SonicRoutes: an application to find the quietest routes

Giulio Bello, Giulio Capecchi, Federico Frati, Jacopo Niccolai  
 [g.bello2@studenti.unipi.it](mailto:g.bello2@studenti.unipi.it), [g.capecchi2@studenti.unipi.it](mailto:g.capecchi2@studenti.unipi.it), [f.frati4@studenti.unipi.it](mailto:f.frati4@studenti.unipi.it), [j.niccolai@studenti.unipi.it](mailto:j.niccolai@studenti.unipi.it)

ABSTRACT

The issue that many cyclists and pedestrians face every day is the danger posed by road traffic. Even in urban areas, speed limits are often high and frequently ignored by road users, which discourages the use of non-motorized means of transportation. Finding the quietest route, which is likely to be less trafficked, can provide cyclists and pedestrians with a tool to ensure their safety. Traditional routing applications are based on the fastest path between two points, whereas in this case, the focus is on the cumulative noise along the road segments connecting the start and end points.

This document presents a brief introduction to the problem to demonstrate the utility of an application like this, an analysis of the client-server architecture used to collect data and manage the functionalities provided to the user, the presentation of the experimental data collected, an analysis of battery consumption during use, and a concluding section to summarize the findings and propose potential future improvements.

1 Introduction

In recent years, several local administrations have started to introduce speed limits of 30 km/h in city centers. It has been demonstrated that this measure benefits both vulnerable road users and motorists. The percentage of fatalities in the former category has significantly decreased, while travel times in these areas have also reduced in parallel with the decrease in road congestion. Slow zones also serve as a tool to boost the economy in the affected areas and encourage social mixing, acting as meeting points for people from otherwise poorly connected areas.

Road noise can be seen both as a disturbance during travel and as an indicator of the presence of motor vehicles. The idea behind the app is to leverage a tool that everyone always has on hand, capable of monitoring both noise and location using a microphone and GPS, the smartphone. By having a large network of users, it would be possible to collect enough data to accurately estimate the average noise pollution experienced on each road segment under analysis. In our case, we focused on the city of Pisa, specifically the area surrounding the Faculty of Engineering, to work with a graph of manageable size.

2 Architecture (or another name for the section)

The solution adopted for the application is based on a Client-Server paradigm. The map data was sourced from OpenStreetMap, filtering intersections, and retaining their coordinates and the roads involved. This allowed us to create a graph that the user can navigate.

2.1 Map

2.1 Client

To access the application, the user must authorize the use of their location and microphone. The client is represented by smartphones, which have a dual functionality. The first is to sample noise and location at regular intervals, calculate the average sampled noise, and communicate this to the server whenever a segment change occurs. This contributes to the data collection, which is essential for the proper functioning of the app. The second functionality is to request the path with the lowest cumulative average noise to reach one of the available checkpoints on the map from the current location.



**Figure 1: Figure Caption and Image above the caption**

2.2 Server

The server is implemented using Flask and exposes resources to be contacted by the client via APIs.

3 Experimental results (or another name for the section)

Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here.

3.1 Heading Level 2 (for subsections)

Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here.

3.2 Heading Level 2 (for subsections)

Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here.

4 Conclusion

Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here. Insert paragraph text here.

REFERENCES

[1] Patricia S. Abril and Robert Plant, 2007. The patent holder's dilemma: Buy, sell, or troll? *Commun. ACM* 50, 1 (Jan, 2007), 36-44. DOI: <https://doi.org/>10.1145/1188913.1188915.

[2] Sten Andler. 1979. Predicate path expressions. In *Proceedings of the 6th. ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages (POPL '79)*. ACM Press, New York, NY, 226-236. DOI:https://doi.org/10.1145/567752.567774

[3] Ian Editor (Ed.). 2007. *The title of book one* (1st. ed.). The name of the series one, Vol. 9. University of Chicago Press, Chicago. DOI:https://doi.org/10.1007/3-540-09237-4.

[4] David Kosiur. 2001. *Understanding Policy-Based Networking* (2nd. ed.). Wiley, New York, NY..

Conference Name:ACM Woodstock conference

Conference Short Name:WOODSTOCK’18

Conference Location:El Paso, Texas USA

ISBN:978-1-4503-0000-0/18/06

Year:2018

Date:June

Copyright Year:2018

Copyright Statement:rightsretained

DOI:10.1145/1234567890

RRH: F. Surname et al.

Price:$15.00