



How Data Mining Can Improve Road Safety in Cities

Elena Butsenko

Department of Business Informatics, Institute of Digital Management Technologies and Information Security, Ural State University of Economics, 620144 Ekaterinburg, Russia; evl@usue.ru; Tel.: +7-922-609-0688

Abstract: Traffic collisions pose a serious problem for cities due to the annually increasing number of vehicles. Information about incidents that occur on roads is important for the corresponding monitoring bodies, authorities, and emergency services. To ensure traffic safety, the data have to be visible, clean, and transparently displayed. This research was, therefore, aimed at developing a methodology for monitoring motor vehicle collision data and applying visualization techniques to evidence from New York City. The method showed that the largest number of motor vehicle traffic crashes occurred in Lower Manhattan due to its high population and traffic density. With these data, the road agencies of the city can put potentially dangerous road sections under control and make them safer for both drivers and pedestrians. Further development of the system may be associated with data analytics and visualization, resulting in new layers of heatmaps that not only provide details on car collision hotspots, which serve as the main target indicator for traffic safety authorities, but also break them down into social facilities, such as schools. This feature will enable assessment of how safe it is around a school and the evaluation of the impact of an underpass or a traffic enforcement camera on the number of collisions. The motor vehicle traffic crash (MVTC) monitoring system will help in comparing city districts and regions in terms of safety, seeing trends, realizing what exactly is happening at interchanges, and understanding the reasons behind. The methodology, in addition, can be supplemented with an analysis of risk factors for MVTCs, the efficiency of adopted measures and road renovations that are carried out, and many other functions.



Citation: Butsenko, Elena. 2022. How Data Mining Can Improve Road Safety in Cities. *Social Sciences* 11: 130. <https://doi.org/10.3390/socsci11030130>

Academic Editors: Richard E. Bilborrow and Nigel Parton

Received: 14 January 2022

Accepted: 14 March 2022

Published: 16 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: data mining; big data; data analysis; dataset processing; data science; data analytics; data visualization; monitoring; heatmap; motor vehicle traffic crash

1. Introduction

The high road accident rates across all countries, the ever-increasing numbers of persons who are injured and of deaths due to traffic collisions, the considerable property damage to vehicles, the harm to and loss of cargo, and the damage to traffic structures pose one of the most acute social problems in the world. Ensuring road safety is becoming a global challenge for humanity.

According to Tufts University (Boston, MA, USA), traffic collisions are some of the most common domestic emergencies¹. The USA has one of the highest road densities in the world, with more than 227 million licensed drivers and 280 million vehicles in operation. This is the reason behind the large number of motor vehicle traffic crashes (MVTCs); in 2018, there were more than 12 million car collisions across the country; as the statistics for 2019 show, motor vehicle crashes took the lives of 36,120 people in the US, which resulted in 11 deaths per 100,000 inhabitants, while in 2020, more than 10 million MVTCs occurred.

Tracking how, where, and why MVTCs happen requires the application of dataset analysis. Most of the information in the modern world is obtained through visual perception, with heatmaps being one of the main data visualization techniques. The methods for their generation can also be employed when developing monitoring systems for emergencies such as MVTCs.

The purpose of this study consists in devising a methodology for monitoring MVTCs based on a heatmap drawn in *Python*. The MVTC heatmap uses data on the number of

traffic collisions that occurred in New York City over 8 years and helps track the intensity of crashes in its different boroughs and the scenes of the incidents; as a result, a set of measures that are likely to prevent emergencies on the roads is worked out. The main task is to single out the locations with the highest MVTC rates.

As an analytical tool resting on artificial intelligence, the heatmap aids in figuring out the relationships between various functions in a dataset. It is a way of representing the data values as colors, where the redder, “hotter” shades are used to denote a strong correlation between elements, while the lighter, “colder” (or bluer) shades reflect a weak one. Thus, heatmap analysis can become one of the best ways to leverage the results from the collected data.

