Sensor Data Fusion Middleware for Cooperative Traffic Applications

Teemu Leppänen [1)], Mikko Perttunen [1)], Pekka Kaipio [2)], Jukka Riekki [1)]

[1)] Department of Electrical and Information Engineering
University of Oulu
Oulu, Finland
{teemu.leppanen, mikko.perttunen,
jukka.riekki}@ee.oulu.fi

[2)] Department of Information Processing Science
University of Oulu
Oulu, Finland
pekka.kaipio@oulu.fi

***Abstract* – Recent advancements in information and communication technology accelerate the development and deployment of intelligent transportation systems. Cooperative traffic systems utilize information collected by road users and traffic operators about their immediate environment sharing it system-wide, enabling services for all traffic entities. In this work, we contribute a data processing and data fusion middleware relying on wireless mobile sensor nodes and public infrastructure services for traffic data collection for future cooperative traffic applications and services. The system is demonstrated with an example application, utilizing mobile phones with integrated sensors as sensor nodes and producing the visualization of travelled routes annotated with travel mode and detected anomalies of road surface. Middleware features include modular component-based architecture, reconfigurable and reusable data processing components, dynamic addition of components and interfaces into the system in runtime and data fusion of heterogeneous sensor data through chaining of components into application specific data fusion levels.**

Keywords - Cooperative systems; sensor networks; wireless sensing; information fusion.

I. INTRODUCTION

Increasing bandwidth and 3G technologies available for wireless communication enable the advancement of intelligent transportation systems towards cooperative traffic systems, where traffic entities collect and share data and become more and more context-aware [1]. In [2], cooperative traffic is given the following description: “Road operators, infrastructure, vehicles, their drivers and other road users will co-operate to deliver the most efficient, safe, secure and comfortable journeys. The vehicle-to-vehicle and vehicle-to-infrastructure cooperative systems will contribute to these objectives beyond the improvements achievable with stand-alone systems.” These actors in traffic can be provided with information, for example, about traffic disorders, road conditions, current weather conditions and usage statistics as well as routes based on dynamic traffic information [3].

The increased need for accurate traffic information has led to the integration of offline traffic information, such as fixed road infrastructure sensor data, with real-time information from supplementary data sources, such as cameras, GPS and cell phone tracking [4]. In addition of providing complementary data, these data sources assist in decreasing uncertainly of the data from individual sources and enhance the decision maker’s performance [4][5]. This will result in many traffic engineering problems to become data fusion problems, producing estimates or an improved model of the system being observed [4]. Different techniques for data fusion exist: statistical approaches, probabilistic approaches and artificial intelligence [4]. Sensor data fusion techniques can be characterized in several domains [5]. In the application domain, only application relevant concepts are considered. In the fusion objective domain, the objectives of the fusion are considered, for example object recognition. The selected sensor types dictate compatibility and complementary nature of different sensors. The sensor configuration, as in concurrent or temporally separated measurements, defines the sensor usage. Finally, the fusion process is usually described as a three-level hierarchy: data fusion, feature fusion and decision fusion. In any of the three levels spatial or temporal fusion may occur [5]. This model with the hierarchical levels with a varying level of detail in information has been widely accepted [4][5].

Participatory sensing is a method for collecting data from many unknown and independent contributors in collaboration [6]. Cooperation between contributors can take several forms [7]. Informative cooperation occurs when users extend the system range by transmitting their sensory data in one-way communication. In descriptive cooperation, sensory data is augmented with intentions such as the intended direction of movement of the user. Coordinative communication allows objects to reason about and modify their behavior depending on other’s intentions, which requires two-way communication. The key question here is: What information should be shared and in which level of processing and fusion? [7] Performing data processing and fusion refines the data and reduces the amount of data transferred in the system. However, it also reduces the sensor network coverage, reliability and accuracy of the data as fewer samples are available. Communication technology issues, such as wireless channel capacity, also limit coverage and reliability.

Wireless (mobile) sensor networks offer several advantages over fixed sensor networks [8]. First, the coverage of a network can be extended easily with smaller costs. Second, nodes can store data locally and provide data only when requested, which contributes towards scalability and energy saving. Third, if a fixed sensor node fails or experiences network failures, mobile data collectors can be used to compensate the data loss. Fourth, localized algorithms in nodes can be utilized in data collection and processing tasks. Khanafer et al. [9] define two main categories for wireless sensor networks for intelligent transportation systems: planar and multi-tiered. Planar architectures utilize mobile sensors and can have infrastructure-less communication paradigm supporting vehicle-to-vehicle networks, forming mobile ad hoc wireless sensor networks. Another paradigm for planar architectures is infrastructure-based supporting vehicle-to-infrastructure and infrastructure-to-vehicle communication by utilizing fixed roadside units. Roadside units relay data to services and can also form clusters of members around them. Some challenges for planar architectures exist, related to network topology and scalability. Furthermore, there are known constrains such as processing capabilities of the nodes, which affects to quality of service among other things. Networks in the second category, multi-tiered architectures, can have better performance by supporting heterogeneous communication, such as wireless local area networks providing more available bandwidth. This means having better quality of service when computations can be relayed to more powerful units in upper layers utilizing the increased bandwidth available. Of course, enabling more communication technologies adds complexity and costs to the system.

As mobile phones are today widely used, equipped with sensors and interfaces to external sensors and they have processor, memory, battery and communication units, they can be also seen as sensor nodes [10]. Mobile data collectors, however, have communication issues when transferring data to the sinks as bandwidth is limited [8]. Also, battery life, processing power and memory capacity are limited. Normal phone usage will limit the data collection capability [10]. Data routing is influenced by the network topology, the quality of the service parameters and contents of the data [8].

