Road traffic monitoring with location-aware sound sensors

Bachelor-Thesis von Dimitar Goshev
29.06.2012

TECHNISCHE UNIVERSITÄT DARMSTADT

Fachbereich Informatik
Telecooperation
Knowledge Engineering

Road traffic monitoring with location-aware sound sensors

Vorgelegte Bachelor-Thesis von Dimitar Goshev

Prüfer: Prof. Dr. Max Mühlhäuser
Betreuer: Immanuel Schweizer, Frederik Janssen

Tag der Einreichung:

Bitte zitieren Sie dieses Dokument als:
URN: urn:nbn:de:tuda-tuprints-@URN-ID
URL: http://tuprints.ulb.tu-darmstadt.de/@tuprints-ID

Dieses Dokument wird bereitgestellt von tuprints,
E-Publishing-Service der TU Darmstadt
http://tuprints.ulb.tu-darmstadt.de
tuprints@ulb.tu-darmstadt.de

Erklärung zur Bachelor-Thesis

Hiermit versichere ich, die vorliegende Bachelor-Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 29.06.2012

(D.Goshev)

Contents

Zusammenfassung

Die Verbreitung von Smartphones steigt rasant an. Gleichzeitig nimmt die Anzahl der Sensoren in Smartphones immer weiter zu, so dass Smartphones immer häufiger als mobile Sensorknoten verwendet werden. Einige Informationen wie z. B. Verkehrsdichte können mit deren Einsatz jedoch nicht direkt gemessen werden. Zurzeit werden für diese Zwecke Kameras oder Induktionsschleifen angewendet. Diese Methoden sind für die Beschaffung umfassender räumlicher Datenmengen zu teuer.

Im Rahmen dieser Thesis wird deshalb die Korrelation zwischen Verkehr und Lärm ausgenutzt um einen virtuellen Sensor aufzubauen. Er nutzt Klassifzierungsmodelle, die bis zu sieben Stufen für die Beurteilung der Verkehrsdichtewerten aufweisen. Für die Erlernung der Modelle nutzen wir Daten aus OpenStreetMap, Wetterdaten und Daten aus da_sense, der Sensorplattform der Telekooperation. Besonders im Fokus waren dabei die Verkehrdaten der Stadt Darmstadt und die Lärmpegel, die wir mit Noisemap, einer Android Sensorapp, gemessen haben.

Die ersten Ergebnisse auf einem kleinen Datenset zeigen die Machbarkeit und liefern gute Resultate. Virtuelle Sensorik wird sich so zu einer günstigen und genauen Alternative zu traditionellen Verkehrmessungen entwickeln.

Abstract

Smartphone penetration is increasing rapidly, as is the amount of sensors deployed in a smartphone. This has lead to a surge in mobile sensor apps to cheaply collect high amounts of sensor data.

But some data cannot be measured directly by smartphones, e.g., traffic activity. Instead traffic activity, today, is measured by using cameras or inductive loops integrated in streets and traffic lights. These sensors are expensive to deploy in large numbers.

In this thesis we build a virtual sensor by exploiting the correlation between traffic and noise pollution. By using the sound pressure data provided by the Noisemap Android app, as well as weather data and road data from OpenStreetMap and combining this with traffic density measurements in Darmstadt, we are able to generate classification models for up to 7 traffic density levels.

In our preliminary evaluation with a small number of measurement samples, the developed method shows promising results. Compared to existing solutions for road traffic monitoring, this technique provides a low-cost, yet accurate alternative.

1 Introduction

The increasing availability of smartphone devices in recent years have provided millions of people around the globe with location-aware data collection instruments. These devices provide capabilities for capturing sound, image, location and other data, enabling participatory sensing in the surrounding environment [6]. It is estimated that by the end of 2011 627 million people worldwide used a smartphone device, accounting for 12.3 percent of the total number of active cellular devices. By 2015, the smartphone user base is expected to be over 1.5 billion [18]. As a consequence various initiatives have emerged, implementing new architectures, with the goal of enabling volunteers to create free sources of location-aware sensor data.

One such initiative is da_sense, by Telelecooperation Lab (TK) at the Department of Computer Science at Technische Universität Darmstadt. The project has collected, as of 2012, more than 100 000 geotagged data values for sound pressure level with its infrastructure, which includes a mobile sound measuring tool for smartphones. Furthermore traffic data from the city of Darmstadt is also collected in the da_sense infrastructure. Another initiative, utilizing data from location-aware devices, is OpenStreetMap(OSM) - a project that creates and provides the largest free road network dataset. As of 2012 more than half million volunteers have contributed to the project, uploading almost 3 billion GPS data points [1].

Although smartphones and other sensor devices provide the ability to observe various phenomena, others are not subject of direct observation. Nevertheless in numerous cases combining and/or processing data from available physical sensors, could provide indirect measurements of related abstract conditions, which are by themselves not measurable. One such instance is road traffic activity, which can not be measured directly by the sensors, that smartphones are commonly equipped with.

The majority of the current solutions for road traffic monitoring are expensive and rely on dedicated sensors deployed on roads, such as inductive loop vehicle detectors, traffic cameras and doppler radar. In many cases the deployment of the hardware presents a significant effort. One instance are the magnetic field detectors, used for collecting the traffic data, provided by da_sense, which are installed under the road surface. However traffic activity has high correlation to sound pressure level [2], which can be measured via microphone [17]. One software that makes use of this possibility is Noisemap - an application, for devices running on the Android operation system, which gathers data on loudness and transfers it to the da_sense platform.

In this work a virtual sensor was developed, which uses the geotagged equivalent continuous sound level(L_{eq}) captured by Noisemap to generate information about traffic activity. The focus of this thesis is on the required preprocessing steps. Before we are able to classify traffic activity levels we need to assign traffic, sound, as well as other data to road segments, for which the traffic is monitored. We, therefore, introduce new methods to automatically extract road segments from the road network data provided by OSM and define a measurement area for sound, surrounding these road segments.

The preprocessed L_{eq} data is used, together with information about the current weather conditions and various road characteristics data, such as surface type and number of lanes, to generate different datasets to train the supervised machine learning models. These models enable a classification of up to 7 traffic activity levels, determined after processing of the traffic data, and monitoring of the number of vehicles. A total of 7860 models were used to evaluate the accuracy of monitoring based on various factors, such as the duration of the sound pressure level measurements, the accuracy of the recording position and type of applied spatial filters. Five of the models are presented in detail in this thesis.

In contrast to traditional solutions, our approach requires no specialized hardware for recording of sound pressure level, as the measurements have been collected using the standard integrated microphone on various mass-market smartphone devices. Furthermore no installation effort is required, as measurements collected at various locations around the road are used and their positions are automatically factored into the model. To the best of our knowledge this is the first approach to use sound pressure levels in creating a virtual traffic density sensor. In our preliminary evaluation with a small number of measurement samples, the developed method shows promising results that it could operate successfully in the field and is able to generate low-cost, yet accurate traffic activity information.

This work is organized as follows. Related work is presented in Section 2. A detailed information about the data sources used in this work is presented in Section 3. An overview of the implementation is presented in Section 4.1. Section 4.2 describes how road segments and spatial filters were created. In Section 4.3 is presented the processing of the traffic data and the method for determining the traffic level. Section 5 describes the dataset generator used for creating training sets and five of the supervised learning models developed during this work. Evaluation of the results is presented in Section 6. Section 7 contain conclusion and discussion about future work.

2 Related work

There has been much work on systems for traffic monitoring, both in the academic and the commercial sector. Though very little research is available in using sound from location-aware devices with omnidirectional microphone for monitoring of the road traffic.

The majority of research for identifying the traffic conditions with location-aware devices, is based on location traces collected from a GPS receiver from on-road vehicles, that track the movement of the vehicles [21]. There exists research for monitoring of the traffic though cellular devices by leveraging mobile phones carried by users, using information gathered at cellular towers [20]. Though the latest provide a low accuracy.

There has been research that describes a method, which uses the microphone of location-aware device for monitoring of the traffic - Mohan et al. present in [12] and [13] a system for rich monitoring of road and traffic conditions using smartphones [13].

The devices used in this system might include any or all of a microphone, camera, GPS receiver, and accelerometer. Nevertheless the use of audio data recorded with the microphone of the device to monitor tor the traffic is relatively limited. A simple detector that performs a discrete Fourier transform on 100ms audio samples and looks for energy spikes in the frequency domain is used to detect honking by vehicles on the road. The honk detection has the purpose to identify noisy and chaotic traffic conditions like that at an unregulated intersection.

There has been research for traffic monitoring by noise with static sensors with omnidirectional microphones, too. However they describe methods for traffic monitoring using multiple omnidirectional microphones.

One such work [8] describes a system, for monitoring the vehicle count as well as their speed. Four identical microphone arrays are placed on a panel installed at a specific position near the road. Each array contains 12 miniature omnidirectional microphones, positioned in a specially designed pattern. A DSP unit is used to analyze and process the recorded signal.

The authors describe that trial demonstrated that the system performs well in free flow-traffic, though at its current form, has difficulties to perform such detection in congested start-stop traffic. No performance statistics are provided for both cases. The authors conclude that further research is required to extract the above mentioned traffic indicators automatically with this system.

Similar approach with two microphone arrays is presented in [11]. As in [8] the authors conclude that further research is required to make the developed system accurate.

3 Data sources

The training sets during this work were build using freely available data collected by crowd-sourcing. Sound pressure level records and information about the road traffic activity in Darmstadt were obtained from the da_sense project by Technische Universität Darmstadt. Road network data was acquired from the OpenStreetMap project.

The developed software for building the models for monitoring of the traffic level, provides abilities to include historical weather data from external weather services. However, since no such free services were available at the time of this work, weather data was collected separately during the period of recording of the sound pressure level measurements. An overview of the type of data, used during this project, its sources and how it was processed is presented in Figure 1.

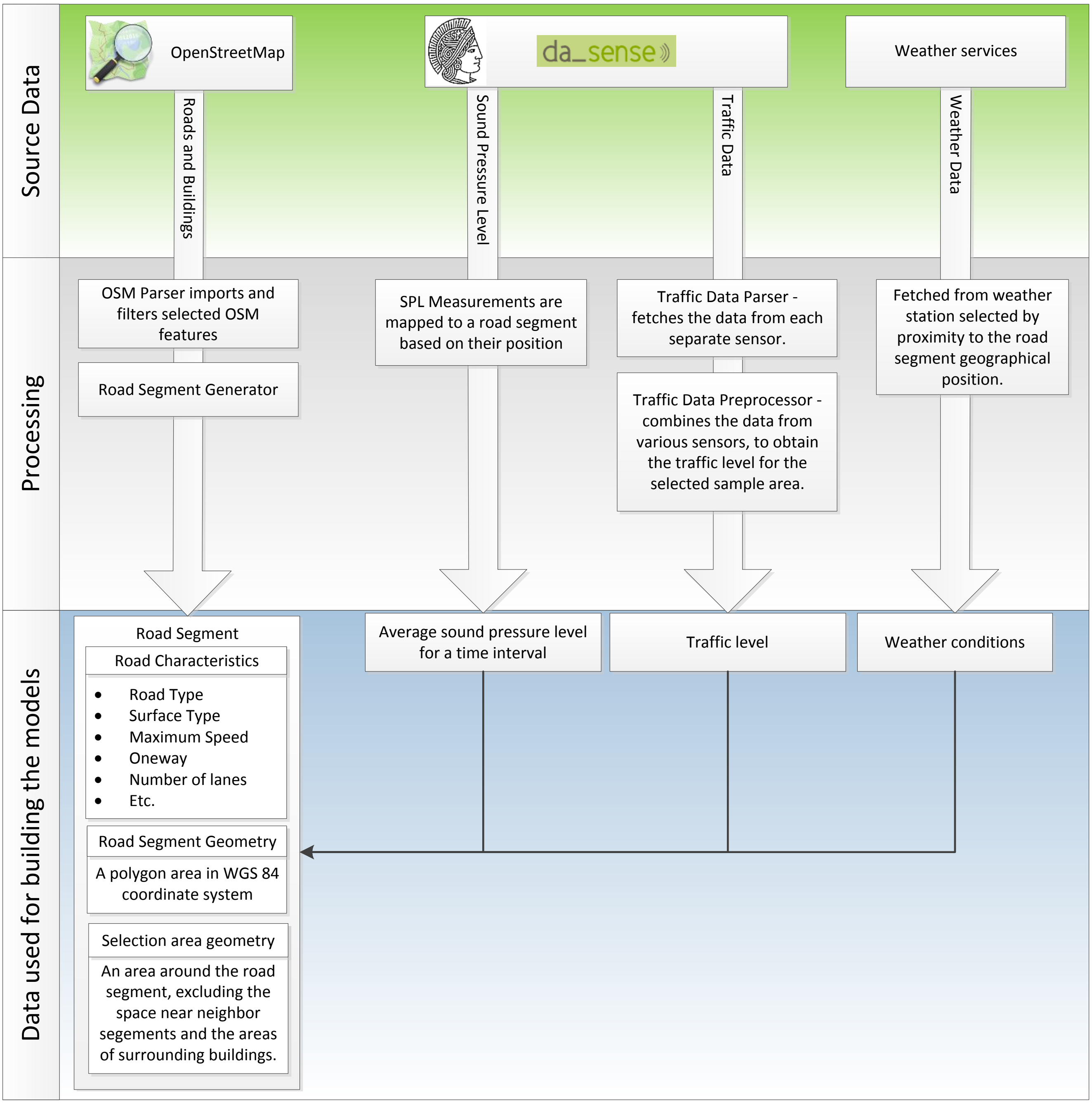


Figure 1: Overview of the source data used in this project.

