

# 3D SAFE ROUTING NAVIGATION APPLICATION FOR PEDESTRIANS AND CYCLISTS BASED ON OPEN SOURCE TOOLS

T. Santhanavanich\*, P. Wuerstle, J. Silberer, V. Loidl, P. Rodrigues, V. Coors

University of Applied Sciences Stuttgart  
Schellingstraße 24

70174 Stuttgart, Germany

(thunyathep.santhanavanich, patrick.wuerstle, jan.silberer, verena.loidl, preston.rodrigues, volker.coors)@hft-stuttgart.de

## Commission IV

**KEY WORDS:** Safe routing, pgRouting, CityGML, SensorThings API, OpenStreetMap, Open Geospatial Consortium

## ABSTRACT:

The recent advancement in Information & Communication Technology (ICT) is seen as a critical enabler to design intelligent smart cities targeting different domains. One such domain is modes of transport in a city. Currently, various cities around the world are envisioning innovative ways to reduce emissions in the cities by increasing physically active mobility. However, there is still limited information about the safety of cyclists and pedestrians within city limits. To address this, we develop a 3D web-based safe routing tool called *Vision Zero*. Our concept prototype used Augsburg city, Germany, as a case study. The implementation is based on open-source tools. In the back-end, the OGC 3D Portrayal Service standard helps to deliver and integrate various 2D and 3D geospatial contents on a web-based client using CesiumJS. The OGC SensorThings API (STAPI) standard is used to manage historical and real-time open road-incident data from the Federal Statistical Office of Germany. The navigation system is built up based on the routing engine *pgRouting*, which calculates the safest route based on the mentioned STAPI server and the road-network dataset from OpenStreetMap.

## 1. INTRODUCTION

The most effective way to reduce emissions in cities is to increase physically active mobility. This not only helps cities reduce emissions and noise but also helps people to live healthily and stress-free (Greenpeace e.V., 2018). However, according to the Federal Statistical Office of Germany, every day, one cyclist dies on German roads (Statistisches Bundesamt, 2018). Although in the past 60 years the number of people killed or injured due to traffic accidents has fallen drastically, the number of fatal bike accidents has remained mostly constant since 1991. Furthermore, in 2017 more people died in cities using bicycles than cars. It indicates that cyclists are the most vulnerable road users and unprotected in city traffic. However, shifting individual motorized transport to active forms of mobility requires clear government guidelines and infrastructure spending. (Statistisches Bundesamt, 2018)

In the 2018 coalition agreement, the German government committed itself to reduce the number of road deaths to zero (Koalitionsvertrag zwischen CDU, 2018). Furthermore, it proposed that municipalities invest in infrastructure. Moreover, it suggested providing an incentive to companies that propose active mobility as an option to multimodal transport (Bundesministerium für Umwelt, 2018). To support the government initiative and ensure an accident-free bicycle commute, in this paper, we propose ① a methodology that can help cyclists and pedestrians identify the most accident-prone areas in the city and ② a development on a proof-of-concept 3D Web-GIS application to educate the general public.

The rest of this paper is organized as follows. Section 2 discusses the related works. Section 3 describes the implement-

ation of our approach with a proof-of-concept prototype. Section 4 discusses the result of our research. Finally, section 5 concludes the overall outcome of the paper.

## 2. BACKGROUND AND CONCEPT

This section describes the necessary elements needed to develop the 3D safe routing navigation application for pedestrians and cyclists. It consists of three main parts, including 1. web-based navigation system, 2. GIS method for identifying accidental risk area, 3. web map or web globe, and 4. data management system.

### 2.1 Web-based Navigation System

**2.1.1 Open-Source Navigation Tool** One of the most popular tools to build up the open-source navigation system is by using *pgRouting*<sup>1</sup> with a street dataset from Open Street Map. *pgRouting* is extension of the PostGIS and PostgreSQL geospatial database which enables the geospatial routing functionality. Several researchers have successfully used it for developing navigation platforms for specific purposes, such as the optimal routing service for emergency scenarios (Choosumrong et al., 2019), for the fastest route (Chadha, Garg, 2018), or the healthy urban route for cyclists and pedestrians (van der Molen et al., 2017). In this research, we used *pgRouting* as the main navigation tool for our back-end.