The Finnish Cooperative traffic program [11] envisions sustainable traffic using extensive information sharing based on novel technologies and services. In the Sensor Data Fusion and Applications project as a part of this program, our focus is on utilizing new sensor data sources and data fusion methods for generating new potential applications [11]. Our vision in [1] includes the usage of mobile phones with integrated sensors, giving information on the behavior of the actors in traffic. Instead of focusing on tailored applications on the field, *our goal is to develop a flexible system that can be used to demonstrate and analyze a variety of cooperative traffic scenarios and applications working cooperatively with little development overhead.* To enable this we derive a wide set of requirements from the literature and our example scenarios and take a bottom up approach, contributing a practical data collection system with the ability to integrate heterogeneous sensor data and infrastructure services as data sources, data fusion capabilities and pluggable data processing components. The demonstration system includes a visualization client application that produces annotated maps and enables end-to-end demonstrations. In this paper, we report the system prototype and an example application featuring real-time travel mode and road surface anomaly detection, with mobile phones with integrated sensor as wireless sensor nodes.

The rest of this paper is organized as follows: Section II reviews the related work in the area. In Section III, we define requirements and scenarios for a cooperative traffic sensor network and describe our system. In Section IV, we compare the developed system against the requirements and represent field test results. Finally, in Section V, thoughts for future work are given.

II. RELATED WORK

We describe here a number of existing applications utilizing mobile traffic-related sensor data and existing solutions for data processing and fusion for intelligent transportation systems.

A. *Traffic sensing applications*

Several participatory sensing systems have been reported. In the Mobile Millennium project [12], GPS enabled mobile phones are used for the collection of traffic data, which is then fused with data from sensors in the road infrastructure. The system is employed for monitoring and estimating the traffic flow in real-time. The produced estimates are then transferred back to the phones. Google Maps for Mobile [13] offers live traffic conditions monitoring by retrieving the speed of vehicles from the GPS data from the mobile phones. TJam [14] uses GPS receivers to predict traffic jams by measuring the velocity of the vehicles. Users' coordinates are transmitted to the service and the probabilities of congestion in a given region are sent back. In Nericell [10], mobile phones with accelerometer sensors are used in vehicles to detect road bumps, among other features. The detection software is installed in the mobile phone itself and they introduce the concept of triggered sensing where less energy consuming sensors are used to trigger the usage of other sensors.

B. *Data fusion platforms*

In the following we describe several platforms developed for processing traffic-related information from sensor networks. The hierarchical Content Delivery Network architecture described by Elshenawy and others [15] has network hierarchies to provide scaling. Vehicles contain on-board units acting as clients in the network. Road-side units work as a surrogate servers having storage for the content and self-healing mechanism for communication failures in the network. In the next hierarchy level, geographical areas are grouped into geographical domains, which are controlled by domain managers. It is the manager’s responsibility to dynamically route content to the vehicles under their domain, which should also decrease delivery time. Domain managers can be grouped together recursively. Services sent add and delete messages, which propagate through the network nodes, based on the geographical area the service covers. On the other hand, on-board units can send discovery messages to locate available services in a geographical area.

In the Cooperative Vehicle Infrastructure System [16][17] project, OSGi open source platform has been selected as the application platform. Communication is done using the CALM standard, providing both wireless local area and cellular network access, which can be selected dynamically by having better quality of service. The system consists of three layers, in the upper layer is the central management working on the system-wide level. The middle layer represents the roadside infrastructure at the regional level and the lowest level represents the individual vehicles.

Tacconi et al. [18] describe an information retrieval scenario, where mobile sinks, for example vehicles, query data from edge nodes in the wireless sensor network. Sensor nodes and sink nodes are aware of their geographical position or spatial distribution. Data routing from the nodes to mobile sinks is done by predicting the current position of the mobile sink based on its mobility information within the network from the original location of the injected query.

The Telematic Management System [19] component suite has many similarities with our approach. It has three main components: the kernel, communication subsystem and the data module. Framework users provide a set of decision modules accessing the communication system and data module. The decision modules and their dependencies on other modules are defined as graphs in XML documents. With the decision modules, users define a protocol component, which propagates incoming messages to interested modules through the graph. The kernel provides interfaces for the decision modules and initiates them. Data modules define a mechanism to access the data concurrently. The communication subsystem uses TCP/IP asynchronously. For each vehicle there is a local queue in the subsystem, which is used to store messages directed to it. Messages from the vehicles are stored in the general system queue.

Lv et al. [21] define ubiquitous intelligent transportation system architecture as vehicle-to-vehicle or vehicle-to-infrastructure communication with access to large-scale information systems. Their system utilizes multi-source real-time location-based mobile sensor data and fixed sensor data via context-awareness technology. The key idea is the interaction between any devices in any location or time. In their system there are five layers. The resource management layer collects data from the mobile sensors and sensors in the roadside, for example weather stations. Preprocessing of the data and classification is also done in this layer. The information awareness layer fuses the data requested by the services running in the system. Next, the intelligent service layer provides access to data and services from other service providers. These services may be accessed via websites or mobile phones for example. The terminal application layer provides the dynamic system level services such as real-time traffic information or road guidance to personal computers and other terminals. Finally, the ubiquitous network layer provides hybrid network communication models to support usage of multiple communication technologies.

III. System and Middleware

We identify various cooperative traffic scenarios to sketch the requirements for the data processing system. In the scenarios, the target users include a car driver, a pedestrian and a traffic operator.