3.1 Road network data

OpenStreetMap(OSM) was selected as source for road network data in this project, as it provides the largest free road network dataset. Studies [14] show that more than 1,630,000 km of the street network in Germany has been covered as of 2011.

OSM has emerged as the leading initiative to create an online editable open source map of the Earth. It follows a peer production model similar to Wikipedia, with the notable difference that it allows only registered users to edit the map. To date more than half million volunteers have contributed to the project, uploading almost 3 billion GPS data points [1]. The vector map data is available under Creative Commons Licence(CC-BY-SA).

Using OSM data for the development of the model for monitoring allows its application for all parts of the road network covered by OSM.

OSM vector data consists of three spatial types of features - place, linear and area [Figure 2]. Each feature is assigned one or more attributes, also referred as *tags*, consisting of *key=value* pairs.

The tagging schema builds a complex taxonomy, which is developing constantly with new feature classes and attributes added, after voting by the community. The community approved taxonomy is not mandatory, since OSM doesn't impose any limitation on the type or number of tags, which could be attached to a feature.

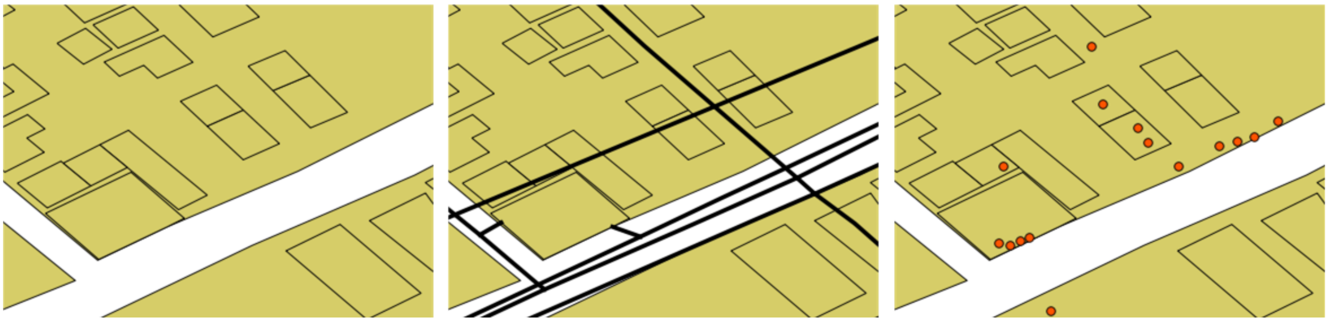


Figure 2: OpenStreetMap raw data types. Left: Area features. Center: Area and linear features. Right: Area and place features.

3.2 Sound pressure measurements

The sound pressure level data used for this project is an extract from the freely available da_sense data for the period between Mon, 22 May 2012 00:00 CEST to Thu, 14 Jun 2012 22:15:00 CEST.

da_sense is an initiative by Technische Universität Darmstadt, implementing a sensor network in the area of Darmstadt, Germany to monitor various environmental phenomena such as carbon oxide, carbon dioxide and others. As part of the project, volunteers are recording sound pressure levels, using location-aware cellular devices.

Measurements of the equivalent continuous sound level(L_{eq}) are collected with the Android application Noisemap, developed by the project. For each measurement value, additionally the position of the device during the recording is stored as a WGS84 geographical coordinate, together with the accuracy of this position.

The data was obtained directly from the MySQL database used for storage of the recorded measurements. The project provides additionally a REST API for data access. However due to SQL's querying capabilities, an an access through JDBC driver for MySQL proved to be the more suitable solution for this project.

Due to issues with the software component for geotagging the measurements, all instances of a series of measurements, are recorded at the same location. Two positions of the location are recorded - one obtained by the GPS receiver of the device with accuracies of less than 5 and less than 10 meters, and another one, which is manually determined and has accuracy of less than a meter.

3.3 Traffic data

The da_sense project makes available information about the traffic activity at 171 crossings in the area of Darmstadt. The activity is recorded by magnetic field detectors placed under the road surface. Each traffic sensor supplies information only for vehicles that drove over the road lane, under which the sensor is located.

The data is collected and sent in intervals of 15 minutes and made accessible as pairs of CSV files for 2 regions of the city. The detectors record two statistics: the number of vehicles that pass over one detector, during the interval and a utilization factor - the time in seconds that the detector was triggered. We are not aware of the accuracy of the provided statistics.

Information from 64 detectors for the period between Mon, 22 May 2012 00:00 CEST to Thu, 14 Jun 2012 22:15:00 CEST was used during this project, in order to determine the traffic level at 17 sections of the road network.

3.4 Weather data

As no freely available historical weather data was available during the project, the weather conditions, for each of the used in the project series of sound pressure measurements, were recorded. This includes only a boolean indicator, if it rained during the series or not. The developed models could gain accuracy, if detailed weather information, such as height of precipitation and wind speed, is included.

4 Implementation

4.1 Overview

Noise is defined as unwanted sound transmitted through air or another medium. Road transport is one of the major sources of noise in an urban environment [3], together with air transport, rail transport and sites of industrial activity. Road traffic noise is the aggregation of the noise from the individual vehicles in a traffic flow. The noise is non-steady, heavily fluctuating and its volume is dependent on various factors, such as:

- the traffic volume.
- the average speed of the traffic, impacting the amount of noise emitted by the engine and air buffeting.
- the composition of the traffic, as the noise emitted by different types of vehicles varies immensely.
- the road gradient, as bigger slope of the road impacts the driving and therefore the noise emitted by the engine.
- the road surface, as the absorption of different surfaces varies, thus altering the noise emitted by rolling tires.
- the driving condition, with free-flowing traffic, causing less acceleration and deceleration and consequently impacts the noise emitted by the engine.
- the current weather conditions.

The main sources of noise generated by a moving vehicle are primarily the engine, transmission system and accessories, road excitation, and air buffeting. Though the contribution of air buffeting in urban environment is significantly lower due to lower speeds. The noise is produced by (1) direct noise emission from various individual sources in the vehicle, such as engine, transmission, exhaust, cooling fan, tires, etc. and (2) vibrations of the surfaces of the vehicle, which are caused either by direct transmission of engine and road-excited vibratory forces, or by airborne noise from the engine surfaces [16].

During this work, we investigated the feasibility to monitor the traffic activity based on noise emitted by the road vehicles. A number of supervised learning models were developed, that were trained to monitor the traffic activity, based on input, including among others, average sound pressure level for various intervals of time. The models were trained to specifically monitor the traffic activity of parts of the road, referred in this work as *road segments*.

The term *road segment* is applied to a uniform section of the road, usable by motor vehicles, that must comply to the following restrictions:

- The section does not intersect any other road sections and is not connected to other roads at any points, but the end ones. In other words the set of road segments represents a planar network graph, where each road segment is an edge of this graph.
- Along the section's length is allowed either exclusively uni-directional(one-way) traffic or bi-directional traffic, but not both.
- Along the section's length, the road has the same characteristics. Bellow are listed the ones, which were of interest for this work:
  - road type classification - one of *motorway*, *trunk*, *primary*, *secondary*, *tertiary*, *residential*, *living street* or *unclassified*
  - maximum permitted speed
  - name of the street
  - number of lanes - this excludes road links, which connect one road section, of one the above listed road types, to another section, of the same or different road type
  - type of the road surface

The set of road segments was automatically generated by processing the described in 3.1 OpenStreetMap data for the area of Darmstadt, Germany. For each segment in the investigated area, three types of geometries were generated:

- a polyline geometry, obtained after preprocessing of the source vector map data.
- a polygon geometry, portraying the two dimensional area, that the road segment covers.

Figure 3: In the plan plan above, the road segment geometry is displayed in black, the selection area in yellow and the geometry of the buildings in white. In this specific case, the selection area covers a space with distance of 10 meters around the road and excludes the area in proximity of 30 meters to neighbor road segments, as well as the area of the surrounding buildings. The recording positions of the non-selected measurements are displayed in blue, and those of the selected measurements in white. In this example only measurements within the selection area are included for building the training examples.

- a set of polygon geometries representing areas around the segment, used as a spatial filters for the sound pressure level measurements, using their recording position. For convenience those areas are referred further simply as *selection areas*. The geometries of the selection areas are created, based not only on the geometry of the respective road segment, but also on that of the neighbor road segments and the surrounding buildings.

For 17 of the road segments in the area of Darmstadt, the da_sense traffic data contains information from sensors that cover all road lanes. These 17 road segments are referred bellow as *sample areas*. The data describes the traffic activity at the sample areas in intervals of 15 minutes. Two different statistics could be obtained from the available data:

- the number of vehicles that drove above the detectors.
- an utilization factor, which represents the duration, in seconds, for which a vehicle was over one of the detectors.

The sound pressure level measurements were retrieved from the da_sense data source and linked to a sample area, based on their recording position. This was achieved by using the above described selection areas as a spatial filter. A more detailed information about this process is described in Section 4.2.3.

The measurements are combined, based on the time of recording, into groups in time intervals of 1, 3, 5, and 15 minutes. Each group was used to calculate the averaged sound pressure value, from measurements recorded in the interval. A separate group of training sets are generated for each case. This was performed in order to evaluate, what accuracy of monitoring could be achieved for various intervals.

From the da_sense traffic data a traffic level is determined and used as a supervisory signal for the training examples. Training sets with 3, 4, 5, 6, and 7 different traffic levels were generated, once based on the number of vehicles per lane and and once on the utilization factor per lane. Two more groups of training sets were created, for which the traffic level was set as a real value, representing the actual number of vehicles per lanes and the utilization factor per lane .As only traffic information for intervals of 15 minutes is currently available, the determined traffic level for this 15 minute interval is used as a supervisory signal for training sets containing samples with measurement data for shorter intervals. Though the software allows an interval with the same duration as the one of used for averaged sound pressure level measurements, to be selected, when data for shorter intervals is made available.

A number of supervised learning classifiers were used to analyze each set of training examples. The models were then evaluated, to determine the feasibility and optionally the accuracy of monitoring of the road traffic with the data type in the specific set of training examples. An overview of the generated training examples is presented in Figure 4.

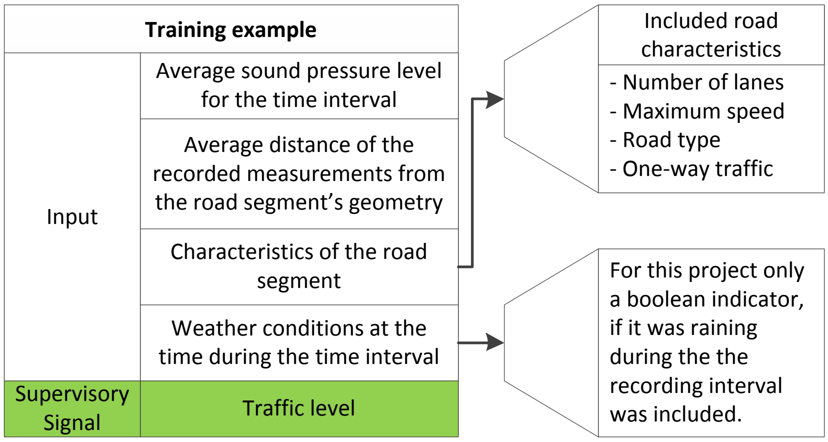


Figure 4: Each of the generated training sets contains examples as described above. The training sets differ in the interval for which recorder measurements are selected, the area used to select measurement in proximity to the road segment and the way the discrete traffic level is determined.

4.2 Road network data processing

A model build with manually created geometries of the road segments and the areas, which will be used as a spatial filter for the sound pressure level measurements, would be virtually unusable, as such data is not available from any public or commercial sources. Therefore one of the goals of this work was to automate the process of generation of the road segments.

The OSM data contains only the polyline the geometry of the roads. However for this project the two dimensional geometry of the road and the geometry of areas around them, used as a spatial filters for the sound pressure level measurement were also generated, based on the polyline geometry and the meta data of the road.

This was performed in order to maximize the estimated accuracy of the distance between the measurement's recording position and the road, and to filter the measurements that would provide the most useful information about the traffic activity. Both proved to be vital statistics for the accuracy of the model.

Three main software components were used to generate the road segments and there selection areas: a parser for the source vector data and a road segment generator developed in Java, and a PostgreSQL database with a PostGIS extension, which adds support for spatially indexed storage of geographic objects and various functions for spatial querying of those objects. An overview of the implementation is presented in Figure 5.

PostGIS was used, not only for storage of the geographic objects, but also for some of the spatial modifications of the polygon geometries.

The vector map data used for this project is a snapshot of the freely available OpenStreetMap(OSM) data for the area of Darmstadt, Germany (49°85'74" N, 8°62'57' 'E ; 9°88'56" N, 8°66'55" E).