**2.1.2 Commercial Navigation API** The development of navigation tools has been a rapidly growing field in the last few years, several IT companies have provided the navigation system as the API services for application developers. The example of these well-known service providers are Google Maps

\* Corresponding author

<sup>1</sup> <https://pgrouting.org/>

platform<sup>2</sup>, OpenRouteService<sup>3</sup>, and Here Maps API<sup>4</sup>. Using one of these services provides an easy-to-use experience for developers to build up a navigation application. However, the drawback of using these services is they lack the facility to create custom algorithms for optimal navigation with the user's input and external data, and they also limit the number of API calls per month. In this paper, we used the services from Here Maps API only to evaluate our navigation system.

## 2.2 The GIS Method for Identifying Accidental Risk Area

Due to the high number of accidents on the road, identifying the accident risk location has been an active part of the research for many years. In the past, the Kernel Density Estimation (KDE) had been widely used by many researchers for accidental analysis and visualizing the spatial clusters of accidents as a result of "hot spots" or "heat map" (Schneider et al., 2004, Pulugurtha et al., 2007, Xie, Yan, 2008, Anderson, 2009). In this paper, we applied the KDE as a tool for creating a danger zone with different levels of accident risk for pedestrians and cyclists.

## 2.3 Web Map or Web Globe

To develop the web-based navigation systems, the web map is the essential fundamental component, as it visualizes the graphical representation of the earth and significantly improves the user's navigated experience. In the recent years, not only the 2D-based web map applications are presented, but also the 3D-based web map or so-called *Web Globe* had been developed and introduced with a significant improvement in visualizing and user experience perspective (Rivlin et al., 1994). Examples of the web globe services are Google Earth, CesiumJS<sup>5</sup>, Nasa-WorldWind, Mapbox, and ArcGIS Earth. In this paper, we selected CesiumJS as it supports the visualization of 3D city models and it is an easy-to-learn platform, an open-source service with many examples and tutorials, and supports from the big developer community. The CesiumJS is an open-source JavaScript library for creating a 3D globe application that supports several GIS standard formats and provides high visual quality through the web.

## 2.4 Data Management System

In this research, we developed our back-end based on the OGC open standards, which benefited in the long run as they foster supports data interoperability, data management, and data transaction from the data source to the web application. The two primary open standards used in this research include the OGC SensorThings API for managing accidental data and the OGC 3D Portrayal Services for managing 3D data contents.

**2.4.1 OGC SensorThings API** The OGC SensorThings API (STAPI) is an OGC standard giving an open and unified framework to interconnect IoT devices, data, and applications, which simplifies communication among several applications and observation tools (Liang et al., 2016). In this research, we used the STAPI for managing daily accident data, which was used for visualizing on the user client, and used as an input for safe routing with pgRouting.

<sup>2</sup> <https://cloud.google.com/maps-platform>

<sup>3</sup> <https://openrouteservice.org/category/api>

<sup>4</sup> <https://developer.here.com>

<sup>5</sup> <https://cesium.com/cesiumjs/>

**2.4.2 OGC 3D Portrayal Services** To address the 3D needs of the geospatial community, the OGC has proposed the 3D Portrayal Services (3DPS) as a geospatial 3D content delivery implementation specification (Hagedorn et al., 2017). In this research, the 3DPS GetScene request was implemented and available through a HTTP/GET KVP-based request.

## 3. IMPLEMENTATION

Overall system architecture is illustrated in figure 2. As shown, it consists of four main parts, namely; **Data Sources**, **Server-side**, and **Users**. Further details of each step are explained as follows.

### 3.1 Data Sources

The three main data sources of our work consist of the street network (figure 2-1), the accident data (figure 2-2), and the 3D city model (figure 2-3).