One scenario sketches services supporting efficient responses to traffic accidents. If an intelligent vehicle notices changes in road conditions, or for example detects a traffic accident from abrupt braking, the vehicle can send this information to be added to the real-time situation model of the road segment. The new information can then be disseminated to other traffic entities in the system. The traffic accident location can be inserted to a database. Moreover, a real-time warning can be issued to the drivers planning to drive in that road by updating the route prediction data in their vehicle navigators. A weather service integrated into the system can be used to determine the current local weather and for example lighting conditions when the accident happened. This is useful information at least during the Nordic winter. This information aggregated with the road condition data and on-board diagnostics data from the vehicle can be stored as accident information to the database, where it is available for further analysis by road authorities. In the future, this allows drivers to be warned to avoid accidents when similar weather and road conditions appear. In addition, mobile phone data or roadside sensors, such as cameras, can be used to detect approaching dangerously speeding vehicles. Information about speeding vehicles, aggregated with the real-time road situation information, can produce an alert issued to other entities in that road segment and to the authorities.

Another scenario is warning the driver of a vehicle when approaching a school area where children are walking or bicycling nearby, which can be detected from children's mobile phone location or roadside sensor data and even aggregated with current time. Additionally, static data for a road segment for improving situation awareness can be requested from a service in the Internet. In Finland, this kind of static road data is available from a public Internet service called DigiRoad [20]. The data includes: road name, entry and exit points, direction, speed limits, number of lanes, traffic signs, location of pedestrian crossings, bus and taxi stops, elements such as illumination, service elements such as car parks and gas stations, and finally even information about scenic locations for tourists.

All this information combined together provides local real-time situation of the traffic in that road segment. Mobile phones with integrated GPS receivers in vehicles without any sensors can provide real-time information to the system, such as the location, speed and direction of the vehicle. This helps detecting the real-time congestion in road segments, which can be used in the system to guide drivers through alternative routes. Eventually the authorities are provided with a real-time situation of traffic in the whole system level. This can be used for real-time traffic monitoring, planning road infrastructure development and in a smaller scale, for planning future heavy transport scenarios.

Although we identified several scenarios, we do not target the data processing middleware for specific sensors or applications. Thus, services offered by the middleware cannot be tailored and optimized as suggested in [22]. Furthermore, isolated data sources and heterogeneous communication networks set challenges for the systems [22].

A. General Requirements

Ducourthial [23] describes an architecture for vehicular networks and lists unique issues not currently experienced within other networks. These networks are highly dynamic, unreliable, asynchronous, penalized with low bandwidth availability and short communication duration. In spite of this, vehicular networks require robustness, high quality of service and real time operation. Khanafer et al. [9] suggest a number of general requirements for intelligent traffic systems. Fault tolerance and real-time communications are essential to maintain the quality of service. As node deployment is not fixed and network topology can change, a scalable communication network is required. Cost and power consumption of a single sensor node should be minimized. System security should be guaranteed as the data is important real-time data, such as real-time traffic situation.

Furthermore, we adapt several general requirements for sensor networks from [1][8][10][24] to the scope of cooperative traffic data processing middleware: 1) the architecture should be data-centric as we will handle large amounts of heterogeneous sensor data, 2) asynchronous communication should be used because of communication outages, 3) component-based architecture and modular components follow from the diversity of the applications, 4) data processing middleware should have means for self-configuration to achieve runtime scalability with additional modules and to react to dynamic changes in network, 5) means for self-maintenance are required in the case of sensor node failures, 6) support for unknown types of future sensors is required, 7) data processing components should be easily deployable to the system and 8) the system should be able to reconfigure functionality and launch components based on data requests from the client applications. Furthermore, from the project goals we derive the following requirements: 9) component chaining is required to implement a modular system with data fusion capabilities, 10) output interfaces should be well defined and extendable to enable the usage of the resulting information in a wide variety of client applications utilizing common ontology and 11) real-time performance is a major issue when considering the usage of information in traffic.

The middleware running in the system should facilitate rapid application development and deployment as the system is developed constantly [25]. The service access points are provided by the middleware. It should also have means to manage system policies and security and privacy functions. To have a consistent quality of service in applications, the same data should be available to all applications. Thus standardized access to data and storage services need to be provided in the middleware.

B. System Description

The purpose of this prototype system is to demonstrate the chosen applications and test the feasibility of the system. Secondarily, the system functions also as a sensor network test bench for data processing algorithms under development. Currently, we have three main components in the prototype system: 1) mobile phones as mobile sensor nodes with the data collection software and integrated sensors, 2) the middleware for data processing in a remote platform in a public network and 3) a client application showing a map in a web browser on end-user workstation. The system prototype features two-way communications as users can utilize sensory data from other users to assist in decision-making, thus, we have both informative and coordinative cooperation [7].

The developed middleware architecture is data-centric and using a centralized database, as we want all the data to be rapidly propagated for the needs of any client application or data processing component. Our aim is to collect as much as possible raw data in our system. Specifically, we do not want to limit the future use of the collected data sets by applying application specific preprocessing as the level of detail is highest in the lowest level and preserving raw data enables the largest set of applications. We do not consider the level of detail of information and data shared in the system in this work. The data processing functionality can indirectly reduce the reliability and accuracy of the data as the potential loss of details occurs in all processing stages. These are important considerations for the data fusion as well [5]. We do not limit the data acquisition methods available, thus streaming, polling and event-based acquisition all can be used (req. 6). Sensor data producers are decoupled from consumers, because the commonly available sensor nodes in the system, such as mobile phones, should not be concerned about the data processing needs of the client applications. The remote data processing platform provides global endpoints for sensor data producers and infrastructure services to enter data into the system and applications to retrieve it. Global endpoints also allow heterogeneous communication methods, regardless of the physical location of the sensor nodes or the applications (req. 2). Based on these features, when considering architecture categories described by [5], our middleware architecture falls to the multi-tiered category of wireless sensor networks. Multi-tiered architectures avoid the problems with sensor network routing, maintain the quality of service better and also support data fusion in higher levels of processing, and when processing requires more powerful computation platforms [6].