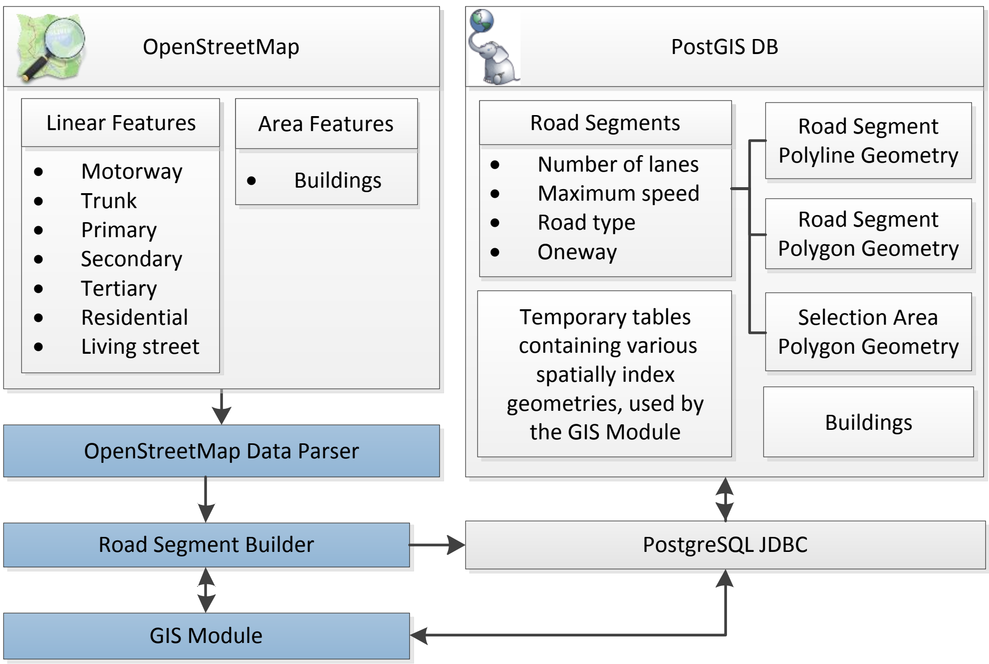


Figure 5: An overview of the software modules, developed to generate the road segments and their respective geometries.

4.2.1 Import and preprocessing of the OpenStreetMap vector data

The OSM data is accessible in OSM format - an XML serializable format. All geographical entities are represented as *nodes*, containing the coordinate in WSG 84 reference coordinate system. Linear features are represented as a a list of nodes, known as *way*. Area features are represented as a closed *way*. Additional attributes of the features are represented as *tags*, consisting of *key*=*value* pairs.

Along with the nodes and ways the OSM data model contains one more base type - *relation*. It consist of ordered list of nodes, ways or other relations and similar to other types, a set of attributes. Relations are used to model logical relationships between features. For example a bus route and a set of buildings belonging to a given university campus are represented as relations.

The first step while working with the OSM dataset was to develop a software for parsing the OSM data format, filtering of different OSM features based on their type, attributes and geographical properties.

OSM dataset includes various types of linear features, which describe ways not intended for automotive vehicles, such as railways, pedestrian paths, etc. A short overview of the available linear features is presented in Figure 6.

Therefore the goal of the developed software was to enable the extraction of only the data, which would be of interest for the constructing of the road segments. These include types of vehicle roads such as motorways, primary roads, secondary roads, and others. The full list of imported road types is available in Section 4.2.2.

The developed software module for importing and preprocessing of the OSM dataset was as a second step extended with capabilities for converting the geometries of the ways to Open Geospatial Consortium (OGC) Well-Known Binary(WKB) and Well-Known Text(WKT) formats to enable import into the PostGIS database. This was achieved by integrating the open-source library Java Topology Suite, which is available under GNU Lesser General Public License (LGPL).

An UML model of the developed in Java module for import of the OSM dataset is presented in Figure 7.

Roads, where automotive vehicles are permited.	Paths and other pedestrian areas.
Bicycle lanes.	Tram lines.
Interpolation of house numbers along the respective street.	Various types of barriers and fences.
Rail lines.	Service roads such as parking aisles.

Figure 6: Some of the types of linear features, which could be obtained through attribute filtering.

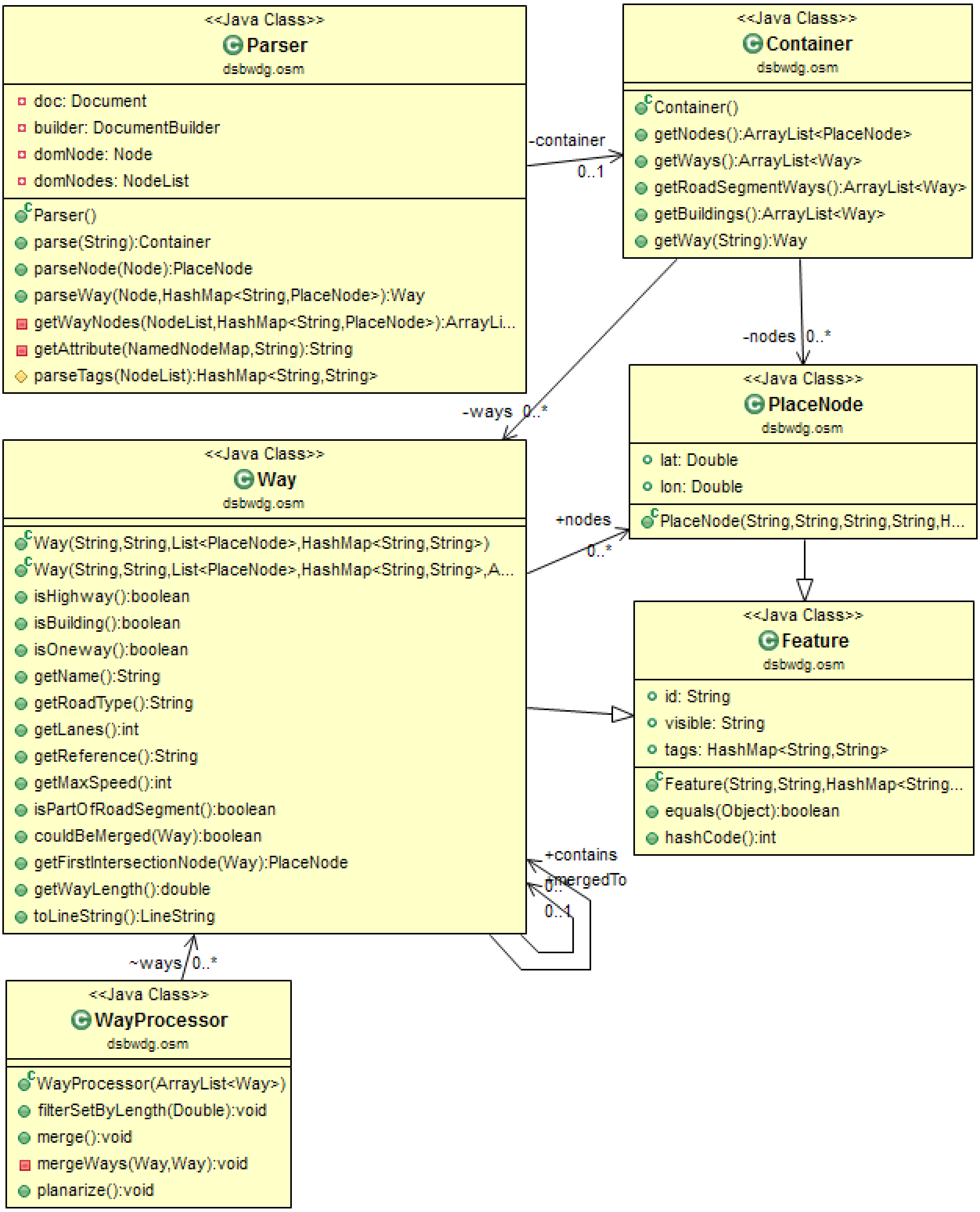


Figure 7: UML schema of the OSM data parser module.

4.2.2 Generation of road segments

Using the described above parser, initially, imported were the linear features, which represent the following road types - *motorway*, *trunk*, *primary*, *secondary*, *tertiary*, *residential*, and *living street* and *unclassified* as well as the different types of road links - *motorway link*, *primary link*, *secondary link* and *tertiary link*.

Immediately after the described above features were imported, one issue was easily noticeable. The OSM ways are describing geographically parts of the roads, which have distinct attributes such as maximum permitted speed, street name, number of lanes, etc. However they are crossing other ways, i.e. the raw OSM linear features build a non-planar network topology [Figure 8]. It should be noted that crossing ways do share a common node and the Quality Assurance tools applied by OSM state to verify that intersecting ways share such common node. This means that the graph of ways is build on top of a planar network graph unless there are errors in the data, which was the case with the data in the region used in the project.

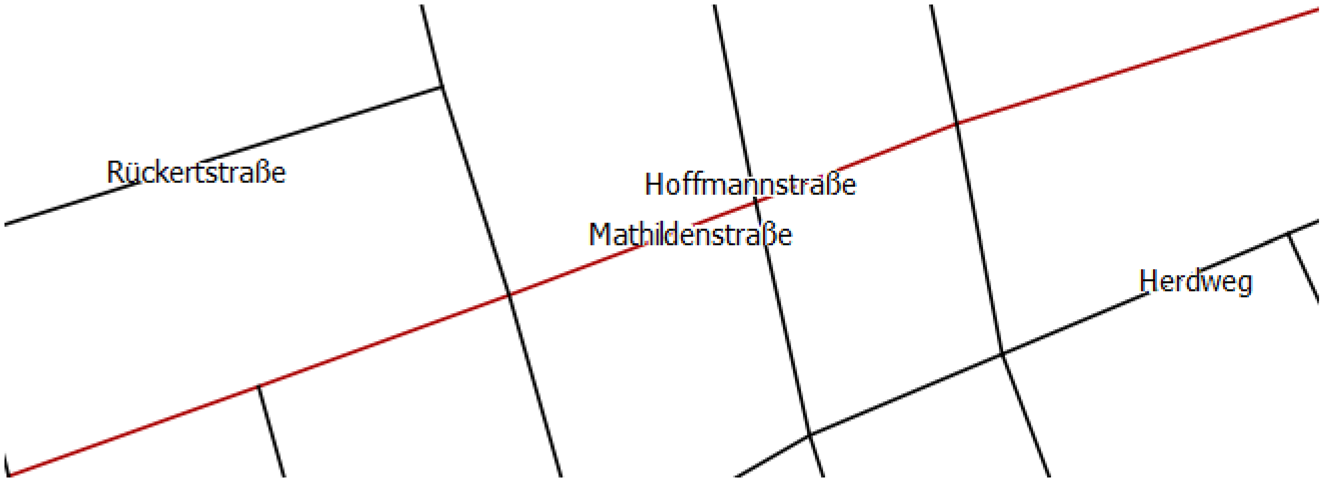


Figure 8: The raw OSM data describe non-planar network topology. E.g. A way describing Mathildenstraße, Darmstadt crossing other ways.

A straight-forward planarization of the network topology didn't yield the desired results. The area of the road segments, as defined in Section 5.1, consisted of multiple separate polylines, which share no set of spatial or non-spatial characteristics, which could allow automated selection of all polylines in this segment.

One of the causes for this issue was that road links cause an undesired split of ways at the ends of the road segment. This was addressed by removing all types of *road link* features. The majority of them were filtered out by their respective attributes. Though as it turned out, a significant amount of the links in the dataset are misclassified [Figure 9]. Additional filtering for those misclassified ways had to be implemented to resolve the issue.

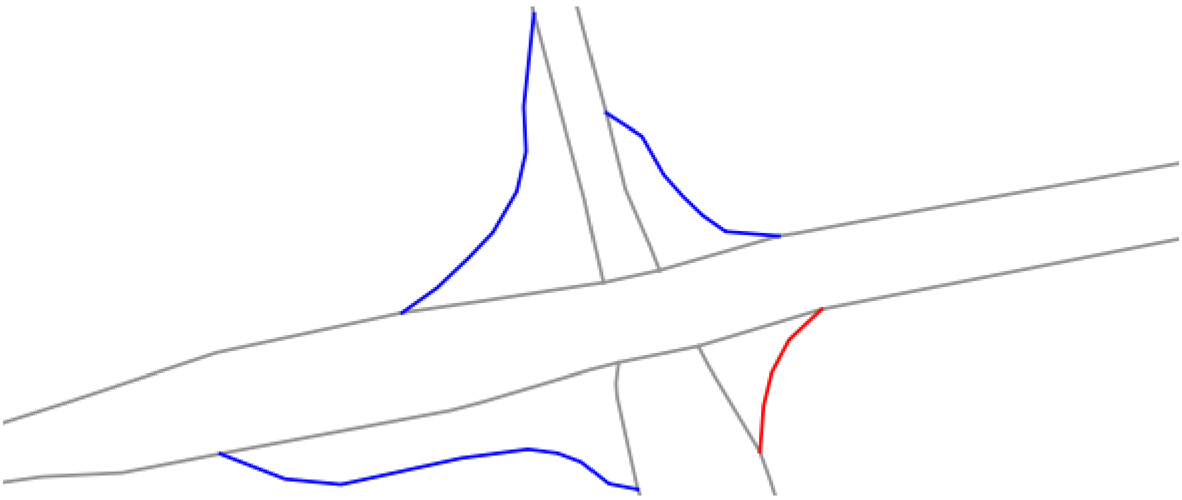


Figure 9: In red is a way misclassified as *primary* road at the crossing of Rheinstraße and Neckarstraße, Darmstadt. The blue ones are correctly classified as *primary link*.