**3.1.1 Street Network Data** The Geofabrik<sup>6</sup> service extracts the data from the OpenStreetMap and provides the 2D GIS data, which are normally updated every day. From this service, we downloaded the 2D street network data in the Schwaben region, Bayern, Germany. Then, we extracted only the data of our study in a boundary of Augsburg city using QGIS software. The final dataset contains 35,561 segments in a total length of approximately 2,850 km. Figure 1 shows the overview of the Augsburg street dataset displaying in QGIS software. The street network data (figure 2-1) was imported to the navigation server (figure 2-4) with the osm2pgrouting<sup>7</sup> tool.

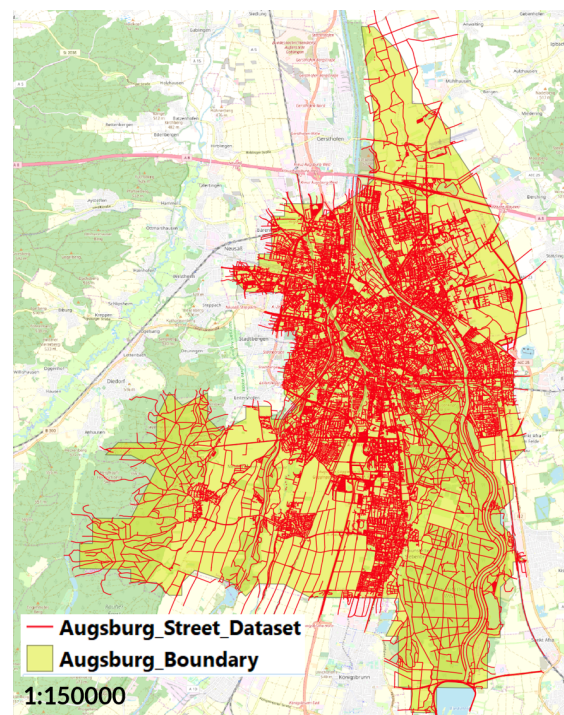


Figure 1. The overview of the input street network dataset in the area of Augsburg displayed on the QGIS software.

**3.1.2 Accident Data** We acquired the open historical road accident geo-data accessed from the download portal provided by the Federal Statistical Office of Germany<sup>8</sup>. The data includes

<sup>6</sup> <http://download.geofabrik.de/>

<sup>7</sup> <https://github.com/pgRouting/osm2pgrouting>

<sup>8</sup> [https://unfallatlas.statistikportal.de/\\_opendata2019.html](https://unfallatlas.statistikportal.de/_opendata2019.html)

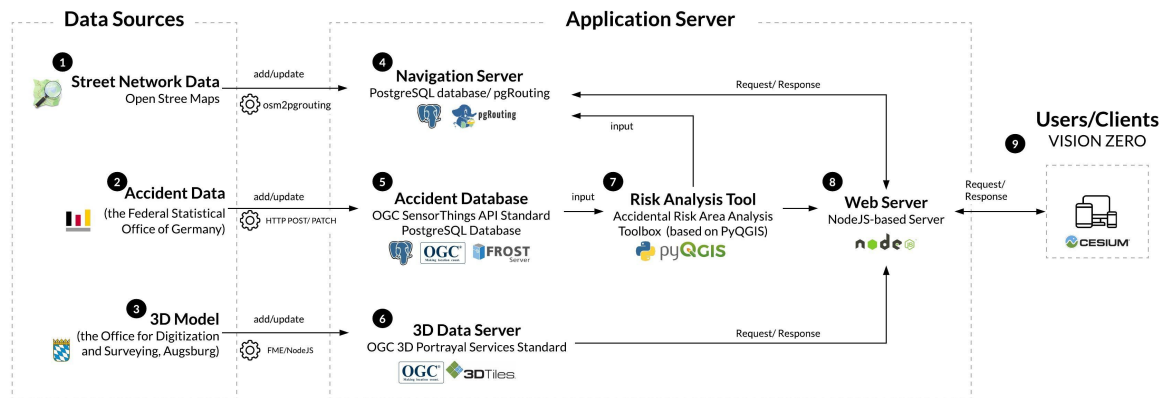


Figure 2. The overall system architecture.

a total of 138,369 and 195,229 incidents in 2016 and 2017, respectively, in the whole of Germany. We filtered these data to achieve only the accidents that happened in our study area of Augsburg city and the accidents in which pedestrians or cyclists were involved. After the data cleaning, there were 824 and 839 matching accidents in 2016 and 2017, respectively.