We selected Global Sensor Network (GSN) [26], a sensor network middleware, as a middleware platform for our system [1]. As we are looking for complete open source implementation, for example partially released CarTel [27] cannot be considered. Lightweight Java implementation, possibility to use only a subset of modules and minimal configuration needs make GSN deployable to a large set of system configurations. Simple API and the number of already implemented features reduce the amount of required

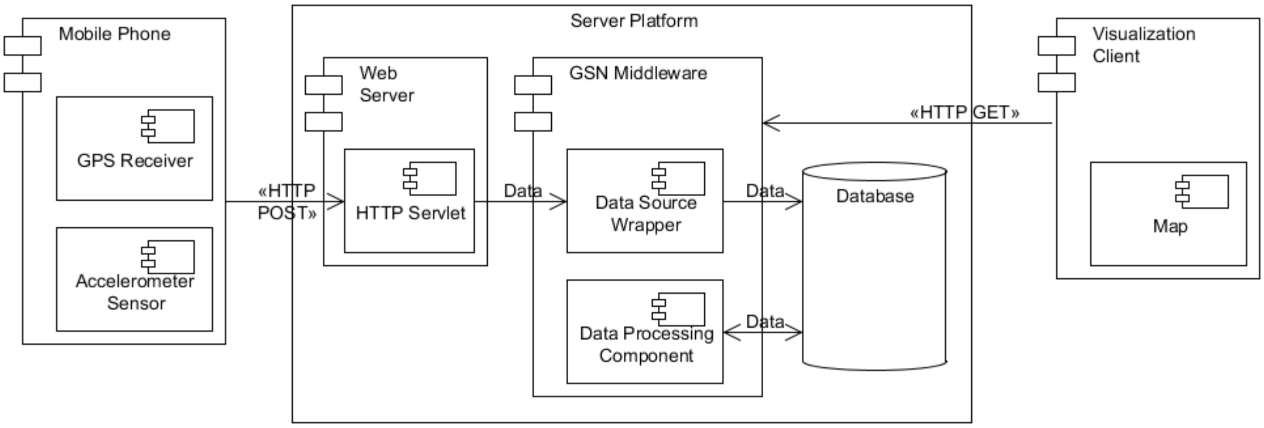


Figure 1. System architecture.

implementation work. The dynamic deployment of sensor nodes within the GSN is handled just by adding component configuration files to the system [26] (req. 7). The basic configuration defines the input/output data streams for a component. Component chaining and data fusion are enabled through defining the data input and output streams for the components (req. 9). The runtime reconfiguration of system functionality can be done by modifying the configuration file, and then GSN will dynamically start the required Java objects in the system (reqs. 4 and 8). The objects are alive only as long as required, and there is a built-in fault tolerance system for components (req. 5). Also, the dynamic use of components will not interfere with on-going data processing as the data streams are shared [26]. Addition of new data types in the system introduces the data schema evolution problem in the current components. This can be solved in GSN, as it offers means for filtering out unwanted data items for the input data stream of a component (req. 6). In case several components request the same data items, overlapping data queries are internally handled in the GSN middleware. The output data in all phases of processing can be saved to a database (req. 1).

C. Middleware components

System architecture is shown in Figure 1. Mobile phones with integrated sensors, serving as sensor nodes, collect data and disseminate it further for data processing to the server platform. In the server platform, we utilize HTTP servlet in a web server and as a sink node receiving data from the sensor nodes as HTTP POST requests. This servlet is responsible for delivering data elements to corresponding components (data source wrappers) in the middleware. For each session, the corresponding data stream is recognized from the session ID and by session we mean all the data produced by individual sensor node. The raw unprocessed data is also stored to a database for future use. In GSN, the data are streamed through MySQL database views from component to component at all steps of processing, allowing even intermediate processing results to be utilized immediately in simultaneously running algorithms and rapidly propagated to client applications (reqs. 1 and 3).

For the required application-specific data processing and data fusion, we have developed template components and component configurations for the use of data processing algorithms. The thread hosting these components will be alive as long as the session is alive and receive all the data in the session. In addition to data processing, these components can also act as interfaces to external, infrastructure or other, services in the network, for example to a public real-time traffic flow service or real-time local weather data. Template components subscribe to their required data source streams and publish single data stream as a result. These components can be deployed to the system any time by introducing new configuration files, thus automatically launching new data processing components. Another way of dynamically starting components in the system is by using the GSN's built-in web interface methods for requesting data from components. This fulfills the requirements 3, 4, 6, 7 and 8. This web interface can also be used to retrieve historical data from the database. The data processing algorithms and their parameters are defined in the template configuration files and can currently be: Java objects, binary executables or external services in the network. The subscribed data source streams are also defined in the configuration file.

For the data fusion of heterogeneous sensor data, we implemented a system to stream each type of sensor data in its own stream. All data stream elements include session ID and timestamp. This also solves the data schema evolution problem, at least partially, since all data and sources are separated from each other and required item types can be filtered from the stream. Applications can subscribe freely multiple types of sensor data for any fusion algorithms without additional overhead. Components can mix data from the different levels of data fusion and a give feedback to the system from higher levels, providing even learning capability to the system [5]. Additionally, data from external services in the network, location as GPS coordinates and user defined filters provided by GSN's built-in component interface can be utilized in data fusion. This way we can limit the data

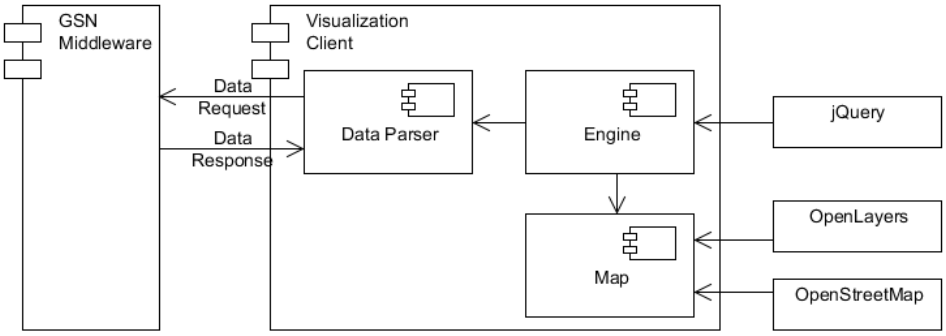


Figure 2. Visualization application architecture.

processing, data fusion or data queries, for example to certain geographical area, to specific time or to certain types and values of collected sensor data. Thus our system introduces freely configurable sensor data fusion for data processing components at every fusion level. This applies to raw and refined as well as to data from external services.