An investigation showed that the greater part of those road links were classified as *unclassified* roads. No further investigation for the reasons causing of that phenomena was undertaken. Nevertheless a default value in one of the various OSM editors would explain that. The OSM *unclassified* highway attribute is used in fact to describe minor public roads typically at the lowest level of the interconnecting grid network. They were initially included in the filtering, since they are considered usable by motor vehicles. An investigation of the selected region, though showed that all *unclassified* roads were misclassified road links and *service* roads, such as parking and others, which were not of interest for this project. Therefore the *unclassified* roads were excluded.

The remaining misclassified links were removed by filtering them by a set of characteristic features that they posses - they have short length, they are one-way roads and they have only one lane. A successful removal was achieved by filtering of ways with length less than 45 meters in combination of the other attributes stated above.

Another reasons that directly planarizing the road network was causing undesired segmentation was that parts of the road, which could be represented as a single way are represented with multiple ways. In many such cases the ways are connected inside of a road segment [Figure 10]. After the splitting of intersecting ways, they resulted in two or more polylines inside of a road segment. This was addressed by merging such ways before the planarization of the network topology was performed.

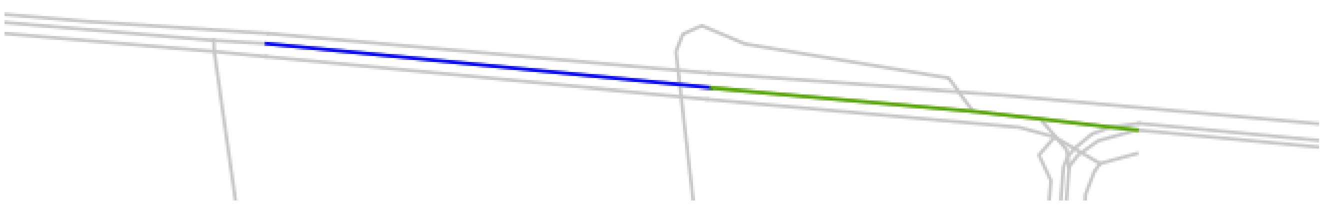


Figure 10: Two ways with same attributes joining inside of a road segment(Bismarkstraße, Darmstadt).

After this steps of preprocessing of the OSM ways, they were split at the nodes of intersection in order to create a planar network topology. In few cases such node did not exist. Instead two nodes within few centimeters from each other were present. This in fact contradicts to the OSM schema, and is a Therefore nodes within 5 centimeters from each other were merged into one, before the split operation was executed.

One last filtering step had to be performed before the road segments and their geometry was generated. The polylines defining road lanes inside of a crossing had to be removed as they would result in generation of road segments, which would be no compliant of the introduced definition. This was achieved through filtering of the polylines with a length of 30 meters or less [Figure 11].

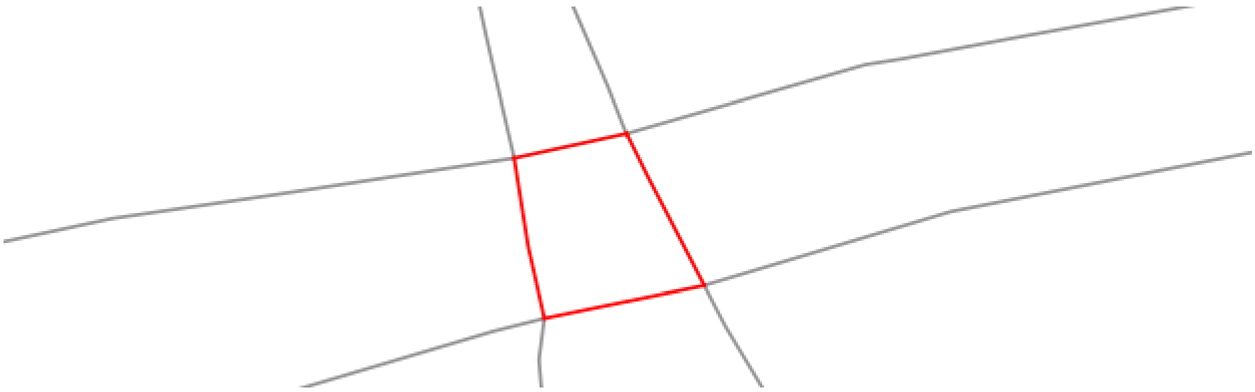


Figure 11: Splitting polylines to create a planar network topology, results in unwanted short polylines inside of crossings(E.g. Intersection of Rheinstraße and Neckarstraße, Darmstadt).

All attributes of the ways such as number of lanes , maximum speed and name of the street have been preserved during the execution of the operations described above.

At this stage the road segments as defined in Section 5.1 consisted of either 1) a single way or 2) two or more parallel ways. A method to identify and merge those parallel ways into a single geometry was applied as next step.

Spatial buffers with a radius of 20 meters were created around each way. For each of the buffers the geometries of the ways contained spatially were selected. As no geometries with such short length were present after the described above filtering operations, the created buffers did not contained spatially any non-parallel ways. The ways, which were mutually contained in the respective buffer of the other were grouped, creating a multi linestring geometry [Figure 12].

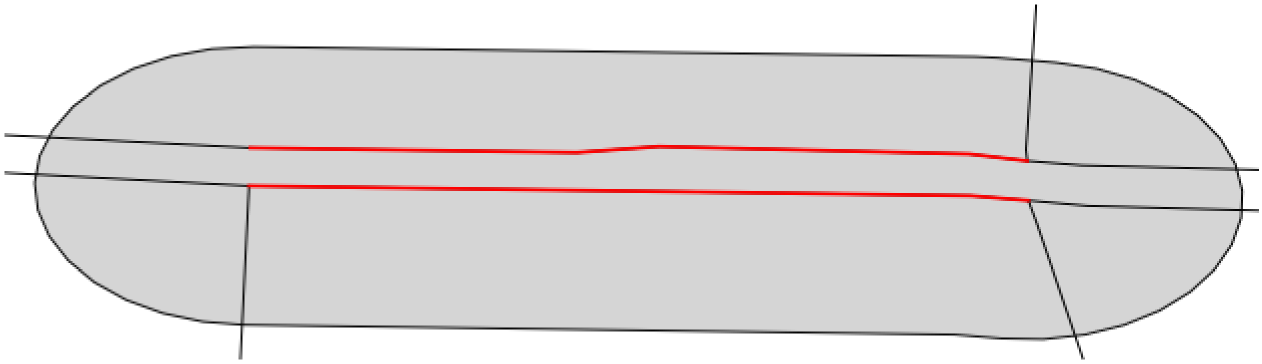


Figure 12: Parallel ways with a maximum distance between them of 30 meters are combined into a single segment.

The way represents a part of the road, which has one or more road lanes and it has a certain length and width. Though its geometry is only available as a polyline, i.e. no width attribute is available. Thus the first step was to create a polygon which represents the 2 dimensional geometry of this part of the road. The number of lanes were used to determine the width of the road. There is no certain width for each lane in the current regulations in Germany and vary between 2.75 and 3.5 meters. For the project 3.5 meters was used as default width as the field samples showed that this is the width of the greatest part of lanes in the examined region. The geometry of the way was used to create a spatial buffer around it with a flat end cap at the polyline ends. The perimeter of the buffer was set at an equal distance from the polyline geometries representing the way[Figure 13].

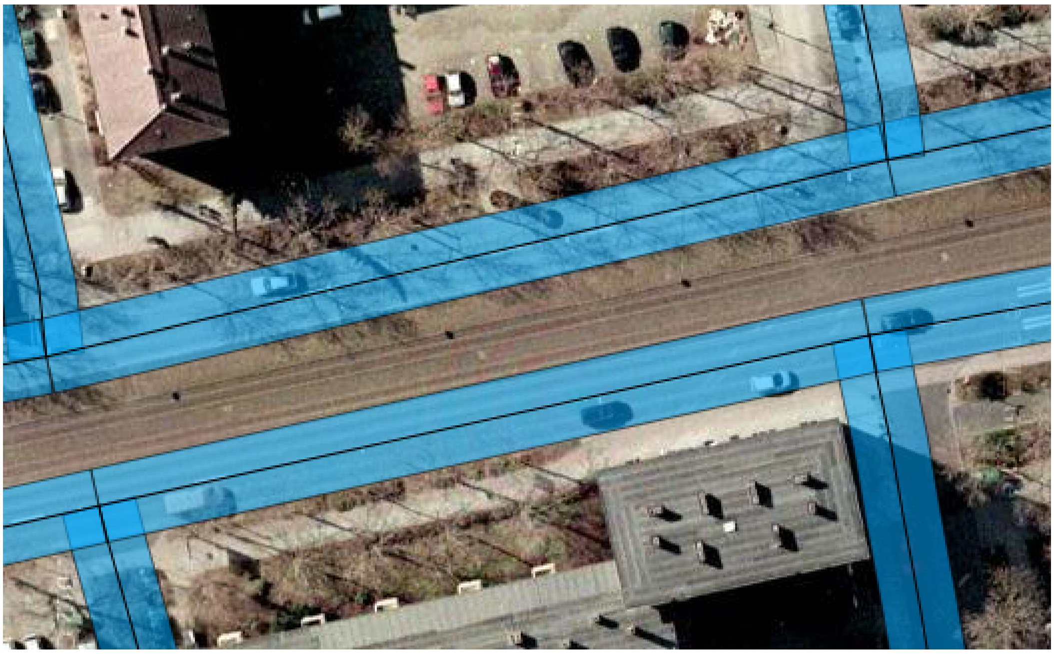


Figure 13: The resulting road segment geometry. The parallel ways in the middle represent a single road segment with a multi polygon geometry.

After the above described procedure the road segment geometry has been created. For the segments represented initially with a single way feature, this was a polygon geometry and for the road segments represented by two or more parallel ways, this was a multi polygon geometry represented the segment.

The possibility of merging each multi polygon geometry into a single polygon geometry was evaluated. The main goal for this was to eliminated small gaps, if such existed, between the parallel polygons, that represented an area, which is part of road. At first a convex hull was used to create the combined shape, that covers all of the contained polygons. However this proved to be significantly inaccurate for non-straight road segments, as a large area outside the road surface was covered by the convex hull as well.

Another approach that was evaluated was building a concave hull from the multi polygon geometries. In this case, too, the created polygon geometries resulted in a worse representation of the actual road geometry, due to the fact that in significant amount of cases the area between those parallel polygons did not belong to the road, as visible in Figure 13.

4.2.3 Generation of spatial filters

One of the crucial parts during developing the models for traffic monitoring, was to determine, which of the sound pressure records, to be selected as input data for the training examples.

A straight-forward approach that was used at first, was to select the measurements recorded at positions in a certain proximity from the road segment.The developed software allows to specify that distance, though 10 meters radius from the road segment was the used for all areas created and used in this project.

One immediately visible problem of limiting the measurements based only on distance from the segment, was that measurements recorded near crossings were selected for two or more segments. Some of the selected measurement were even recorded at the area of other segment and not the targeted one. Simply excluding the area of other road segments, was considered to be inefficient solution as it would only minimize the number of measurements used selected for two or more segments.

Another consideration with the above described approach was the sound pressure measurements, which are recorded at positions at the area of the target road segment or near it, that even though are not selected for neighbor segments are in close proximity to them. Such measurements could be heavily impacted by noise emitted from vehicles on those neighbor segments.

Therefore, in order to filter out measurements selected for two or more segments and to minimize the impact of the noise emitted by vehicles on neighbor segments, the area near neighbor segments was excluded from the selection areas. The developed software allows to specify the exact distance of the area, which needs to be excluded. Selection areas excluding the space in radius of 10, 20 and 30 meters from neighbor segments were created and used in the models.

In order to achieve the above described operation, two spatial buffers were created around each segment: A - with a radius equal to the target radius of the selection area from the road and B - with a radius equal to the distance of the area near neighbor segments, which needs to be excluded. The area of intersection of the buffer A of the target road segment with buffers $B_1 ... B_n$ of the neighbor road segments, was excluded from the area of buffer A. More simply this is defined as follows:

$$SelctionArea = A \setminus (B_1 \cup B_2 \cup .... \cup B_n)$$

This resulted in selection areas as the one presented in Figure 14.