2. Previous Research on the Topic

A heatmap, also known as a thematic map, is a method for visualizing the intensity of a particular indicator within a specific area on the map. Data on the map can be assigned a hatching of different patterns, colors of a certain degree of saturation (choropleth map), or dots (dot distribution map) (Borland and Taylor 2007). A thematic map provides a graphical representation of statistical data that are divided into groups based on geographical factors (for example, by administrative divisions). It portrays territories on the map, which are colored (or hatched) in accordance with the value of the studied indicator observed across the area.

The history of computer-generated heatmaps originated in 1991 when C. Kinney coined the term to depict financial market information in real time (Rowell et al. 2020). Hierarchical color coding is employed in various data representation systems—for instance, in cluster analysis (Yu et al. 2020), web analytics (Kirsh 2021), and generating biological heatmaps (Chen et al. 2020).

It should be noted that, apart from colors, other graphical means can be used as a third coordinate to display 3-D data in 2-D space. For example, as shown in Lu et al. (2021), when plotting a bubble diagram, the authors expressed the third dimension through the area or size of a circle. In Friendly Michael (2021), the values of the variables were visualized with the height of the bars in the bar chart. Supplementary dimensions can be added by including new groups of data.

In recent years, there has been a large number of studies covering the development of visualization systems for aiding traffic safety. Let us take a closer look at the latest major scientific developments in data mining that can be applied in order to make urban traffic safer. In Bishop (2021), for example, I. Bishop considered the impact of visualization on infrastructure and changes in the landscape of the area. The author suggested utilizing data mining to examine the spatial correlations between indicators. The method showed that median visibility could be seen as the best condition for analyzing and visualizing the processed data.

In Albuquerque et al. (2021), the authors worked out an intelligent transport solution for Lisbon. They applied a cross-industry standard process, CRISP-DM, to understand problems with the traffic in the city using artificial intelligence. The methods of analysis and classification were employed to make models for calculating traffic indices that were correlated with external factors.

A visualization system based on urban traffic data mining was described in Bachechi et al. (2022). The use of such a system allows the areas of roads that are congested at specific times to be detected and enables the monitoring of traffic sensors. The authors produced a dashboard based on traffic sensor data that simulated traffic flow in two cities in Spain. The application of the dashboard was aimed at reducing traffic issues and mitigating transport emissions.

The need for increased traffic safety as a societal concern was reflected in the paper by Berres et al. (2021). The authors used both abstract and spatial visualization methods to carry out data mining of big traffic safety datasets collected in Hamilton County, Ohio, USA. The traffic safety indicators were captured in different locations: on the highways, in

the city center, and in remote areas. The study stands out from all the rest precisely because of the employment of spatiotemporal patterns that characterize motor vehicle collisions in order to give a better understanding of common incident consistency in the collected data.

A system for improving traffic safety in the UK based on the incorporation of the visualization method into machine learning was devised in 2017 (Liu et al. 2022). It utilized a hybrid algorithm, LightGBM-TPE, which is claimed to perform better than other similar machine learning algorithms in terms of accuracy and recall. The algorithm established the geographic coordinates, time, and day of the week as four risk factors for traffic collisions.

The application of data mining to the visualization of MVTCs in Saudi Arabia was explored in Almajyul and Al-Ghreimil (2022). The paper introduced a software tool that represented a system for visualizing motor vehicle collisions and eased the perception of traffic data in order for the authorities to be able to limit their number. The system integrated a word-cloud, a bar chart, an interactive map, a doughnut chart, and a calendar as visualization methods.

An extensive review and analysis of scientific studies on intra-city traffic data visualization was presented in Clarinval and Dumas (2021). The authors identified mobility issues such as pollution and congested roads. Information visualization was noted as a medium for making the most out of these traffic data. As a result of the study, the authors discovered that there was insufficient attention to research on MVTC visualization and a lack of visualizations that engaged citizens in reflection on mobility-related issues.

In view of the above, creating a system for monitoring MVTCs in a major city will give a fresh look into the problem of car collisions and make it possible to find optimal solutions for reducing their number (Butsenko 2015; Shorikov Andrey 2021). The task of developing new tools for monitoring motor vehicle collision data and subsequent decision making aimed at handling the traffic proves to be critical and relevant. Meeting this challenge will stimulate sustainable development of the city.