To visualize the data processing results from the middleware, we developed a web application displaying data on a map. In addition to GPS navigation and track logging systems, we have the ability to query and display properties of multiple clients dynamically with travel path history. The map shows the current location and the path history of the sensor nodes and their travel mode. Also detected anomalies on the road are displayed with a warning sign. The application polls the middleware for requested data, using GSN's built-in web interface over HTTP. Data is returned in XML based documents in the simplified format of GPS Exchange Format (GPX). TABLE 1 presents an example of this format. The first waypoint, described by element <wpt>, gives the current location and the element <rte> describes the path history as a series of waypoints. The visualization application architecture is presented in Figure 2 and Figure 3 shows example outputs of the visualization application on a map.

The application runs completely on the client-side, although relaying on up to date libraries from the Internet. Implementation of communication and operations are done with JavaScript and Ajax (jQuery) technologies in engine component. The data parser component parses the GPX data for the annotated map. Maps are based on OpenStreetMap and the dynamic map content is displayed with OpenLayers library. These libraries are loaded from their home sites at the start.

D. *Example application*

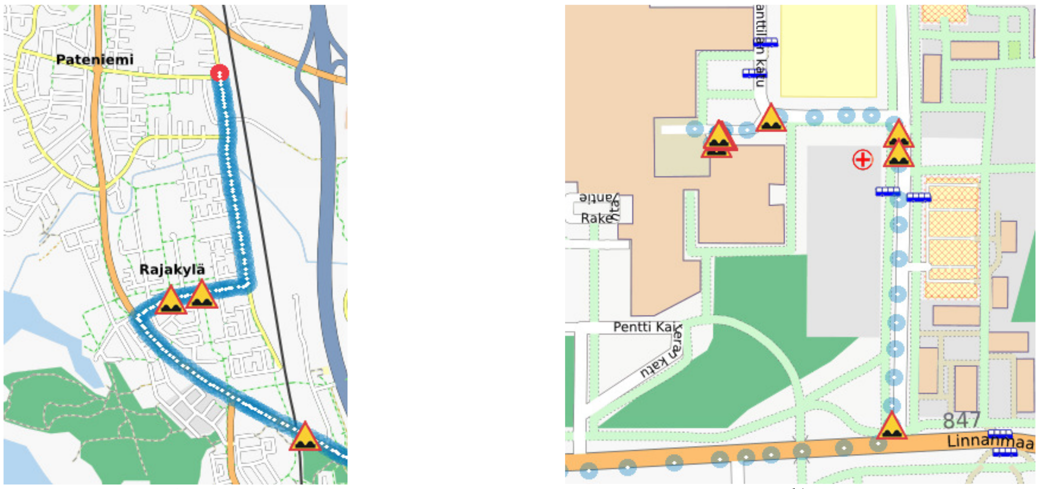
To demonstrate the system capabilities, integration of external data processing algorithms and heterogeneous data fusion in the components, we implemented an example application for recognition of the user's travelling mode and detection of anomalies of the road surface.

First, in the mobile phone sensor nodes, data are collected in the frequency of 38 hertz from the built-in accelerometer sensor and in the frequency of 1 hertz from the built-in GPS receiver. Data are stored temporarily on the phone memory and sent at given intervals as HTTP POST requests over the available communication network (for example GPRS or WLAN) to the remote platform end-point handler component, the data source wrapper. The current implementation of the collection software is written in Python for the Nokia N95 and in C for the N900 mobile phones.

In the middleware, we implemented components for map-matching for GPS location data, travel mode detection and road surface anomaly detection for the accelerometer data. Detailed descriptions of the data processing algorithms used in the applications can be found in [28][29][30]. First, the lowest level of data fusion occurs when accelerometer data and GPS location data are fused based on timestamps. The travel mode is detected from this data with timestamps and known locations and labeled accordingly, which constitutes an example of feature level fusion. Currently we can recognize several travel modes: stationary, walking,

TABLE 1. GPX DOCUMENT EXAMPLE

```
<gpx>
    <version>1.0</version>
    <creator>SDFA</creator>
    <wpt>
        <lat>65.059446</lat>
        <lon>25.472444</lon>
        <ele>9.5</ele>
        <type>Driving</type>
    </wpt>
    <rte>
        <name>Oulu</name>
        <desc>Testing</desc>
        <rtept>
          <wpt>
              <lat>65.059338</lat>
              <lon>25.473037</lon>
              <ele>10.1</ele>
              <type>Driving</type>
          </wpt>
          <wpt>
              <lat>65.059338</lat>
              <lon>25.473037</lon>
              <ele>9.8</ele>
              <type>Driving</type>
          </wpt>
        </rtept>
    </rte>
</gpx>
```



a) b)

Figure 3. Visualization client example outputs: (a) a red dot showing the current location, blue dots showing path history and warning signs showing detected anomalies, (b) additional information such as bus stops and a health center (from OpenStreetMap) shown in a close-up.

jogging, bicycling and driving a vehicle. Next, the map-matching component receives the fused and labeled data from the previous step and matches the location to the nearest road segment. For the road segment data, we use static local OpenStreetMap database downloaded from the Internet and this can be considered being an infrastructure service in the system. As a result, location coordinates for the travel mode are updated accordingly and are published for further use in the system. This can be considered a second example of feature level data fusion, refining sensor data with infrastructure service. Next, the accelerometer data and travel modes labeled as "driving" with the corrected GPS location are received by the road surface anomaly detection component. It performs anomaly detection and publishes the data. When refined data are shown in the map in a web browser, it constitutes an example of decision level data fusion supporting the user's decision making in the example application. The data flow between components is shown in Figure 4 from the perspective of data fusion.