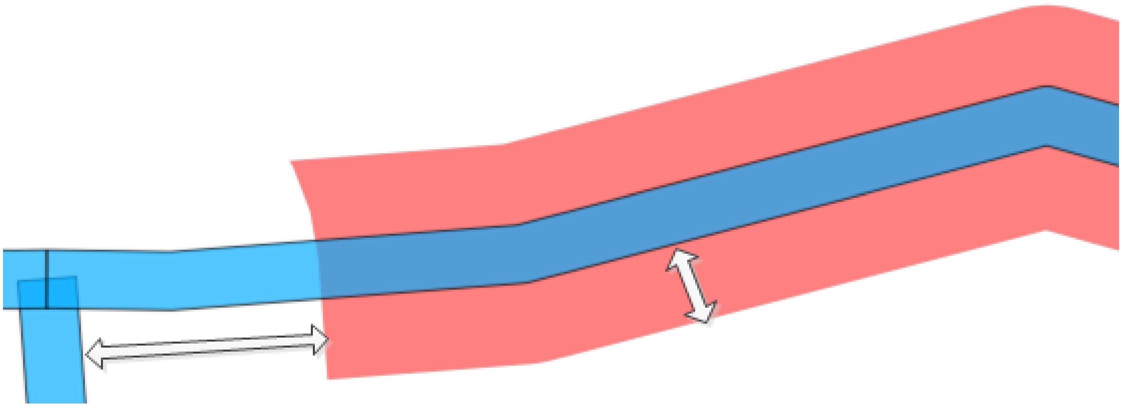


Figure 14: A selection area with a radius of 10 meters and a minimum distance of 30 meters from non-associated road segments.

The created geometry as described presented a good estimation of the area, where noise emitted by vehicles in the road segment, would be mostly perceptible, with one exception. In urban areas in near proximity to the roads there are various types of buildings and parts or all of their area was included in the created polygon geometry. To avoid the use of measurements recorded in closed space areas, the geometry of buildings was imported, too from the OSM dataset and their area was excluded from the geometry of the selection areas as well.

The described above operation resulted in the final version of the selection areas used created for this project. An example for a selection area is presented in Figure fig:RoadSelection.

Accuracy of the stored recording position of the sound pressure measurements must also be taken into consideration. The measurements used in this project contain two positions - the first one is obtained from the GPS receiver of the recording device. The positions have accuracies of up to 5 and up to 10 meters. Another one is manually determined an its accuracy is up to 1 meter. Based on the accuracy of their recording position, measurements, which are recorded outside of the selection area, could be falsely selected in the area, while others, which are recorded inside could be falsely filtered out.

To handle this issue, the selection area, considered as the ideal basis area, where the measurements are recorded, could be used as a basis for another spatial filter. An inward offsetting of the polygon geometry, would shrunk the area, to minimize the number of included measurements, that are recorded outside of it. An outward offsetting(buffering) of the geometry, would expand it, to minimize the number of filtered out measurements, that are in fact recorded inside of the area, but falsely filtered out due to the accuracy of their position.

Figure 15: An end result for an area(in red) for spatial filtering of sound pressure measurements for the associated road segment(in black). The intersection areas with the geometries of the surronding buildings(in white) are excluded.

The traffic level used for the developed models was obtained from the da_sense project. The project provides information for sensors located at crossings, regulated by traffic lights. At each crossing a number of magnet field detectors are located under the road lanes, each one providing information for the lane, under which it is placed. Such detectors are installed at a total of 171 crossings in the area of Darmstadt. On most of these places, detectors are not installed under all lanes.

A scheme with the locations of the installed detectors is available for each of the crossings. All schemes were thoroughly inspected to determine, if either there are detectors at the crossing, placed under all lanes of a road segment or if the traffic information could be obtained from detectors located at two neighbor crossings.

A total of 17 road segments were identified in that way. These segments are referred bellow as *sample areas*. For 11 of them, the traffic information was acquired from detectors located at only one crossing and for further 6, the traffic information was obtained from detectors located at two neighbor crossings. An example for the second case is presented in Figure 16.

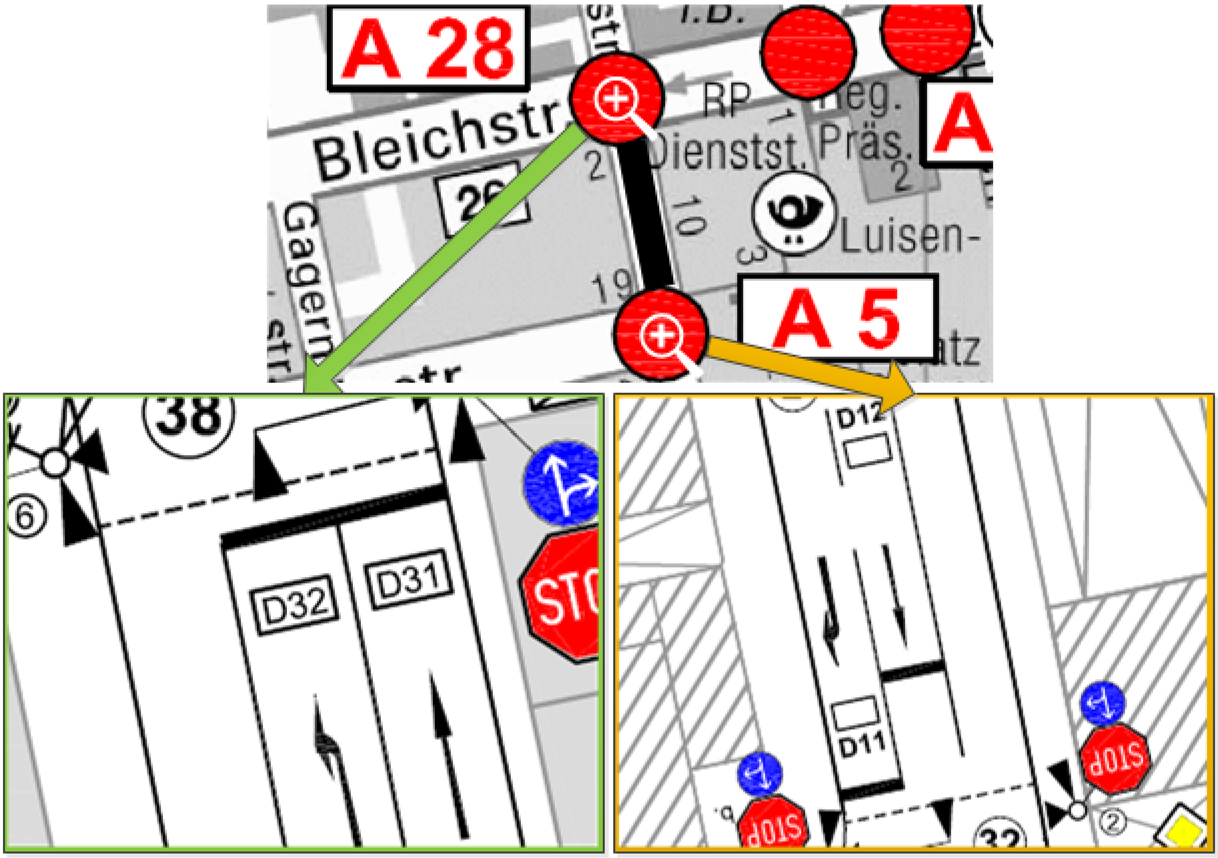


Figure 16: Detectors at the traffic lights A 5 and A 28 are used to obtain traffic data for a road segment between the two traffic lights at Grafenstr., Darmstadt.
The data from the sensors, labeled with D32 and D31, at traffic light A 28(left), is combined with the data from the sensors, labeled with D11 and D12, at traffic light A 5(right). Each sensor records the number of vehicles, that drove over it and the duration, for which a vehicle was over the sensor.

The traffic information for all sensors is made available by the da_sense project as a pair of CSV files, containing the records made by the sensors in a 15 minute interval. One of the files contains information for the north region of Darmstadt and another one - for the south region.

For each sensor, two string tokens are used as identifiers for the recorded data in the CSV files - one for the vehicle count and the other for the utilization factor. This tokens differ from the sensor labels, visible in Figure 16. The mapping between the sensor label and the tokens is specified in a separate translation table for each crossing.

To import the traffic data, a parser in Java was written, which reads the recorded values from the pair of files, combines the values from the target sensors and stores them into a PostgreSQL database. For fetching the CSV files, the open-source library OpenCSV was used, which is available under Apache 2.0 license. Imported was the recorded data for the period from Mon, 22 May 2012 00:00 CEST to Thu, 14 Jun 2012 22:15:00 CEST.

At 3 of the 17 road segments information from all or part of the sensors was missing for the whole mentioned above period. For other segments the recorded information was missing only for certain periods. Since in cases with missing information no special values were used, but simply zeros, the set of intervals for which the recorded values are missing, overlaps with the set of intervals for which the recorded values were in fact zeros.

The source traffic data was used to calculate a traffic level, which was used as a supervisory signal for the training examples in the developed classification models. Generated were groups of training sets with 3, 4, 5, 6 and 7 discrete traffic levels, once based on the number of vehicles and once based on the utilization factor.

The number of lanes of the road segment were also considered as factor in determining of the traffic level, as a level solely based on the above mentioned data is fairly inaccurate. The number of lanes is obtained after processing of the source OpenStreetMap data as described in Section 4.2.2.

For the one group of models the number of vehicles that drove through the road segment, was divided to the number of lanes of the respective road segment [Figure 17].

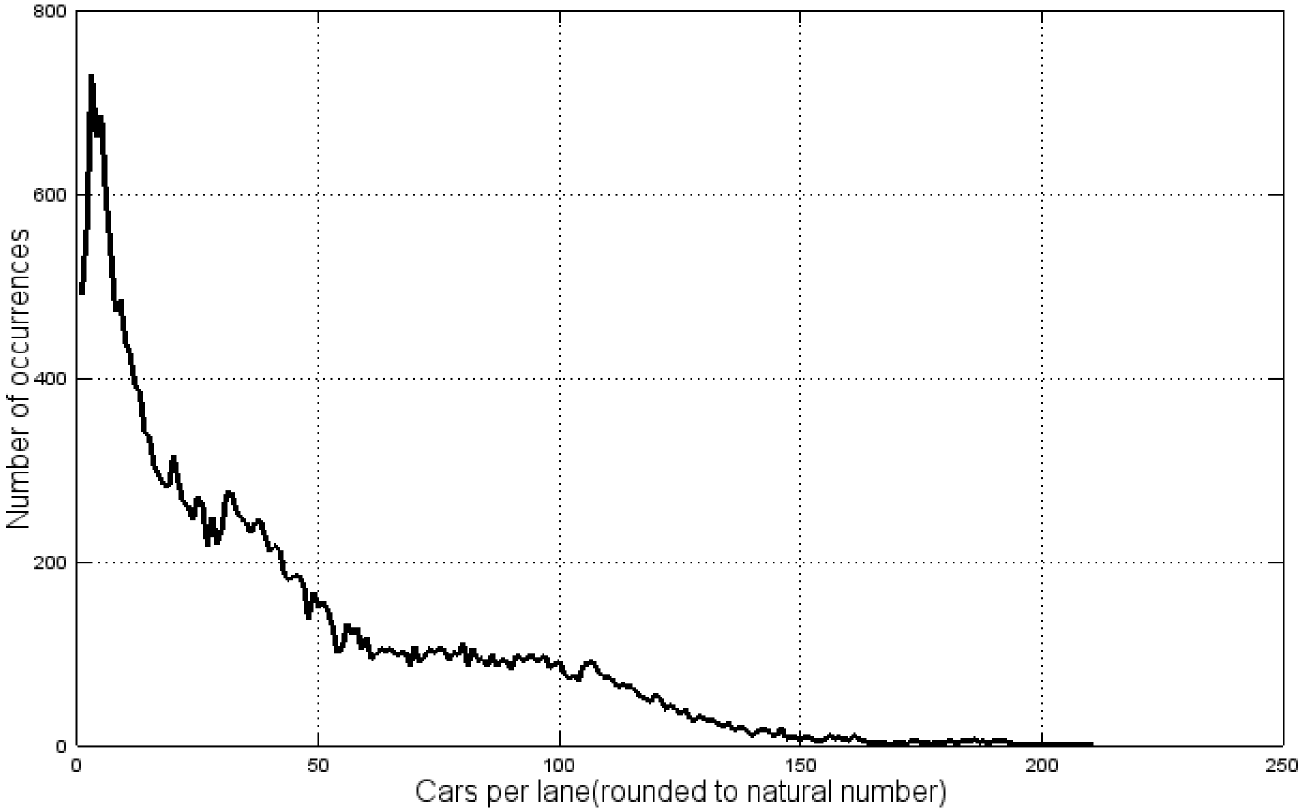


Figure 17: The frequency at which the calculated number of vehicles per lane occur for the 15 minute intervals for the period between Mon, 22 May 2012 00:00 CEST to Thu, 14 Jun 2012 22:15:00 CEST. The number of lanes is obtained from the associated to the sample area road segment, generated after processing of the OpenStreetMap source data. The number is rounded up and excludes intervals for which the calculated values are zero, since both missing and actual zero values are denoted in the same way, in the source data.

The resulting range of vehicles per lane was capped to a maximum n. For the number n, 15 different values were used, between 100 and 170 in steps of 5. The calculated values were divided in equal ranges and each range was mapped to a respective traffic level [Figure 18]. For each of the 15 values in combination with every range a separate training set was generated, in which the input vectors of training examples were assigned to the calculated traffic level for each of the combination, as a supervisory signal.

For the first of the two groups of regression models, as a supervisory signal was used the actual number of vehicles per lane and the for the second group - the actual utilization factor divided by the number of lanes.