**3.1.3 3D City Model Data** The 3D city model data (figure 2-3) were used to visualize on our client application to give a better insight into the 3D environment to the user. The CityGML data in our study area was given to us for academic purposes by the Office for Digitization, Broadband, and Surveying, Augsburg (Landesamt für Digitalisierung, Breitband und Vermessung, 2019). This dataset was originally in the CityGML format, which is one of the OGC standards for representation and exchange of 3D city models and had been successfully used in many application domains (Gröger, Plümer, 2012). However, the CityGML was not fit for the web-based visualization, so it had to be converted to the 3D Tiles format for effectively visualizing the 3D geospatial contents over the web or application. The FME software<sup>9</sup> with the education license was used to convert the model from CityGML format to 3D Tiles format.

## 3.2 Application Server

The server-side of our project consisted of five important elements including the navigation server 2-4), the accident database 2-5), the 3D data server 2-6), the risk analysis tool 2-7), and the web server 2-8).

**3.2.1 Navigation Server** The navigation server was built up based on the PostgreSQL database with the pgRouting extension. As mentioned in chapter 3.1.1, we imported the Open Street Map data of our study area to this server. Then, we used it as a basis to calculate the navigation result according to the parameters from the users. These parameters are shown with examples in table 1.

Parameters	Data type	Example
Starting Position	Array	[lon,lat]
Destination Position	Array	[lon,lat]
Secure Routing Option	Boolean	True

Table 1. User parameter for navigation

<sup>9</sup> <https://www.safe.com/>

When users use the navigation feature on the application (figure 2-9), the HTTP request is sent through the web server (figure 2-8) with the mentioned parameters, which will be matched with the vertex in the network and trigger the route calculation on the navigation server and return the best route to the users. In the current implementation, the Dijkstra algorithm is used to calculate the shortest route by taking the length in meters as an input cost parameter. To avoid the edge with a high risk of accidents, the Risk Analysis Tool (figure 2-7) keeps updating the negative value to the cost field on the high-risk edge as the algorithm computes only on edges with positive costs.

**3.2.2 Accident Database** In the accident database (figure 2-5), the FROST SensorThings API server<sup>10</sup> is used to collect the accident database. According to the SensorThings API standard, the data can be uploaded to the server using the HTTP POST protocol. In order to manage the data transfer from the CSV source to the SensorThings server efficiently, we had written a Python script to automate this process.

**3.2.3 3D Data Server** The 3D data server (figure 2-6) is implemented based on the OGC 3D Portrayal Services. We implemented this server based on Node.js. In this research, this server was mainly used to manage the 3D city model dataset, which is the 3D Tiles format. The main reason for using the 3DPS standard was to broaden the availability of the server not to limit one research application and to make this approach usable in other research or applications.

**3.2.4 Risk Analysis Server** The risk analysis server (figure 2-7) is used to calculate the area with high risk of accidents. The server used PyQGIS to calculate KDE to find the hotspot of the risk area was based on the data from the accident database (figure 2-5). The server was programmed to watch if there is a new update to the accident database then this analysis job will be triggered to update the updated result area. The resulting area will be then updated to both the navigation server and the web server. The navigation server used the vector result to calculate the negative value to the cost field of the street vertex intersecting with high-risk areas. The high accident area was exported as a raster sending to a web server to show the user this area as a heatmap layer on the 3D application.