IV. RESULTS AND DISCUSSION

Considering the requirements for cooperative traffic and wireless sensor network system implementation, we have demonstrated data-centric multi-tiered system architecture able to accomplish data fusion in various levels of data processing of sensor data to support decision making for the end-user. The example application demonstrates data fusion

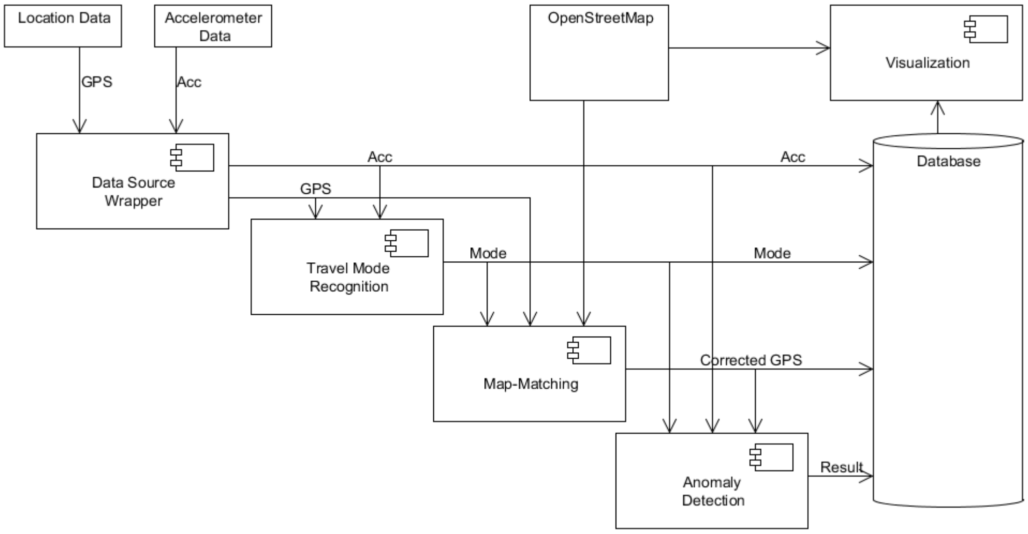


Figure 4. Data flow in the example application.

in the application domain, but the system allows also objective domain and the three-level hierarchies of data fusion. Informative and coordinative cooperation are demonstrated in the system. Collected raw and refined sensor data can be freely utilized by the system components, thus all information is available for information sharing. Applications can configure in components the required level of detail of the requested data. The system is also able to dynamically reconfigure functionality allowing runtime deployment of physical sensors and data processing components into the middleware. Furthermore, the system uses a mobile phone network and hence has support for asynchronous communication. Moreover, the system supports new types of future sensors and applications by providing public interfaces for data sources, for clients and into infrastructure services. The implemented example applications prove the system's ability to run multiple data processing components simultaneously, its ability for easy configuration and implementing component chaining for data processing. Scalability remains an open question until the applications are deployed for large-scale user tests in real-world situations, but this issue can be addressed with multiple server platforms all running the same middleware and perhaps even the same applications. These platforms can be easily interconnected using GSN's built-in features.

Compared to the OSGi platform [16][17], we consider both infrastructure services and individual vehicles being in the lowest layer. The upper layers constitute the data fusion capabilities and applications, which can be run in parallel. Concerning the Content Delivery Network [15] and the scenario given by [18], our system is different from the mobile phone scenario because of the routing is done in the mobile network and managed by the mobile network operator. Currently, we do not have a mechanism for clients to locate local services in the geographical area. The vehicle route prediction capability to reach the next node is not needed in the mobile phone network. In comparison to existing systems [10][12][13][14][27], we offer in addition runtime pluggable data processing components and heterogeneous data fusion capability, which are launched based on client application configuration and data requests. Google Maps for Mobile offer cooperative traffic applications, which are based on location data solely. CarTel, TJam and Nericell locally process data on the nodes. TJam also uses migratory services in nodes, which is a feature we might consider in the future. The Mobile Millennium uses roadside sensors and historical data, both of these features we would like to have in the future. In [19], a similar middleware is described with kernel, data module and decision module components. However, we do not utilize a general system queue and the middleware itself takes care of concurrent access to the data. The ubiquitous intelligent transportation system [21] also resembles our system. In their resource layer, they handle the preprocessing and classification of data, which is different from our approach as we have the possibility to access raw data in all layers in the system. Furthermore, we do not distribute functionality to separate layers for data processing, services or applications as data streams can be made available to all subscribers. For multiple communication technologies we utilize global endpoint handler components, which map to the ubiquitous network layer.

Figure 5. Test system set-up in vehicle. Mobile phone installed in car dock in the middle.