The same procedure was used map the utilization factor to a traffic level, but instead of the number of vehicles per lane, calculated was the duration, for which a vehicle was over a detector, divided by the number of lanes [Figure 19]. For n were used 11 different values between 50 to 100 in steps of 5.

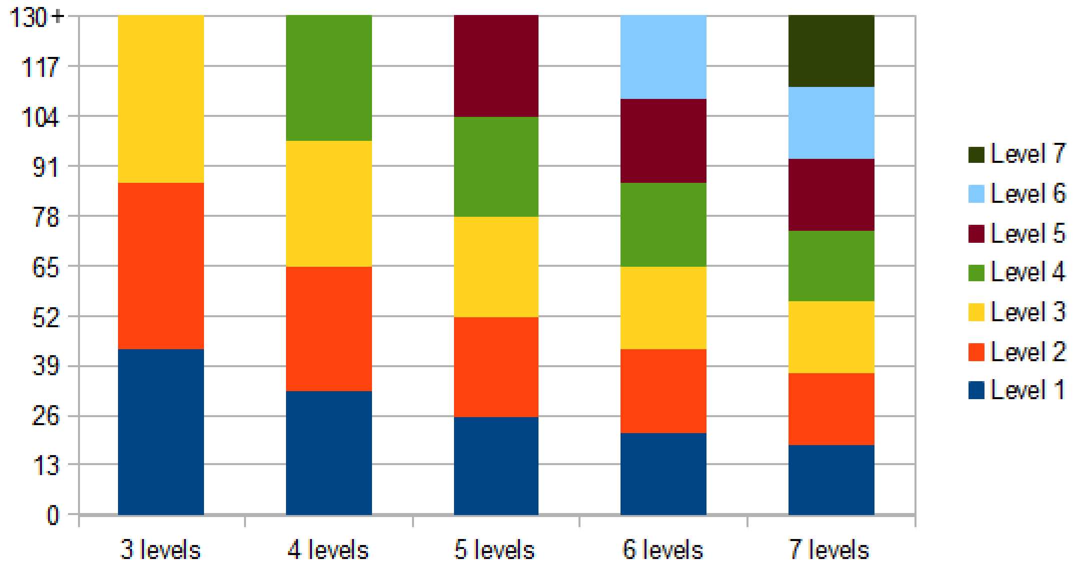


Figure 18: An example of the mapping between the number of vehicles per lanes that drove through a road segment and a discrete traffic level. The maximum number n in this example is set to 130 vehicles per lane. All instances, for which the number is higher than 130 vehicles per lane, are mapped to the highest level.

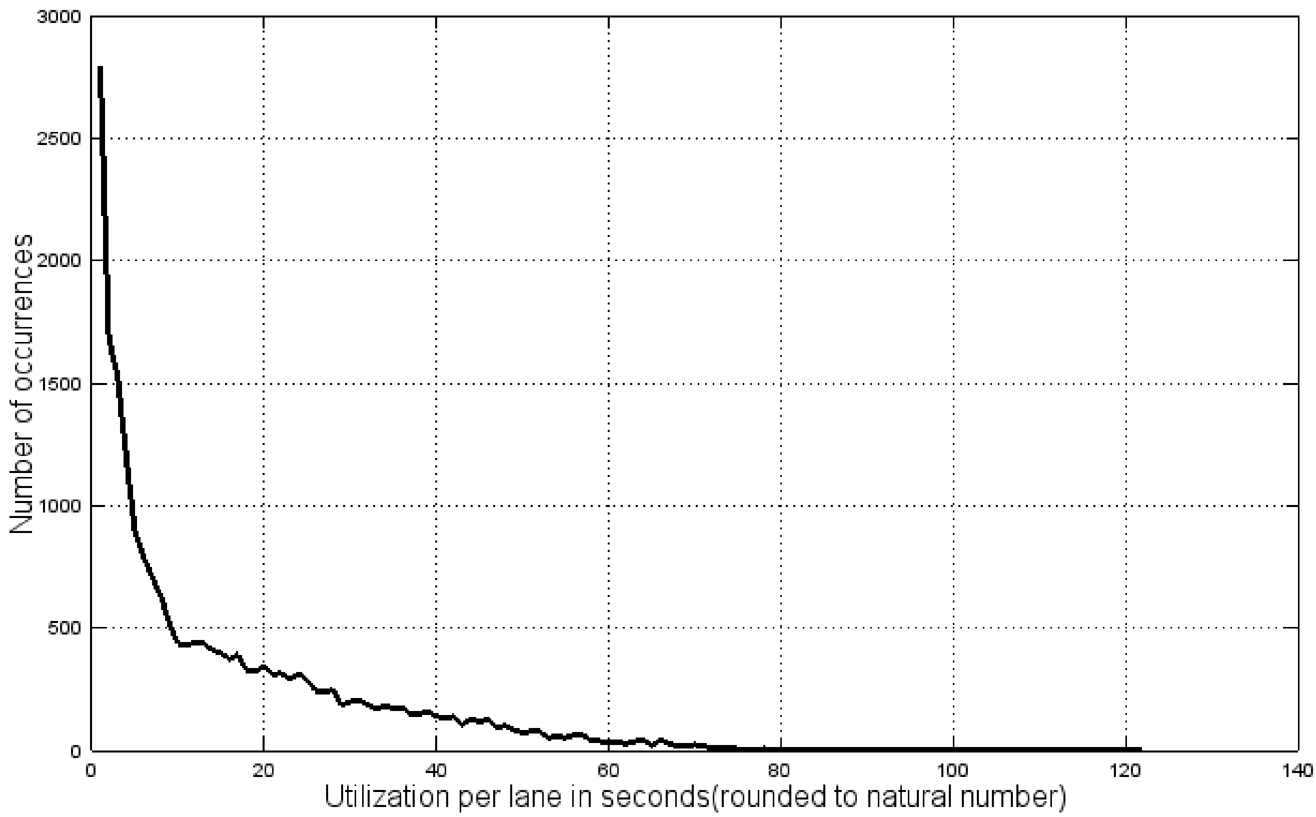


Figure 19: The frequency at which the calculated duration, in seconds, for which a vehicle was over a detector divided by the number of lanes, occur for the 15 minute intervals for the period between Mon, 22 May 2012 00:00 CEST to Thu, 14 Jun 2012 22:15:00 CEST. The number is rounded up and excludes intervals, for which the calculated values are zero, since both missing and actual zero values are denoted in the same way, in the source data.

5.1 Generation of training datasets

A software module to generate training sets from the processed data and produce classifier or regression function using supervised learning algorithms. The module relies on the open source libraries Weka, available under GNU General Public License and LIBSVM [7], available under BSD license.

Since the processed data as described, in the previous sections, allows the multiple different sets to be build, the module includes a dataset generator, which allows to construct training data, with the desired properties, that the algorithms will learn from.

The generator allows to create training sets differing by the type of the area used to select the measurements included in the set, the time interval for which they are selected and various others. The full list of options that could be selected for the training set is listed bellow:

- Type of the recording position - each of the measurements recorded during this work contain two different positions - one acquired by the GPS receiver of the recording device, with accuracies of less then 5 and less than 10 meters and another determined manually, with an accuracy of less than one meter. Using the manually determined position, results in better spatial filtering of measurements, which are associated to a road segment and a more accurate average distance of the recording positions to the road geometry.

- Selection area type - the type of the area, which is gonna be created and used as a spatial filter to select sound pressure level measurements. The area could be customized based on the maximum distance from the road segment(*drs*), the minimum distance from neighbor road segments(*dns*) and the outward or inward polygon offsetting in meters(*exp*). Three selection area types were used during this work, differing by the minimum distance from neighbor road segments, with *dns* values of 10, 20 and 30 meters.

- Time interval *ti* for which the traffic is monitored. Measurements recorded in the specified intervals are used to calculate and averaged sound pressure level, which would be used as an input for the training samples. Training samples in each created set include the averaged sound pressure value for non-overlapping time intervals of length *ti*. Though the software allows an interval with the same duration as the one of used for averaged sound pressure level measurements, to be selected, when data for shorter intervals is made available.. The total L_{eq} for the interval is calculated applying the following equation [19]:

  $$L_{eq} = 10 * log_{10}(1/n * \sum_{1}^{n} 10^{L/10})\ dB$$

  Training sets with intervals of 1, 3, 5, 10 and 15 minutes were generated for this project. As only traffic information for intervals of 15 minutes is currently available, the determined traffic level for this 15 minute interval is used as a supervisory signal for training sets containing samples with measurement data for shorter intervals.

- The type of supervisory signal(output), of the training examples in the set. A real value would result in sets for training regression models, a class - for classification models.

- The supervisory signal used for regression models. For such models either the number of vehicles per lane or the duration for which a vehicle was over one of the detectors divided by the number of lanes, could be selected.

- The number of traffic levels - specifies the number of traffic levels that are going to be used as a supervisory signal for this model. A detailed information is available in Section 5.3.

- Method for determining the traffic level - either the number of vehicles per lane or the duration for which a vehicle was over one of the detectors divided by the number of lanes.

- The range used for determining a traffic level. A detailed information is available in Section 5.3.

An overview of the types of training sets is presented in Figure 20. A total of 7860 training sets were generated, for this project. Each of the sets contains a training example as described in Figure 21.

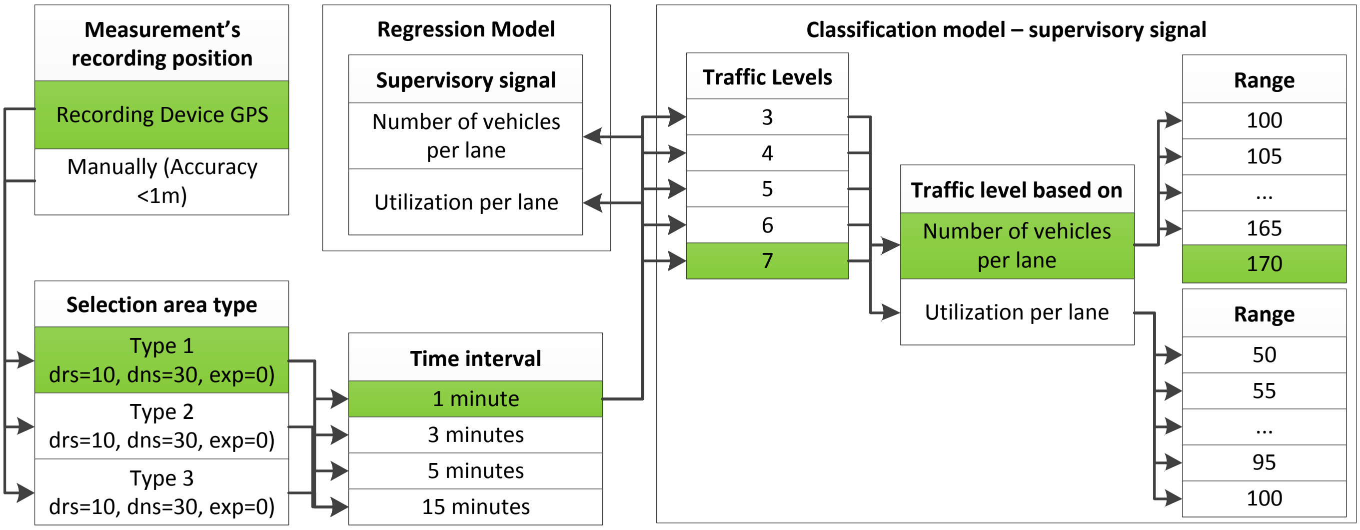


Figure 20: The example demonstrates one of the possible training sets.

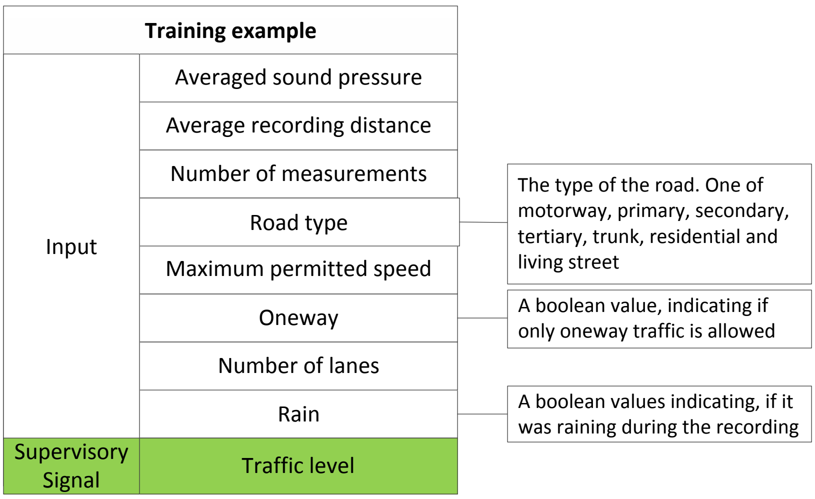


Figure 21: Each of the generated training sets contains examples as described above.