3. Materials and Methods

The main techniques employed to devise the proposed methodology for monitoring MVTC data comprised data-mining methods, namely, methods for big data visualization. The solution was based on *Python*, a high-level programming language that was used to write the software.

PyCharm, an integrated and cross-platform environment, served as a development tool.

Since *Python* is a console programming language, to display the final result as an MVTC heatmap, it needs a library that enables the output of certain graphical information (Lienemann et al. 2020). This particular solution used the *Jupyter Notebook* library, which is an interactive shell for the *Python* programming language. It is especially good as a standard command shell of an operating system, particularly with regard to *Windows*, where the basic one has limited capabilities. The default behavior is similar to that of the shells of *UNIX*-like systems; however, operating in a *Python* environment leads to greater customization and flexibility. Thus, *Jupyter Notebook* is the most convenient web environment for *Python* in order to combine code, text, and images and to distribute them to other users.

Another software library, *Pandas*, which provides a means of data processing and analysis, was required to work with .CSV files. It is built on top of the *NumPy* package, which is considered to be a lower-level tool (Harris et al. 2020). The library contains data structures and operations for manipulating numerical tables and time series. Its main functions involve collection, cleaning, initial assessment, analysis, and modeling of data (Lemenkova 2020; Sapre Atharva 2020).

To map heat markers, the *Folium* package was used, which is an open-source intuitive *Python* library that visualizes spatial data in an interactive manner.

4. Results

The results of the study are summarized as a description of a step-by-step methodology for monitoring and visualizing the MVTC data and their implementation.

4.1. Algorithm and Development Diagram

Setting up a system for monitoring traffic collisions starts with devising an algorithm and preparing a development diagram for it, which are required in order to have a full understanding of how the finished software will work.

The algorithm for creating an MVTC monitoring system includes the following steps:

Step 1. Initializing the dataset. Downloading the “Motor Vehicle Collisions—Crashes” dataset from data.cityofnewyork.us.

Step 2. Initializing libraries: *OS* (enabling work with files: deleting, saving, or modifying), *Folium* (for plotting a heatmap based on a clean map and heat markers), *Pandas* (allowing operations with a dataset: deleting, modifying, or creating data).

Step 3. Defining variables: *carCrashDataset*—for further work with the dataset.

Step 4. Cleaning the *longitude* and *latitude* columns of incorrect data.

Step 5. Creating the locations variable to store the clean data.

Step 6. Generating an empty (clean) black-and-white map and saving it to variable *m*.

Step 7. Plotting heat markers using the *Heatmap* function.

Step 8. Generating a heatmap by combining the results from steps 6 and 7.

Step 9. Saving the map as a *Heatmap.html* file.

Step 10. The final step of the algorithm: displaying the map on the screen.

Figure 1 shows a flowchart of the development of the MVTC monitoring system.

Before beginning the development process, the workspace needs to be prepared. It involves installing the required software, such as the programming language, development environment (and making adjustments to the configuration), and appropriate libraries.

The program design stage comprises coding and outputting the result. If the program does not run or operates incorrectly, the previous stage needs to be repeated until the desired result is produced (Brink et al. 2016).

4.2. Plotting the MVTC Heatmap

Generating a traffic collision heatmap entails downloading a dataset in .CSV format from the official New York City Open Data website. The dataset provided by the New York Police Department has almost 2 million registered motor vehicle collisions and is updated daily. The file size is about 300 MB (Figure 2) (NYC Open Data 2021).

According to *step 2* of the aforementioned algorithm, the *Jupyter* environment must be properly initialized. The library may not work correctly due to the need for loading a large amount of data. To avoid this problem, the config value, *iopub_data_rate_limit*, must be changed by entering a specific command (Figure 3) (Lu et al. 2021).

The environment is fully ready for the development after successfully opening it up and creating a new project.

Next, the required *Folium* and *Pandas* libraries need to be initialized along with the *OS* system library. This will help save the final result as an .html file in order to view it without the need for an internet connection and independently of the project (Figure 4).

