 seconds or less. The 60-second boundary makes the analysts concerned. This topic will clearly be discussed with domain experts to find out potential reasons for the overlaps.

Currently, we store low-frequency data in the MySQL database. As the future work we identify exploring possibilities to extend the framework for high-frequency data support.

B. Two Interactive Visualisation Prototypes

We implemented two interactive visualisation prototypes. The prototype visualization developed with Javascript and D3.js is depicted in Figure 6. D3.js[2] is an open-sourced JavaScript library. The library provides a large collection of data manipulation and visualization components, allowing developers to bind arbitrary data to a *Document Object Model* (DOM), without being tied to a proprietary framework.

To visualise the multivariate facet of the data, we used the force-directed graph and chord diagram to visualise the correlations between data variables. In the force-directed graph, a physics engine is embedded to enable dynamic interaction. In the graph, 1) each node represents a different variable; 2) the thickness of the link between two nodes represents the correlation coefficient between the two variables; (3) the user can select and drag any node, if a node is more correlated with more nodes, then this variable has a stronger effect or "force" on all other variables; 4) when one node is selected, the links to all variables that are correlated to it are highlighted. The chord diagram is another graph visualising the quantified correlations. When the user select one variable, all variables that are correlated to it are highlighted.

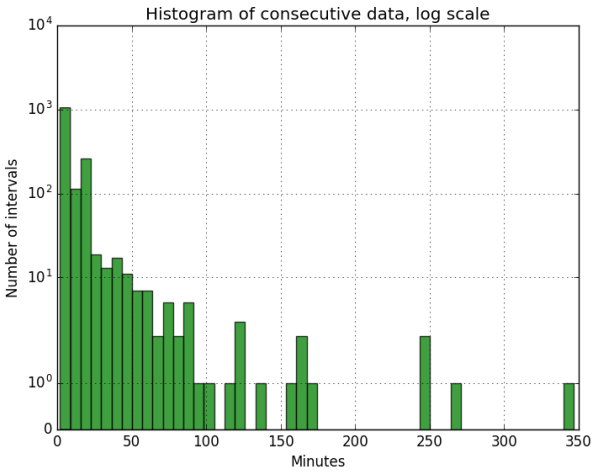

Figure 3. Consecutive periods of uninterrupted data in our test-data set

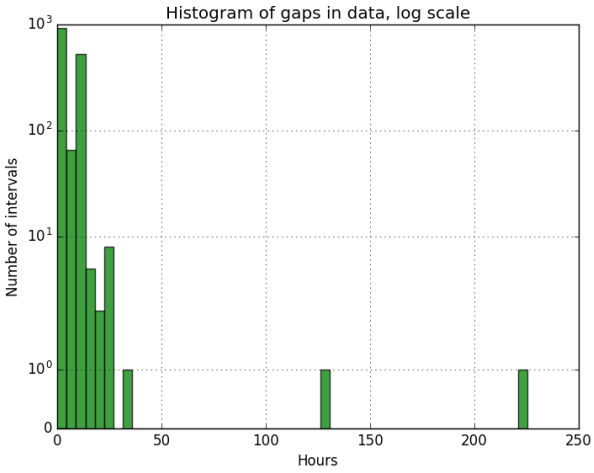

Figure 4. Gaps in our test-data set

We implemented an interactive line chart for timed exploration of different variables. Once the user select a time period, the changes of variables and the motion of the vessel will be animated synchronously. With WebGL and Three.js[3], we implemented the 3D vessel motion animation, in which we embedded an interactive 2D trajectory visualisation using the Google Maps® Javascript API.

The prototype developed with Tableau® is depicted in Figure 7. As Tableau® does not support dynamic graphs such as force-directed graph, we used glyphs to visualise

[2]https://d3js.org/

[3]http://threejs.org/

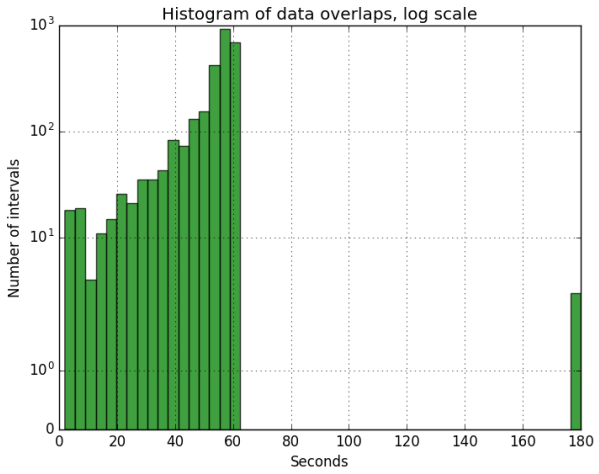

Figure 5. Overlaps in our test-data set

the correlations between variables. The main advantage of Tableau® is that it supports "drag and drop" in creating data visualisations, without the need of any programming. However, Tableau®, as a commercial software, has limitations by itself compared to Javascript and D3.js. If the cost is not a concern, then developing some components in Javascript and D3.js, and using the dashboard or storybook of Tableau® as the output interface through its Javascript API, could be a good hybrid approach.