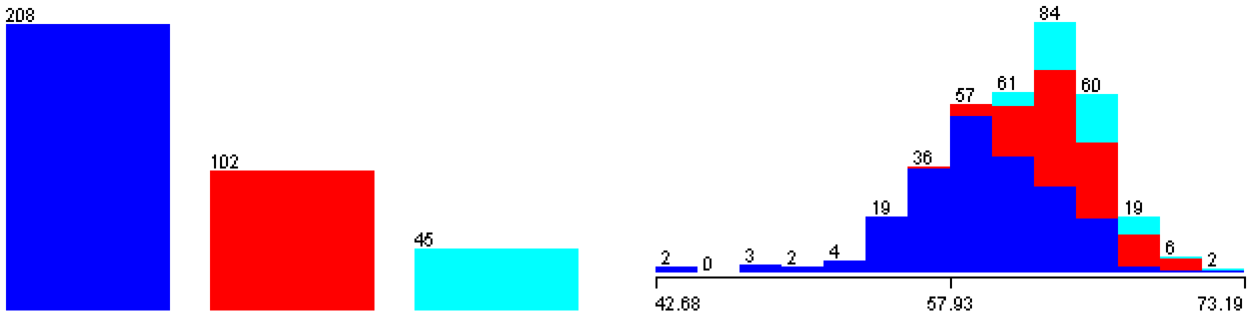
Due to missing entries in the source data of traffic activity and the scarce amount of sound pressure measurements near the 17 road segments used as sample areas in this project, training data, containing averaged sound pressure for 15 minutes, contained insufficient data, in order to be meaningfully analyzed further by learning algorithms.

We present five models - four classification models, labeled with A, B, C and D, and a regression model labeled as D, trainied with data, containing averaged sound pressure for 1 and 3 minutes. The models could be used as a reference for analyzing the accuracy of this method for traffic monitoring, when sufficient amount of data is available.

- **Model A** The model is trained with data containing the averaged sound pressure for non-overlapping intervals of 1 minute. A selection area with a maximum distance from the road segment of 10 meters, a minimum distance of 30 meters from neighbor segments and outward offsetting of 6 meters is used to filter the measurements. The measurements are filtered, based on the position of the device during their recording, acquired from the GPS receiver.

  The training set consist of 355 instances, for which as supervisory signal is used a class with 3 values, representing 3 traffic levels. The level is determined based on the number of vehicles per lane with a range maximum of 165 vehicles. An overview of the metrics of the traffic level and averaged sound pressure attributes is in Figure 22.

  The model uses a multilayer perception(MLP), a feed-forward neural network, with 2 hidden layers, each containing 9 neurons. A back-propagation algorithm with momentum [15] is used to train the network for 100 epochs. The amount of the weights is updated with a learning rate of 0.9 and the momentum applied to the weights during updating is 0.3.



(a) Traffic level **(b)** Averaged sound pressure

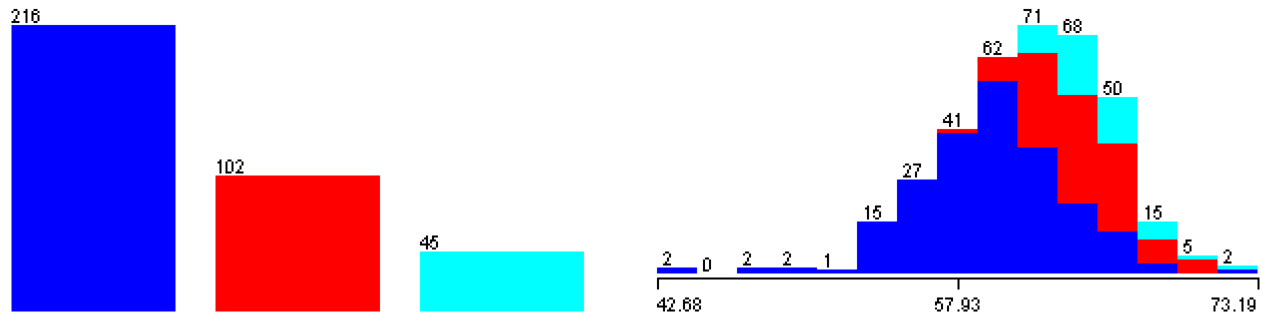
Figure 22: Distribution of the traffic level (a) and the averaged sound pressure (b) attributes for the instances included in model A.

- **Model B**

  Model B is a support vector machine model, trained with data containing averaged sound pressure for non-overlapping intervals of 1 minute as well. In contrast to model A though, the training data for this model is created using the manually determined recording position is used. This results in a slightly bigger number of measurements included, and respectively a total number of training samples(363), even though no outward offsetting of the selection area is used this time.

  The maximum distance of the selection area from the road segment and the minimum distance from neighbor segments of respectively 10 and 30 meters are used as in model A, as are the number of traffic levels and the method, with which the traffic level is determined. An overview of the metrics of the traffic level and averaged sound pressure attributes is presented in Figure 22.

  The model utilizes a C-Support vector classification [4]. A polynomial kernel $K(x_i, x_j) = (\gamma(x_i^T x_j)^d$ [10] with $\gamma = 0.3$ and $d = 3$ is used. The tolerance of termination criterion ϵ is set 0.1 and the cost parameter C of the minimizing function to 0.8.



(a) Traffic level **(b)** Averaged sound pressure

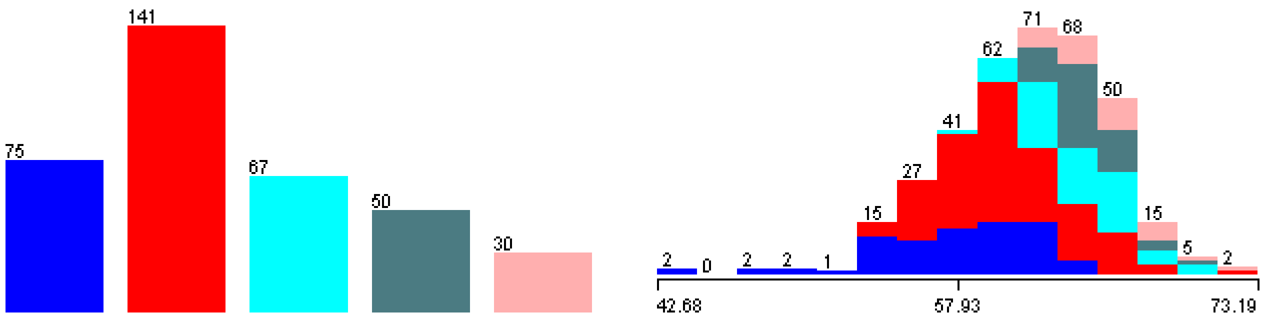
Figure 23: Distribution of the traffic level (a) and the averaged sound pressure (b) attributes for the instances included in model B.

- **Model C**

  In contrast to the other classification models presented here, 5 traffic levels are used in this model. The model is included to demonstrate the possibility of more accurate monitoring of the traffic activity in comparison to the other presented models. The traffic level is determined based on the number of vehicles per lane. The range maximum is set to 165 vehicles.

  Similarly to model B, the manually determined recording position is used for building the training set for the model. The selection area used for filtering of the measurements is constructed based on a spatial buffer with radius of 10 meters around the road segment and exclude the space in radius of 30 meters from neighbor segments. No inward or outward offsetting is applied to its geometry, as well and therefore the number of training samples is the same - 363.

  For this model a bayesian network is used as a classifier. TAN(Tree Augmented Naive Bayes) [9] search algorithm is used for structure learning with global metrics evaluated by leave one out cross-validation. The conditional probabilities are estimated with $P(x_i = k|pa(x_i) = j) = (N_{ijk} + N'ijk)/(N_{ij} + N'ij)$ [5].



(a) Traffic level **(b)** Averaged sound pressure

Figure 24: Distribution of the traffic level (a) and the averaged sound pressure (b) attributes for the instances included in model C.

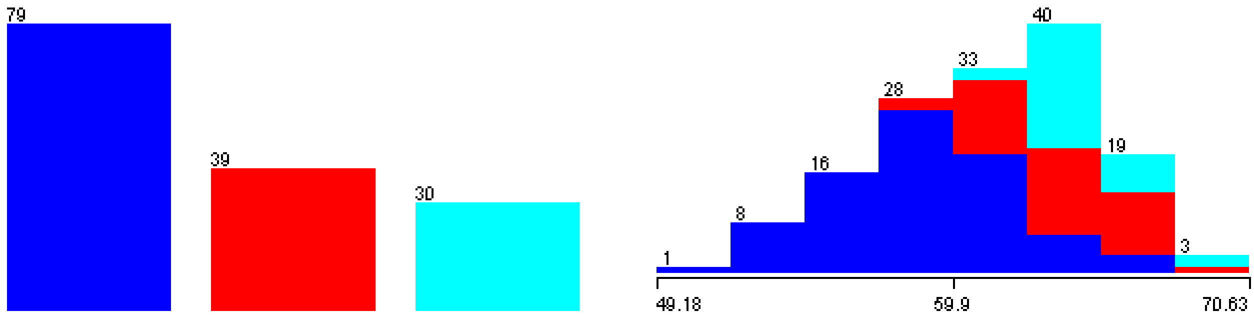
- **Model D**

  This training samples in this model, contain the averaged sound pressure for an interval of 3 minutes, in contrast to the other models presented here, where 1 minute interval is used. Consequently, the training set for this model contains significantly less training samples in relation to the rest of the models - 148.

  One of 3 traffic levels is assigned as supervisory signal. The traffic levels are calculated based on the number of vehicles per lane with a range maximum of 135 vehicles.

  The selected measurements and their average distance in the training data are based on the manually determined recording position. The same selection area is applied as spatial filter, as for the other models using the manually determined position.

Similarly to model B, C-Support vector classification is used. A radial basis function(RBF) $K(x_i, x_j) = exp(-\gamma \|x_i - x_j\|^2)$ [10] is used as a kernel with $\gamma = 0.5$. The tolerance of termination criterion ϵ is set 0.1 and the cost parameter C of the minimizing function to 0.8.



(a) Traffic level **(b)** Averaged sound pressure

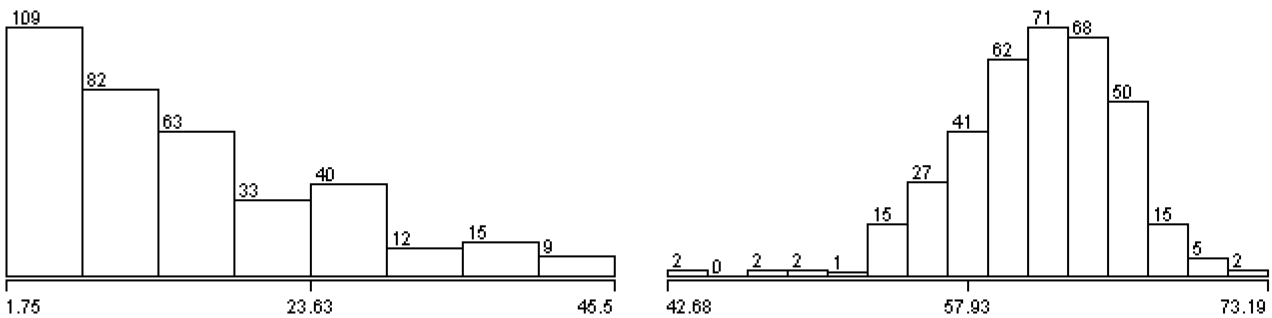
Figure 25: Distribution of the traffic level (a) and the averaged sound pressure (b) attributes for the instances included in model D.

- **Model E**

  The last model presented in this, produces a regression function, that estimates the conditional expectation of the supervisory signal given the input vector. The supervisory signal for the training samples is the utilization per lane, i.e. the total duration for which a vehicle was over one of the detectors for the road segment, divided by the number of lanes of the segment.

  The training samples contain as input the averaged sound pressure for an interval of 1 minute. The recording position with accuracy of less than 1 meter is considered, for the average distance of the road segment. As a spatial filter is applied the same selection area as described for models, which are trained with data sets based on the manually determined position.

  The model utilizes a ν-Support vector regression. A polynomial kernel $K(x_i, x_j) = (\gamma(x_i^T x_j)^d$ [10] with $\gamma = 0.3$ and $d = 4$ is used. The tolerance of termination criterion ϵ is set 0.1 and the cost parameter C and *nu* of the minimizing function are set to 11 and 0.2 respectively.



(a) Utilization per lane in seconds **(b)** Averaged sound pressure

Figure 26: Distribution of the utilization per lane, in seconds (a) and the averaged sound pressure (b) attributes for the instances included in model E.

As a mean of comparison the results of four simple linear models are also presented, which estimate the correlation factor between the averaged sound pressure measured for an interval of 1 minute in a selection area. The measurements are selected based on the recording positions with 1 meter accuracy. The same spatial filter is applied for selecting the measurements as for the models, presented in Section 5.2, which include training data generated, based on the same type of recording position.

The averaged sound pressure value for the interval is used as the only input for the training samples in the models. As a supervisory signal in the models is set respectively to the number of vehicles, the number of vehicles per lane, the duration for which a vehicle was over one of the detectors in seconds(utilization factor) and the utilization factor divided by the number of lanes for the 15 minute interval.