After that, in accordance with *step 3*, direct work with the MVTC dataset can be begun. The entire data array is loaded into the memory by using the *read_csv* function. It should be remembered that this dataset is very large, and to operate on it properly in the *Pandas* library, the *low_memory* parameter has to be adjusted.

During the processing of the dataset, it turned out that several cells of the *latitude* and *longitude* columns were missing values. To avoid errors and incorrect calculations of the coordinates of points on the map, they were temporarily assigned the value 0 with the *fillna* function (Figure 5).

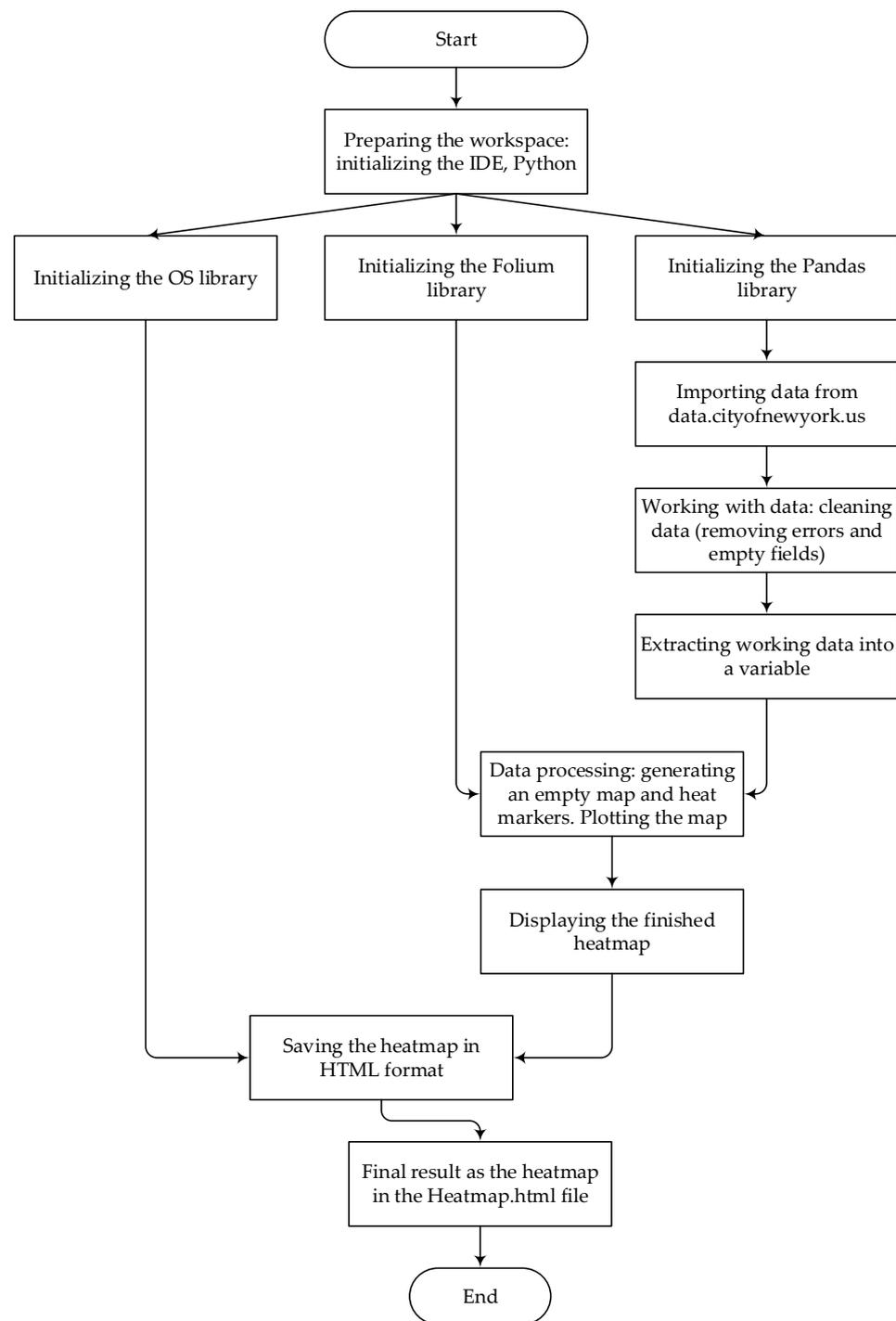


Figure 1. Diagram of the development of an MVTC monitoring system based on a heatmap.