VI. Conclusion and Future Work

In this paper, we present a detailed review on data integration and visualisation techniques in the context of monitoring sensor data from maritime vessel operations. Following the principles in our discussions, we present proof-of-concept prototypes for offline data integration and visualisation.

There is still several important aspects that we have to explore before we can build up a solid visual analytics framework. For data integration, although we argue that relational databases have advantages for our current stage, we envision that transition to NoSQL could be necessary in the future in different scenarios. First, cost and size of computing power and data storage is decreasing. We have already seen a shift from large computer-rooms to mobile phones. It is likely that in the future low-cost, high-performance computing will be available in the size of a match-box. At this stage it makes sense to build a computing cluster on board a vessel. Second, onshore operational centers manage a fleet of vessels in real time. Large amounts of data are fused here, and we can see each vessel as an independent data source. Communication channel bandwidth is also subject to increase. We therefore predict, that vessels will send high-frequency multivariate data to onshore centers. It can be in the form of real-time streams, or previously buffered data downloaded on request. Some of the usage scenarios include emergency situation detection where alarms should be raised in near-real-time, therefore high-performance scalable analytics becomes important.

For data visualisation, many visualisation and interaction techniques need to be included in our prototype, esp. the ones in categories "(B) View Manipulation" and "(C) Process and Provenance".

Acknowledgements

The authors would like to thank for the valuable inputs from Ibrahim A. Hameed, Bikram Kawan, Rune Valle and other members from BDL; Rune Garen, Leif R. Solaas, Thomas Oksavik, Krzysztof Swider, and Are Folkestad from Rolls-Royce Marine AS; Per Erik Dalen from Aalesund Knowledge Park AS and GCE BLUE Maritime. This research is partially supported by the project "GCE BLUE Maritime Big Data" funded by Innovation Norway and the project "AIS and Maritime Transport" funded by MarKom 2020.

References

[1] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, "Visual analytics: Definition, Process, and Challenges," in *Information Visualization*, ser. LNCS, vol. 4950. Springer, 2008, pp. 154–175.

[2] UN Global Pulse, "Big Data for Development : Challenges & Opportunities," *United Nations*, 2012.

[3] J. Kehrer and H. Hauser, "Visualization and visual analysis of multifaceted scientific data: A survey," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 19, no. 3, pp. 495–513, 2013.

[4] X. Chu, J. Morcos, I. F. Ilyas, M. Ouzzani, P. Papotti, N. Tang, and Y. Ye, "KATARA: Reliable data cleaning with knowledge bases and crowdsourcing," *Proc. VLDB Endow.*, vol. 8, no. 12, pp. 1952–1955, Aug. 2015.

[5] J.-D. Fekete, J. J. Van Wijk, J. T. Stasko, and C. North, "The value of information visualization," in *Information Visualization*, ser. LNCS, vol. 4950. Springer, 2008, pp. 154–175.

[6] H. Wang, O. Osen, G. Li, W. Li, H.-N. Dai, and W. Zeng, "Big data and industrial internet of things for the maritime industry in northwestern norway," in *TENCON 2015: IEEE Region 10 Conference*, 2015.

[7] H. Wang, S. Fossen, F. Han, and I. A. Hameed, "Data-driven Identification and Analysis of Propeller Ventilation," in *Oceans 2016: MTS/IEEE Oceans Conference*, 2016.

[8] G. Li, B. Kawan, H. Wang, A. Styve, O. L. Osen, and H. Zhang, "Analysis and Modelling of Sensor Data for Ship Motion Prediction," in *Oceans 2016: MTS/IEEE Oceans Conference*, 2016.

[9] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. H. Byers, "Big data: The next frontier for innovation, competition, and productivity," McKinsey Global Institute, Tech. Rep., 2011.

[10] J. Dean, *Big Data, Data Mining, and Machine Learning: Value Creation for Business Leaders and Practitioners*. John Wiley & Sons, Inc, 2014.

[11] J. J. Thomas and K. A. Cook, Eds., *Illuminating the path:[the research and development agenda for visual analytics]*. IEEE CS Press, 2005.

[12] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Visual Languages, 1996. Proceedings., IEEE Symposium on*. IEEE, 1996, pp. 336–343.

[13] G.-D. Sun, Y.-C. Wu, R.-H. Liang, and S.-X. Liu, "A survey of visual analytics techniques and applications: State-of-the-art research and future challenges," *Journal of Computer Science and Technology*, vol. 28, no. 5, pp. 852–867, 2013.

[14] A. Kerren and F. Schreiber, "Toward the role of interaction in visual analytics," in *Proceedings of the Winter Simulation Conference*. Winter Simulation Conference, 2012, p. 420.

[15] M. J. Riveiro, "Visual analytics for maritime anomaly detection," Ph.D. dissertation, Örebro university, 2011.

[16] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, Jan. 2008.