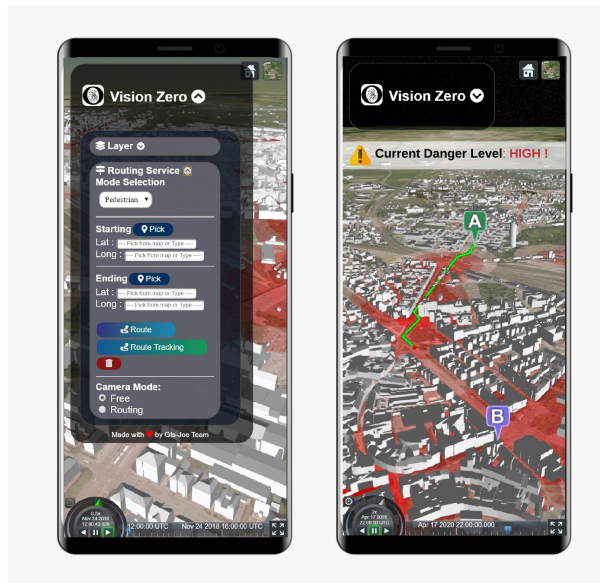


Figure 3. Application screenshots with screen sizes of 375x812, tested with Google-Chrome web browser.

### 3.3 Vision Zero Application

We developed the 3D web client application called *Vision Zero* (figure 2-9) based on the *CesiumJS* web globe library. Figure 3 shows the example of *Vision Zero* application's user interface on the mobile using the Google Chrome web browser. To use the navigation tool, users are prompted to input the starting position and ending position by input [lat,lon] or select position directly on the app.

In the back-end, *Node.js* with *Express.js* framework is used for the web server (figure 2-8). The user's request is sent to the web server and transferred to the navigation server (figure 2-4). By default, the route will consider the risk area and avoid those areas and provide a safe routing result to users. Also, the 3D data content from 3D data server (figure 2-6) is sent to the client through this web server.

For the data visualization, the app visualizes the 3D building models in Augsburg with the terrain models and allow users to understand the real-world environment better in the city. The routing result also visualized as a timely feature animated on the application. The geofencing concept is also applied to show an alert to users whenever users are passing through a high accident risk area.

## 4. DISCUSSION

We developed a web-based application that gives a visualization of risk-level for pedestrians and cyclists by the three-dimensional hypothesized environment. By integrating accident statistics, technical sensor data, and subjective user data, the application reacts to the needs of cyclists and pedestrians to find the safe route. However, with the safe-routing feature, guiding all cyclists to use the safest routes might affect the fluidity of the traffic on the road, or even could that decrease the safety level of that route.

<sup>10</sup> <https://github.com/FraunhoferIOSB/FROST-Server>

From the perspective of visualization, our application is based on a 3D terrain and also visualizes the risk area with a heat map. This approach bears certain advantages over a 2D representation. It gives valuable context for the risk evaluation. For example, users can understand why certain areas have a high-risk assessment over others if they are located at a steep slope. A GIS application in a 3D perspective visualizes more layers than a common 2D Web-GIS. In this application, the terrain model, the 3D city model, and 3D geographic data are shown. For example, the terrain model is essential for cyclists and pedestrians to select an adequate route for themselves. This can also be easily added as a parameter for the routing in the future. Also, the applications appeal in 3D is higher due to a better impression of the environment and a raised user experience.

From the perspective of using open standards for data management, the STAPI allows high interoperability of IoT and sensor data management. It can be easily expanded to supply the data stream to multiple projects and applications. Furthermore, it provides additional sensor measurements to be easily incorporated into the project. With a similar goal in mind, the OGC 3DPS standard is used and implemented in this application on the server-side using *Node.js*. It is used to specify the way geospatial 3D content is delivered to the client. Additionally, it benefits the 3D data interoperability of the application. In this research, we only use 3D Tiles as 3D data sources. Still, in future other data formats such as point clouds, i3s, glTF can be added to the server, which helps to expand to serve more projects or integrate into the existing infrastructures similar to the STAPI. The 3D Tile data provided by the 3DPS shows outstanding performance for visualizing different layers of massive 3D data through the web. Although the 3D Web-GIS requires high bandwidth and high processing power, the rapid development of computer hardware technology and internet technology has compensated this limitation. The routing, as an integral part of this application, is done by the *pgRouting* extension to PostgreSQL. Currently, we use the *Dijkstra* algorithm for the routing service. In the future, other routing algorithms and other open-source routing tools like *Neo4j* appear to be an alternative solution that can be implemented and compared to the solution in place, *pgRouting*.