Once this is done, all fields have the correct values and can be further transformed. Subsequent testing revealed that the set contained incorrect width and longitude data. The width and longitude vary between -180 and 180 degrees and must be non-zero. The corrections need to be made by excluding such data. Firstly, the zero values are deleted, followed by those beyond the range of -180 to 180 (Figure 6).

The set contains data since 2012. To make it relevant, we drew a heatmap for the whole year, that is, for the previous 12 months. One of the greatest advantages of the *Pandas* library is that it can read dates as *strings*. To process annual data, it is necessary to set a

condition that the date in the *CRASH DATE* column is less than or equal to a specific value in the last year—for example, from 13 May 2020 to 13 May 2021 (Figure 7).

Updated October 16, 2021		Update	
Data Last Updated October 16, 2021	Metadata Last Updated April 19, 2021	Update Frequency	Daily
Date Created April 28, 2014		Automation	Yes
		Date Made Public	5/7/2014
Views 297K	Downloads 71.4K	Dataset Information	
Data Provided by Police Department (NYPD)	Dataset Owner NYC OpenData	Agency	Police Department (NYPD)
		Data Collection	
		Data Collection	Motor Vehicle Collisions

Figure 2. Description of the employed dataset.

```
Terminal: Local x +
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.
(venv) C:\Users\root\PycharmProjects\carCrashHeatMap>jupyter
er notebook --NotebookApp.iopub_data_rate_limit=1.0e10
```

Figure 3. Running the *Jupyter* environment to work with the dataset.

```
In [1]: import os
import folium
import pandas as pd
from folium.plugins import HeatMap
```

Figure 4. Adding the libraries required for operation.

```
In [2]: carCrashDataset = pd.read_csv("carCrashDataset.csv", low_memory=False)
carCrashDataset['LATITUDE'].fillna(0, inplace = True)
carCrashDataset['LONGITUDE'].fillna(0, inplace = True)
```

Figure 5. Adding the dataset and initial value processing.

```
clearLat = carCrashDataset[(carCrashDataset['LATITUDE'] != 0)]
clearLat = clearLat[(clearLat['LATITUDE'] >= -180)]
clearLat = clearLat[(clearLat['LATITUDE'] <= 180)]

clearLon = clearLat[(clearLat['LONGITUDE'] != 0)]
clearLon = clearLon[(clearLon['LONGITUDE'] >= -180)]
clearDataset = clearLon[(clearLon['LONGITUDE'] <= 180)]
```

Figure 6. Cleaning the dataset of incorrect data.

```
carCrashDataset = carCrashDataset[~(carCrashDataset['CRASH DATE'] <= '05/13/2020
```

Figure 7. Selecting the data starting from 13 May 2020.

The data for generating the MVTC heatmap are ready. Since this dataset has other columns, those required for further use can be written to the *locations* variable (Figure 8). The appropriate data are now prepared for the heat overlay.

```
locations = clearDataset[['LATITUDE', 'LONGITUDE']]
```

Figure 8. Uploading data for plotting a map into the *locations* variable.

To put the heat overlay on, the map has to be initialized by utilizing the *Folium* library. In this paper, a *stamentoner* map was generated to produce a clean black-and-white image (Figure 9).



Figure 9. Base layer of the heatmap.