[17] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R. E. Gruber, "Bigtable: A distributed storage system for structured data," *ACM Trans. Comput. Syst.*, vol. 26, no. 2, pp. 4:1–4:26, Jun. 2008.

[18] G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, P. Vosshall, and W. Vogels, "Dynamo: amazon's highly available key-value store," in *ACM SIGOPS Operating Systems Review*, vol. 41, no. 6. ACM, 2007, pp. 205–220.

[19] K. Banker, *MongoDB in Action*. Greenwich, CT, USA: Manning Publications Co., 2011.

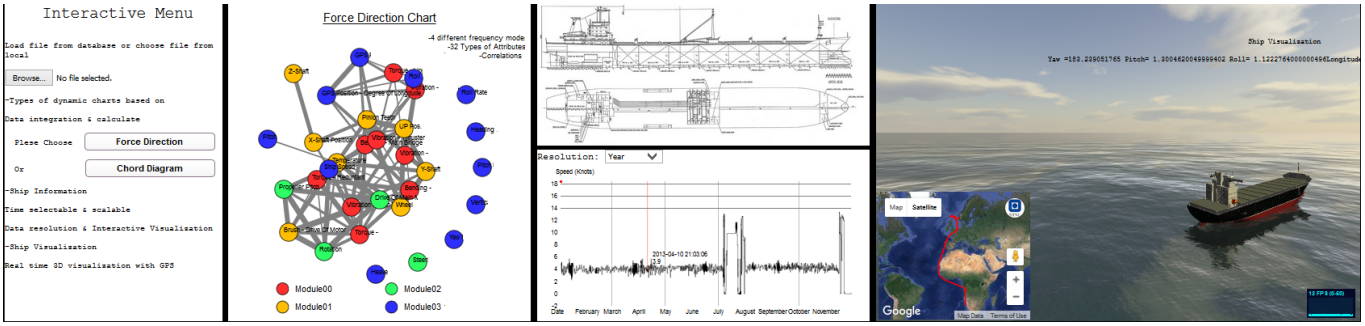

Figure 6. Visualisation prototype with Javascript and D3.js

Figure 7. Visualisation prototype with Tableau®

[20] C. Olston, B. Reed, U. Srivastava, R. Kumar, and A. Tomkins, "Pig latin: a not-so-foreign language for data processing," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. ACM, 2008, pp. 1099–1110.

[21] A. Thusoo, J. S. Sarma, N. Jain, Z. Shao, P. Chakka, S. Anthony, H. Liu, P. Wyckoff, and R. Murthy, "Hive: a warehousing solution over a map-reduce framework," *Proceedings of the VLDB Endowment*, vol. 2, no. 2, pp. 1626–1629, 2009.

[22] Microsoft Corporation, "Maximum Capacity Specifications for SQL Server," 2016, [Accessed 2016-02-22]. [Online]. Available: https://msdn.microsoft.com/en-us/library/ms143432.aspx

[23] D. A. Keim, "Information visualization and visual data mining," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 8, no. 1, pp. 1–8, 2002.

[24] C. Johnson, "Top scientific visualization research problems," *Computer graphics and applications, IEEE*, vol. 24, no. 4, pp. 13–17, 2004.

[25] P. C. Wong and R. D. Bergeron, "30 years of multidimensional multivariate visualization." in *Scientific Visualization*, 1994, pp. 3–33.

[26] R. A. Becker and W. S. Cleveland, "Brushing scatterplots," *Technometrics*, vol. 29, no. 2, pp. 127–142, 1987.

[27] P. Joia, F. V. Paulovich, D. Coimbra, J. A. Cuminato, and L. G. Nonato, "Local affine multidimensional projection," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 17, no. 12, pp. 2563–2571, 2011.

[28] A. Tatu, G. Albuquerque, M. Eisemann, P. Bak, H. Theisel, M. Magnor, and D. Keim, "Automated analytical methods to support visual exploration of high-dimensional data," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 17, no. 5, pp. 584–597, 2011.

[29] C. Turkay, A. Lundervold, A. J. Lundervold, and H. Hauser, "Representative factor generation for the interactive visual analysis of high-dimensional data," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 18, no. 12, pp. 2621–2630, 2012.

[30] M. O. Ward, "Multivariate data glyphs: Principles and practice," in *Handbook of data visualization*. Springer, 2008, pp. 179–198.

[31] W. Aigner, S. Miksch, W. Muller, H. Schumann, and C. Tominski, "Visual methods for analyzing time-oriented data," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 14, no. 1, pp. 47–60, 2008.

[32] D. Delen and H. Demirkan, "Data, information and analytics as services," *Decision Support Systems*, vol. 55, no. 1, pp. 359–363, 2013.

[33] J. S. Yi, Y. ah Kang, J. T. Stasko, and J. A. Jacko, "Toward a deeper understanding of the role of interaction in information visualization," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 13, no. 6, pp. 1224–1231, 2007.

[34] J. Heer and B. Shneiderman, "Interactive dynamics for visual analysis," *Queue*, vol. 10, no. 2, p. 30, 2012.