We conducted a small scale field testing for the system in real environment by walking and driving a vehicle in the city of Oulu, Finland, simultaneously collecting data by mobile phones with integrated accelerometer sensor and GPS receiver. Test system set-up in a vehicle can be seen in Figure 5. The test route consisted of 3 kilometers in a suburb area, 1 kilometer in a park and 1 kilometer in city center. Data was transmitted using HTTP over GPRS to the sink node in the intervals of 10, 30 and 60 seconds. See TABLE 2 for the results. Data transmission failed in 5 out of 99 testing runs (5.1%). There is no significant difference in failures concerning different data transmission intervals. However, an interesting detail was that all the transmissions failed around the same location in the city. For real-time applications, shorter intervals are feasible, but this depends on the requirements of the data processing algorithms. Another consideration for real-time applications is the speed of the vehicles, which contributes to the quality of collected data [15]. We also estimated the bandwidth required per node to be around three kilobytes per second for all the intervals, as the required message length for sending accelerometer data with GPS data is about 100 bytes and the accelerometer sensor runs in the mobile phone in 38Hz. In the future, we might have other sensors integrated into the mobile phone serving as a sensor node and we need to have estimation of the required bandwidth for the data amounts. When utilizing 3G technologies the available bandwidth can be significantly increased. As expected, during the testing, receiving of the GPS signal was sometimes disrupted in cities, 17.3% out of the total testing time, to be exact. However, this depends also on the GPS receiver hardware. The battery lifetime is always limited when using mobile

TABLE 2. FIELD TESTING RESULTS

Testing results	**Measurement intervals**		
	10s	***30s***	***60s***
Tests run	23	41	35
Failed transmissions	1	2	2

phones transmitting data continuously, but as our phones were often mounted in a vehicle, its power system was used. Cost of a single sensor node is not a considerable issue if the user's personal mobile phones can be utilized. What is more, the cost of air time and available bandwidth also varies locally.

Sometimes, the travel mode recognition did not work accordingly and we needed to abandon its results and publish all data to the anomaly detection component. The two mobile phone models had different accelerometer sensors, the Nokia N900 accelerometer being much more accurate than Nokia N95 accelerometer, so we needed to parameterize recognition components for each phone model. This means that online learning should be utilized to enable adapting the component for different physical sensors.

Also, we tested the feasibility of GSN middleware in a normal desktop PC (2.40 gigahertz processor with one gigabytes of RAM running Windows XP) with randomly generated data sets. We used 50 data sources simultaneously streaming 50 kilobytes of payload to the system at every 20 milliseconds. The delivery time from a data source wrapper to a data processing component was less than one millisecond per payload. This is very a promising result considering real-time capabilities of the system (req. 11). However, these tests have less real-world value as the used hardware components, available network bandwidth and programming skills of the developers largely contribute to the real-time performance in the prototype.

In this work, we have not considered security and privacy issues, which are unavoidable when collecting, fusing and visualizing data from multiple clients and with the integration of roadside infrastructure data and sensors. Koenders et al. [16] give a list of success factors for intelligent transportation systems. There is a need for communication partners in the system to work and penetration of similarly equipped vehicles is critical. This is where vehicle manufacturers need to co-operate by utilizing standards. Authorities can stimulate this, however clear benefits are needed as investments are required. For the individual users, personal privacy and reliability of the services needs to be guaranteed. Security and privacy concerns in the system can be addressed, for example by introducing user groups with restricted access to the data and results. In our prototype, the reliability is largely dependent on mobile phone network reliability and does not require additional investments from the users or traffic operators.

V. CONCLUSION AND FUTURE WORK

We have started with a generic implementation of sensor network middleware for cooperative traffic applications, which meet the given requirements, drawn from the literature and from the project goals. Considering the goals, we have demonstrated modular component-based prototype sensor network implementation capable of sharing traffic related information collected by mobile sensor nodes and realizing the described usage scenarios. In the next phase, the developed prototype system will be deployed to be used with multiple sensor nodes and client applications. Increased scalability needs may require introducing multiple middleware platforms for distributing the system load. It would be possible to introduce remote interconnected application-specific platforms in the infrastructure providing data fusion featuring separate data sources in remote platforms.

An important goal of our future work is to implement a common ontology and a specific data fusion model in the middleware. This includes information on vehicle type, location, route, event, time, road condition and possible future application-specific parameters. Also, we will develop interfaces to sensors in instrumented vehicles and introduce usage of an on-board diagnostics module installed in test vehicles. These open many new possibilities as more detailed data of the vehicle behavior can be utilized. Usage of GPS data would allow us to feature also descriptive cooperation, such as the user's intentions or direction of movement, in the system [7]. Furthermore, interfaces to public Finnish government infrastructure services available in the Internet, such as DigiTraffic [31], a real-time traffic flow information service using roadside sensors, and DigiRoad, the static database containing road data and elements, allow us to widen the system perspective to the whole traffic system level. This information can be easily fused with additional local real-time data in the system and provide feedback to road users and operators.

ACKNOWLEDGMENT

The authors would like to thank all the personnel in the Sensor Data Fusion and Applications project. This work was funded by the National Technology Agency of Finland and was carried out in Sensor Data Fusion and Applications project as a part of the Cooperative Traffic research program of the Strategic Centre for Science, Technology and Innovation in the Field of ICT.

REFERENCES

[1] T. Leppanen, M. Perttunen, J. Riekki, and P. Kaipio, "Sensor Network Architecture for Cooperative Traffic Applications," in Proceedings of International Conference on Wireless and Mobile Communications, pp. 400-403, September 20-25, Valencia, Spain, 2010.

[2] F. Minarini, "eSafety for Road Transport: Investing in Preventive Safety and Co-operative Systems, the EU Approach," Advanced Microsystems for Automotive Applications, pp. 487-492, 2005.

[3] Cooperative Vehicle-Inftrastructure Systems. URL: http://www.cvisproject.org/ 12.07.2011.

[4] N. Faouzi, H. Leung, and A. Kurian, "Data fusion in intelligent transportation systems: Progress and challenges - A survey," Information Fusion, Vol. 12, 1, pp. 4, 2011.