Step 7 involves uploading the ready-to-use data onto the map and applying the preset *locations* variable, followed by specifying values for the *radius* and *min_opacity* parameters. The *radius* and *min_opacity* are 20 and 0.1, respectively (McKinney 2017). *Radius* = 20 means that the point plotted on the map will be approximately 20 meters. *Min_opacity* refers to the arrangement of the colors in sequence from cold to hot, where 0.1 is the coldest (blue color). *Max_opacity* is calculated automatically. After all of the values have been loaded and processed, the data are plotted on the base layer generated above (Figure 10).

```
HeatMap(locations, radius = 20, min_opacity = 0.1).add_to(m)
```

Figure 10. Generating an MVTC heatmap.

To produce the outcome, the *m* variable is employed, which contains a rendered map and the result of the *HeatMap* function (hotspots and their depth with radius) (Figure 11).

The plotted map shows water areas around the city in black, Northern Brooklyn and the South Bronx are highlighted in yellow, and Lower Manhattan is highlighted in red,

which indicates a large number of car collisions in this part of the city due to the high population and traffic density.

To be able to view the output outside this environment, we shall use the *save* function and save the map as a heatmap file in .html format to the local storage (Figure 12).



Figure 11. Finished MVTC heatmap. View from the inside of the *Jupyter* environment.

```
m.save('Heatmap.html')
```

Figure 12. Saving the heatmap in .html format.

The final heatmap can be viewed and zoomed in the browser (Figure 13).

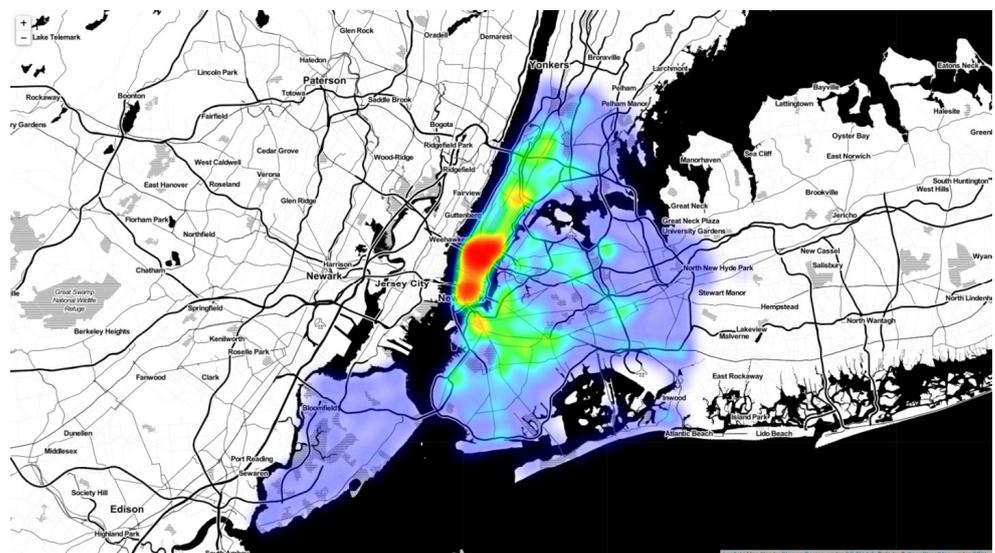


Figure 13. General view of the city and the MVTC heatmap overlay.

Figure 14 shows a zoomed-in MVTC heatmap. Figure 15 demonstrates a zoomed-in heatmap of traffic collisions in the western part of the city.



Figure 14. Zoomed-in MVTC heatmap of the city.