## 5. CONCLUSION

In conclusion, we developed a safe routing navigation application called *Vision Zero*. It relies on the open-source tools and standards, including OGC SensorThings API for accident data, OGC 3D Portrayal Services for 3D data content, PyQGIS library for the analysis of danger zone areas, and *pgRouting* with PostgreSQL as a navigation tool. All data sources are open data except the CityGML 3D city model, which is limited to use only for research purposes.

The app uses open data to visualize users' risk levels on city roads in real-time. The proof-of-concept was implemented for the city of Augsburg. For future work, we plan to integrate other data such as real-time weather or user feedback into the analysis model in our application.

## ACKNOWLEDGEMENTS

This work has been jointly developed in the project *i\_city* (funding number: 03FH9I01IA). The project *i\_city* is supported by the German Federal Ministry of Education and Research (BMBF).

The authors are responsible for the content of this publication. The authors would also like to thank the anonymous reviewers who have helped to increase the quality of the paper.

Xie, Z., Yan, J., 2008. Kernel Density Estimation of traffic accidents in a network space. *Computers, Environment and Urban Systems*, 32(5), 396–406. <https://doi.org/10.1016/j.compenvurbsys.2008.05.001>.

## REFERENCES

Anderson, T. K., 2009. Kernel density estimation and K-means clustering to profile road accident hotspots. *Accident Analysis & Prevention*, 41(3), 359–364.

Bundesministerium für Umwelt, Naturschutz, B. u. R., 2018. Klimaschutz in zahlen - fakten, trends und impulse deutscher klimapolitik.

Chadha, C., Garg, S., 2018. Shortest path analysis on geospatial data using PgRouting. *International Conference on Innovative Computing and Communications*, Springer Singapore, 201–214.

Choosumrong, S., Humhong, C., Raghavan, V., Fenoy, G., 2019. Development of optimal routing service for emergency scenarios using pgRouting and FOSS4G. *Spatial Information Research*, 27(4), 465–474. <https://doi.org/10.1007/s41324-019-00248-2>.

Greenpeace e.V., 2018. *Radfahrende schützen - klimaschutz stärken: Sichere und attraktive wege für mehr radverkehr in städten*.

Gröger, G., Plümer, L., 2012. CityGML – Interoperable semantic 3D city models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 71, 12–33. <https://doi.org/10.1016/j.isprsjprs.2012.04.004>.

Hagedorn, B., Thum, S., Reitz, T., Coors, V., Gutbell, R., 2017. OGC® 3D Portrayal Service 1.0. Version 1.0.

Koalitionsvertrag zwischen CDU, C. u. S., 2018. Ein neuer aufbruch für europa, eine neue dynamik für deutschland, ein neuer zusammenhalt für unser land.

Landesamt für Digitalisierung, Breitband und Vermessung, 2019. 3d-gebäudemodell: Grundlage für vielfältige anwendungen.

Liang, S., Huang, C.-Y., Khalafbeigi, T., 2016. OGC SensorThings API Part 1: Sensing, Version 1.0.

Pulugurtha, S. S., Krishnakumar, V. K., Nambisan, S. S., 2007. New methods to identify and rank high pedestrian crash zones: An illustration. *Accident Analysis & Prevention*, 39(4), 800–811. <https://doi.org/10.1016/j.aap.2006.12.001>.

Rivlin, E., Botafogo, R., Shneiderman, B., 1994. Navigating in hyperspace: designing a structure-based toolbox. *Communications of the ACM*, 37(2), 87–97.

Schneider, R. J., Ryznar, R. M., Khattak, A. J., 2004. An accident waiting to happen: a spatial approach to proactive pedestrian planning. *Accident Analysis & Prevention*, 36(2), 193–211.

Statistisches Bundesamt, 2018. Unfallentwicklung auf deutschen straßen 2017.

van der Molen, M., Ligtenberg, A., Vreugdenhil, C., Steeneveld, G.-J., 2017. Development of a Healthy Urban Route Planner for cyclists and pedestrians in Amsterdam. *EGU General Assembly Conference Abstracts*, EGU General Assembly Conference Abstracts, 12361.