[5] B. Dasarathy, "Sensor fusion potential exploitation-innovative architectures and illustrative applications," in Proceedings of the IEEE, Vol. 85, 1, pp. 24, 1997.

[6] C. Haythornthwaite, "Crowds and Communities: Light and Heavyweight Models of Peer Production," in Proceeedings of Hawaii International Conference on System Sciences, pp. 1-10, 2009.

[7] K. Lidstrom, T. Larsson, and L. Stranden, " Safety Considerations for Cooperating Vehicles using Wireless Communication," in

Proceedings of 5th IEEE International Conference on Industrial Informatics, Vol. 2, pp. 995, June 23-27, 2007.

[8] V. Dyo, "Middleware design for integration of sensor network and mobile devices," in Proceedings of the 2nd international doctoral symposium on Middleware, pp. 1-5, New York, NY, USA, 2007.

[9] M. Khanafer, M. Guennoun, and H. Mouftah, "WSN Architectures for Intelligent Transportation Systems," in Proceedings of 3rd International Conference on New Technologies, Mobility and Security, pp. 1, December 20-23, 2009.

[10] P. Mohan, V. Padmanabhan, and R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in Proceedings of the 6th ACM conference on Embedded network sensor systems, pp. 323-336, New York, NY, USA, 2008.

[11] The Finnish Cooperative Traffic Programme. URL: http://www.cooperativetraffic.fi/ 12.07.2011.

[12] The Mobile Millenium Project. URL: http://traffic.berkeley.edu/, 12.07.2011.

[13] Google Maps for Mobile. URL: http://www.google.com/mobile/products/maps.html, 12.07.2011.

[14] O. Riva, T. Nadeem, C. Borcea, and L. Iftode, "Context-Aware Migratory Services in Ad Hoc Networks," IEEE Transactions on Mobile Computing, Vol. 6, 12, pp. 1313-1328, 2007.

[15] M. Elshenawy, M. El-Darieby, and B. Abdulhai, "Scalable and location-aware its content management in vehicular environments," in proceedings of IEEE Intelligent Vehicles Symposium, pp. 627, June, 2010.

[16] E. Koenders and J. Vreeswijk, "Cooperative infrastructure," in Proceedings of IEEE Intelligent Vehicles Symposium, pp. 721, June, 2008.

[17] O. Brickley, S. Chong, M. Klepal, A. Tabatabaei, and D. Pesch, "A Data Dissemination Strategy for Cooperative Vehicular Systems," in Proceedings of 65th IEEE Vehicular Technology Conference, pp. 2501, April, 2007.

[18] D. Tacconi, D. Miorandi, J. Carreras, F. Chiti, and R. Fantacci, "Using wireless sensor networks to support intelligent transportation systems," Ad Hoc Networks, Vol. 8, 5, pp. 462, 2010.

[19] S. Bengochea, A. Talamona, and M. Parent, " A software framework for vehicle-infrastructure cooperative applications," in Proceeedings of IEEE Intelligent Transportation Systems, pp. 797, September, 2005.

[20] Digiroad – A National Road and Street Database. URL: http://www.digiroad.fi/en_GB/, 12.07.2011.

[21] W. Lv, B. Du, D. Ma, T. Zhu, and C. Wang, "Applied research of data sensing and service to ubiquitous intelligent transportation system," Frontiers of Computer Science in China, Vol. 4, 3, pp. 417-426, 2010.

[22] Y. Yu, B. Krishnamachari, and V. Prasanna, "Issues in Designing Middleware for Wireless Sensor Networks," IEEE Network, Vol. 18, pp. 15-21, 2003.

[23] B. Ducourthial, "About efficiency in wireless communication frameworks on vehicular networks," in Proceedings of The First International Workshop on Wireless Networking for Intelligent Transportation Systems, pp. 3:1-3:9, New York, NY, USA, 2007.

[24] K. Modukuri, S. Hariri, N. Chalfoun, and M. Yousif, "Autonomous Middleware Framework for Sensor Networks," Journal of Pervasive Computing and Communications, Vol. 1, pp. 337-345, 2008.

[25] T. Kosch, I. Kulp, M. Bechler, M. Strassberger, B. Weyl, and R. Lasowski," Communication architecture for cooperative systems in Europe", IEEE Communications Magazine, Vol. 47, 5, pp. 116, May, 2009.

[26] K. Aberer, M. Hauswirth, and A. Salehi, "A middleware for fast and flexible sensor network deployment," in Proceedings of the 32nd international conference on Very large data bases, Seoul, Korea, pp. 1199-1202, 2006.

[27] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H Balakrishnan, and S. Madden, "CarTel: a distributed mobile sensor computing system," in Proceedings of the 4th international conference on Embedded networked sensor systems, pp. 125-138, New York, NY, USA, 2006.

[28] O. Mazhelis, "Using recursive Bayesian estimation for matching GPS measurements to imperfect road network data," in Proceedings of 13th International IEEE Conference on Intelligent Transportation Systems, pp. 1492, September, 2010.

[29] J. Kantola, M. Perttunen, T. Leppanen, J. Collin, and J. Riekki, "Context Awareness for GPS-Enabled Phones," in Proceedings of ION Technical Meeting, Manassas, VA, USA, 2010.

[30] M. Perttunen, O. Mazhelis, F. Cong, M. Kauppila, T. Leppänen, J. Kantola, J. Collin, S. Pirttikangas, J. Haverinen, T. Ristaniemi, and J. Riekki, "Distributed Road Surface Condition Monitoring Using Mobile Phones," in the 8th International Conference on Ubiquitous Intelligence and Computing, Banff, Canada, September 2-4, 2011.

[31] Digitraffic. URL: http://www.infotripla.fi/digitraffic/english/, 12.07.2011.