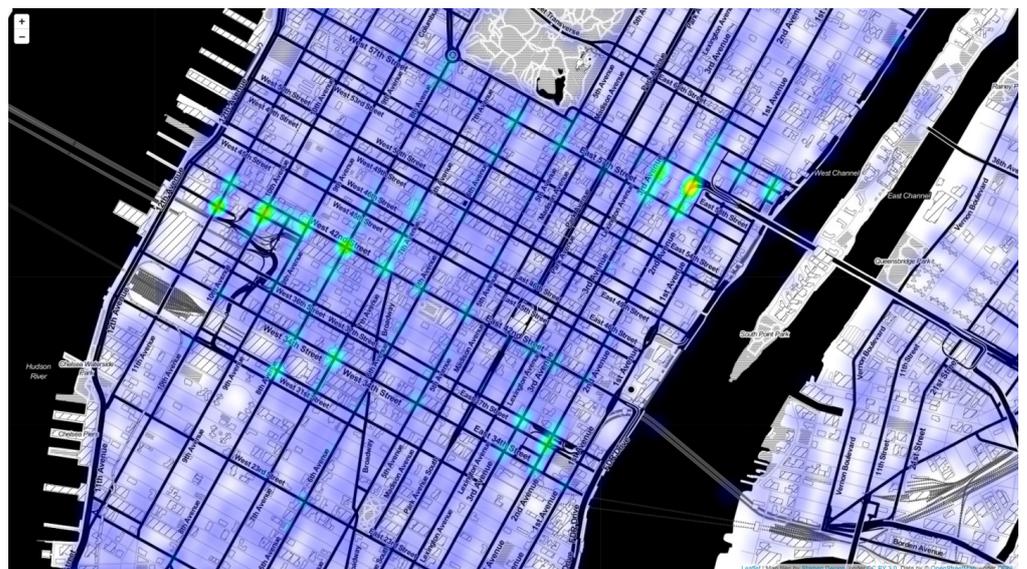


Figure 15. Zoomed-in heatmap of traffic collisions in the western part of the city.

This completes the description of the implementation of the methodology for monitoring and visualizing the data on MVTCs in a major city.

5. Conclusions

Traffic safety is a challenging area that needs to be studied using scientific methods. The understanding and prediction of MVTCs are key for social engineering in general and urban planning in particular. The worked-out methodology for analyzing and visualizing MVTC data will contribute to a better understanding of the traffic in major cities and give the authorities valuable insights.

The study resulted in a semi-independent MVTC heatmap that was developed to track the locations of motor vehicle collisions and their intensity in a particular area of New York City with a time lag of 1–2 days.

With these data, the road agencies of the city can identify potentially dangerous road sections and make them safer for both drivers and pedestrians or issue public service announcements to be broadcast on the TV or the internet to raise public awareness and promote responsible and attentive driving.

The application of the proposed MVTC monitoring system is associated with the processing and transmission of the heatmap data directly to the emergency services of the city. In such emergencies, the appropriate services will be able to analyze the intensity of traffic collisions throughout the city and make rational decisions when they occur. Further lines of effort will be related to enhancing the efficiency of obtaining and processing data on the MVTCs in cities. On top of this, more types of data representation and visualization will be added—for example, hierarchical (group) visualization in order to enable the study of processes at different levels of detail (Faccin 2022). Hierarchical visualization has great potential for the research and online monitoring of multidimensional dynamic data.

Funding: This research received no external funding.

Data Availability Statement: Available online: <https://data.cityofnewyork.us/Public-Safety/Motor-Vehicle-Collisions-Crashes/h9gi-nx95>, accessed on 21 May 2021.

Conflicts of Interest: The author declares no conflict of interest.

Note

¹ Tufts Research University <https://www.tufts.edu/> Accessed on 30 May 2021.