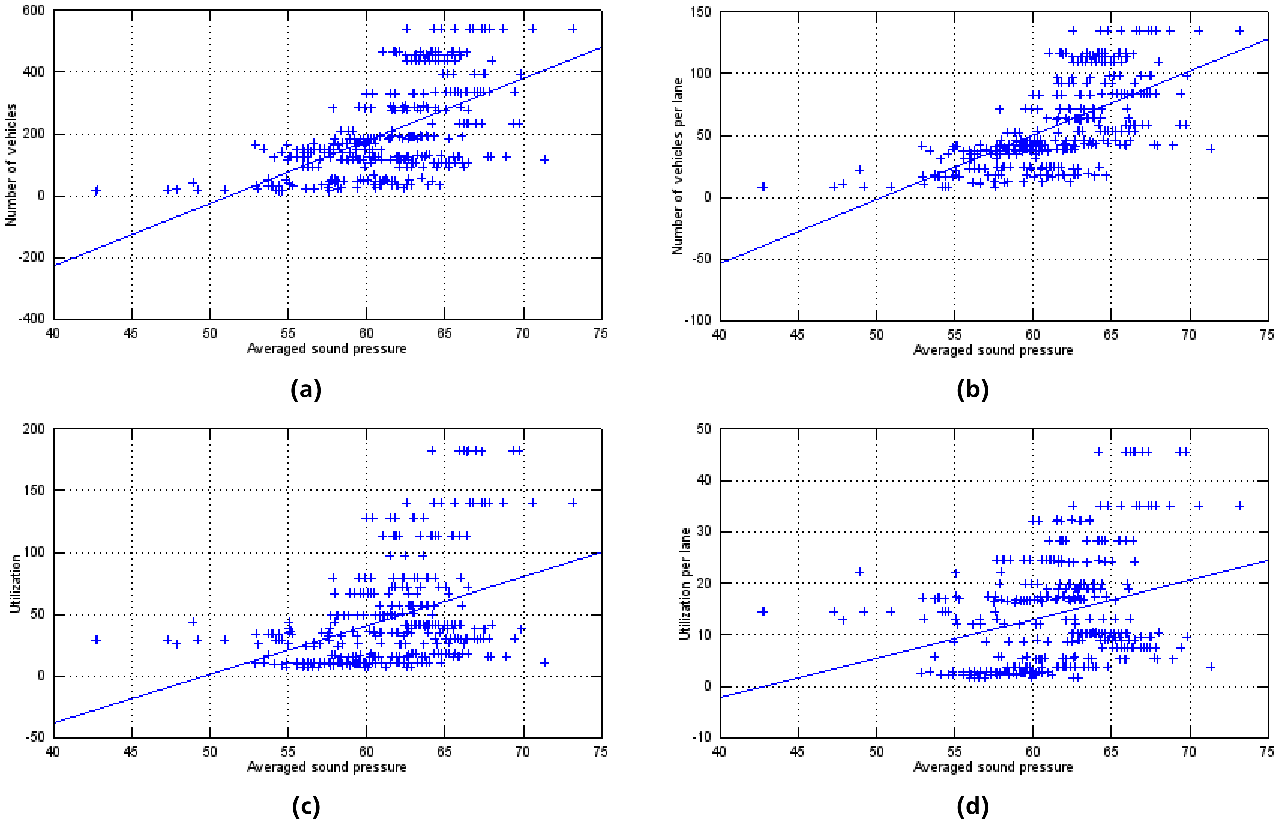


Figure 27: The number of vehicles vs averaged sound pressure level (27a) with a correlation factor of 0.5632. The number of vehicles vs averaged sound pressure level (27b), withe a correlation factor of 0.6228. The utilization vs averaged sound pressure level (27c), with a correlation factor of 0.3868. The utilization per lane vs averaged sound pressure level (27a) with correlation factor of 0.2835.

A 10-fold cross-validation, was used as a validation technique to estimate the performance models described in Section 5.2. With this method the generated training sets are divided into 10 subsets of equal size. Sequentially one subset is tested using the classifier produced by the model, trained on the rest 9 subsets. With this validation technique, each instance of the whole training set is predicted and the cross-validation accuracy is the percentage of samples, which are correctly classified.

The results from models A and B, shows that, as expected, models using measurements with recording position of less than 1 meter perform better than those using measurements , which recording position is determined by a GPS receiver(accuracy of less than 5 and less than 10 meters). Though their performance is close and sound pressure levels data geotagged based on information from a GPS receiver could be used to predict the traffic activity, when lower number of traffic activity levels are used. A detailed statistics for the performance of this model is presented in [Table 28a] and [Table 28b].

Even a better estimation of the traffic activity, by using higher number of traffic activity levels, could be achieved for measurements with good accuracy of the recording position. With accuracy of the position of less than 1 meter, model C was able to classify correctly 93.1129 % of the test instances, with 5 possible traffic levels.A detailed statistics for the performance of this model is presented in [Table 28c].

Despite that, in comparison to the rest of the models, the training set of model D contains significantly less instances, which were used to train the model, the produced classifier was able to correctly classify 94.5946 % of the test instances - only slightly worse than the produced by model B, where the averaged sound pressure for an interval of 1 minute is used. Considering the amount of training samples, the good results of the model hint, that a larger interval of measurement tend to increase the accuracy of the model. Due to the insufficient amount of data, such a conclusion require a further investigation. A detailed statistics of the model's performance is presented in [Table 28d].

The result of the regression model presented in this work(model E), clearly shows the gain in accuracy from the attributes included in the training sets, additionally to the averaged sound pressure. With a correlation factor of 0.9238 it shows that factors such as distance, weather conditions and various road characteristics have high impact on the accuracy of the model. For reference the presented in the beginning of this section linear model that estimates the conditional expectation of the same statistics given only the averaged sound pressure as input, shows a correlation factor of only 0.2835. A detailed statistics for the performance of this model is presented in [Table 28e]. [Table 28e].

Correctly Classified Instances	90.6336 %
Incorrectly Classified Instances	9.3664 %
Kappa statistic	0.8231
Mean absolute error	0.093
Root mean squared error	0.2239
Relative absolute error	25.2441 %
Root relative squared error	52.2163 %

(a) Model A - Position:<10m(GPS), Interval:1min, Levels:3

Correctly Classified Instances	95.5923 %
Incorrectly Classified Instances	4.4077 %
Kappa statistic	0.919
Mean absolute error	0.0294
Root mean squared error	0.1714
Relative absolute error	7.9749 %
Root relative squared error	39.9727 %

(b) Model B - Position:<1m, Interval:1min, Levels:3

Correctly Classified Instances	93.1129 %
Incorrectly Classified Instances	6.8871 %
Kappa statistic	0.9075
Mean absolute error	0.0428
Root mean squared error	0.1537
Relative absolute error	14.3256 %
Root relative squared error	39.762 %

(c) Model C - Position:<1m, Interval:1min, Levels:5

Correctly Classified Instances	94.5946 %
Incorrectly Classified Instances	5.4054 %
Kappa statistic	0.9101
Mean absolute error	0.036
Root mean squared error	0.1898
Relative absolute error	8.9202 %
Root relative squared error	42.2824 %

(d) Model D - Position:<1m, Interval:3min, Levels:3

Correlation coefficient	0.9238
Mean absolute error	2.268
Root mean squared error	4.1727
Relative absolute error	26.0719 %
Root relative squared error	39.2033 %

(e) Model E - Position:<1m, Interval:1min, Level:Utilization per lane

Figure 28: Statistics of the models' performance

As all of the training data contains information only for road segment with no specific regulations for the maximum permitted speed for the vehicles, no conclusion could be made for it's impact on the accuracy of models trained with data, including information for road segment with different speed limits.

The conclusions from the model results were also confirmed from the ranking filter tests that were performed - one based on gain ratio attribute evaluation and another one based on evaluation of the the worth of an attribute by computing the value of the chi-squared statistic with respect to the class.

7 Conclusion and future work

In this work a new approach for road traffic monitoring is presented. The method uses sound pressure measurements from both static devices with known location as well as mobile devices, aware of their geographical position. In our preliminary evaluation with a small number of measurement samples, the developed method shows promising results. Compared to existing solutions for road traffic monitoring, this technique presents a low-cost alternative. To the best of our knowledge this is the first approach using sound pressure records to create virtual traffic activity measurements.

We introduced a solution to automatically create road segments, for which the traffic could be monitored and build their two dimensional geometry, in order to calculate the distance of the mobile sensor from the road. Furthermore we presented a method defining a measurement area for sound, surrounding these road segments, that provides the ability to most accurately select input data from mobile sensors.

A software module, which allows to build and evaluate a number of supervised learning models, was implemented. It enables to generate various data combinations for training the models, to most precisely determine what information improves their accuracy.

Our preliminary evaluation indicates that our approach is indeed working. Some models yield a high recall and precision. Please be advised that the data set is too small for any final conclusions. Also be advised that both recall and precision could decrease a little as the measurement interval is increased to the real 15 minutes. Nevertheless the results can be considered a successful first step. Now we need further models, which are trained with datasets containing a reliable amount of sound pressure measurements, averaged for the same interval as that of the provided traffic data. We are positive that our method will operate successfully in the field but further research to evaluate its accuracy is required.

The road segments as is today already contain a large amount of additional information. Nevertheless we observed, during field research, that other factors could also influence the noise level. These include physical characteristics of the road, such as it's gradient and smoothness of it's surface, and weather characteristics, such as wind speed and height of precipitation. Taking those factors into account could have positive impact on the models' performance. The input data of the models must not necessarily contain only sound pressure level measurements. Combining information about sound pressure levels with data from sensors observing other phenomena, such as carbon dioxide could also be considered for improving the model.

An infrastructure for real-time traffic monitoring, using this method, could contain not only mobile sensors, but a combination of mobile and stationary ones. To this end we suggest the deployment of a selected amount of not necessarily low-cost sensors, that have proven high accuracy, in order to monitor the performance of the virtual sensor. Those sensors can also be used to continuously tune the infrastructure performance.

List of Figures

References

[1] Openstreetmap statistics report. `http://www.openstreetmap.org/stats/data_stats.html`.

[2] Inga Baubonyte and Regina Grazuleviciene. Road traffic flow and environmental noise in kaunas city. *Engineering*, 1(1):49–54, 2007.

[3] Lce Den Boer. Traffic noise reduction in europe. *CE Delft*, 14(August):2057–2068, 2007.

[4] Bernhard E. Boser and et al. A training algorithm for optimal margin classifiers. In *Proceedings of the 5th annual ACM workshop on computational learning theory*, pages 144–152. ACM Press, 1992.

[5] Remco R. Bouckaert. Bayesian network classifiers in weka for version 3-5-7. *Network*, pages 1–47, 2008.

[6] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava. Participatory sensing. *IEEE Internet Computing*, 14(1):12–42, 2006.

[7] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[8] Shiping Chen Shiping Chen, Ziping Sun Ziping Sun, and B Bridge. Traffic monitoring using digital sound field mapping, 2001.

[9] Nir Friedman, Dan Geiger, and Moises Goldszmidt. Bayesian network classifiers, 1997.

[10] Chih-Wei Hsu, Chih-Chung Chang, , and Chih-Jen Lin. A practical guide to support vector classification. *Bioinformatics*, 1(1):1–15, 2010.

[11] Roberto Lopez-Valcarce, Carlos Mosquera, and Fernando Perez-Gonzalez. Estimation of road vehicle speed using two omnidirectional microphones: A maximum likelihood approach. *EURASIP Journal on Advances in Signal Processing*, 2004(8):1059–1077.

[12] Prashanth Mohan, Venkata N. Padmanabhan, and Ramachandran Ramjee. Nericell : Rich monitoring of road and traffic conditions using mobile smartphones. *Proceedings of the 6th ACM conference on Embedded network sensor systems*, (c):323–336, 2008.

[13] Prashanth Mohan, Venkata N Padmanabhan, and Ramachandran Ramjee. Trafficsense : Rich monitoring of road and traffic conditions using mobile smartphones. *The 6th ACM Conference on Embedded Networked Sensor Systems*, pages 1–29, 2008.

[14] Pascal Neis, Dennis Zielstra, and Alexander Zipf. The street network evolution of crowdsourced maps: Openstreetmap in germany 2007-2011. *Future Internet*, 4(1):1–21, 2011.

[15] V V Phansalkar and P S Sastry. Analysis of the Back-propagation Algorithm with Momentum. *IEEE Transactions on Neural Networks*, 5(3):505–506, 1994.

[16] T. Priede. Origins of automotive vehicle noise. *Sound And Vibration*, 15(March 1969):61–73, 1971.

[17] Immanuel Schweizer, Roman Bärtl, Axel Schulz, Forian Probst, and Max Mühläuser. Noisemap - real-time participatory noise maps. *Second International Workshop on Sensing Applications on Mobile Phones*, page to appear, Nov 2011.

[18] Andrew Sheehy. Worldwide smartphone markets: 2011 to 2015 - analysis, data, insight and forecasts.

[19] Kumares Chandra Sinha and Samuel Labi. *Transportation Decision Making: Principles of Project Evaluation and Programming*, volume 75. John Wiley and Sons, 2007.

[20] Brian L. Smith, Mike Fontaine, and Matt Green. Cell phone probes as an atms tool. *Transportation*, 2003.

[21] Jungkeun Yoon, Brian Noble, and Mingyan Liu. Surface street traffic estimation. *Electrical Engineering*, 1001:220–232, 2007.