References

- Albuquerque, Vitória, Ana Oliveira, Jorge Lourenço Barbosa, Rui Simão Rodrigues, Francisco Andrade, Miguel Sales Dias, and João Carlos Ferreira. 2021. Smart Cities: Data-Driven Solutions to Understand Disruptive Problems in Transportation—The Lisbon Case Study. *Energies* 14: 3044. [CrossRef]
- Almajyul, Aljawharah, and Nadia Al-Ghreimil. 2022. Traffic Accident Analyzer: A Visual Analytics Tool for Traffic Accidents Dataset. In *Proceedings of Sixth International Congress on Information and Communication Technology*. Lecture Notes in Networks and Systems. Singapore: Springer, Volume 235. [CrossRef]
- Bachechi, Chiara, Laura Po, and Federica Rollo. 2022. Big Data Analytics and Visualization in Traffic Monitoring. *Big Data Research* 27: 100292. [CrossRef]
- Berres, Anne S., Haowen Xu, Sarah A. Tennille, Joseph Severino, Srinath Ravulaparthi, and Jibonananda Sanyal. 2021. Explorative visualization for traffic safety using adaptive study areas. *Transportation Research Record* 2675: 51–69. [CrossRef]
- Bishop, Ian David. 2021. Analysis and visualization of temporal variation in visual impacts. *Landscape and Urban Planning* 210: 104068. [CrossRef]
- Borland, David, and Russell M. Taylor. 2007. Rainbow Color Map (Still) Considered Harmful. *IEEE Computer Graphics and Applications* 27: 14–17. Available online: <https://www.ncbi.nlm.nih.gov/pubmed/17388198> (accessed on 15 December 2021). [CrossRef]
- Brink, Henrik, Joseph Richards, and Mark Fetherolf. 2016. *Real-World Machine Learning*. Shelter Island: Manning Publications, 427p, ISBN 9781617291920.
- Butsenko, Elena. 2015. Improving the model of investment planning on the basis of network modelling. *Upravlenets* 1: 38–42.
- Chen, Chengjie, Hao Chen, Yi Zhang, Hannah R. Thomas, Margaret H. Frank, Yehua He, and Rui Xia. 2020. TBtools: An integrative toolkit developed for interactive analyses of big biological data. *Molecular Plant* 13: 1194–202. [CrossRef] [PubMed]
- Clarival, Antoine, and Bruno Dumas. 2021. Intra-City Traffic Data Visualization: A Systematic Literature Review. *IEEE Transactions on Intelligent Transportation Systems* 1–18. [CrossRef]
- Faccin, Eduardo. 2022. *Vernier, Visualization of Dynamic Multidimensional and Hierarchical Datasets*. Groningen: University of Groningen. [CrossRef]
- Friendly Michael, Howard Wainer. 2021. *A History of Data Visualization and Graphic Communication*. Cambridge: Harvard University Press.
- Harris, Charles R., K. Jarrod Millman, Stéfan J. Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel Smith, and et al. 2020. Array programming with NumPy. *Nature* 585: 357–62. [CrossRef] [PubMed]
- Kirsh, Ilan. 2021. *Visualizing Web Users' Attention to Text with Selection Heatmaps*. Cham: Springer, pp. 517–20.
- Lemenkova, Polina. 2020. Python libraries matplotlib, seaborn and pandas for visualization geospatial datasets generated by QGIS. *Analele stiintifice ale Universitatii "Alexandru Ioan Cuza" din Iasi-seria Geografie* 64: 13–32.

- Lienemann, Pat, Luwis A.R.R. Andradi, Nat Gustafson-Sundell, and Evan Rusch. 2020. Automating Collection Analysis Data Visualization in Jupyter Notebook: What's Possible and Why Would You Do It. Paper Presented at the Electronic Resources & Libraries (ER&L), Austin, TX, USA, March 9.
- Liu, Xiao, Kun Li, and Haocheng Xu. 2022. Analysis and Visualization of Accidents Severity Based on Lightgbm-Tpe. Available online: <https://ssrn.com/abstract=4024035> (accessed on 18 February 2022).
- Lu, Min, Joel Lanir, Chufeng Wang, Yucong Yao, Wen Zhang, Oliver Deussen, and Hui Huang. 2021. Modeling Just Noticeable Differences in Charts. *IEEE Transactions on Visualization and Computer Graphics* 28: 718–26. [CrossRef] [PubMed]
- McKinney, Wes. 2017. *Python for Data Analysis*, 2nd ed. Sebastopol: O'Reilly Media, Inc.
- NYC Open Data. 2021. Motor Vehicle Collisions—Crashes. Available online: <https://data.cityofnewyork.us/Public-Safety/Motor-Vehicle-Collisions-Crashes/h9gi-nx95> (accessed on 21 May 2021).
- Rowell, Katherine, Lindsay Betzendahl, and Cambria Brown. 2020. *Visualizing Health and Healthcare Data: Creating Clear and Compelling Visualizations to "See How You're Doing"*. Hoboken: Wiley, p. 99. ISBN 9781119680864.
- Sapre Atharva, Shubham Vartak. 2020. Scientific Computing and Data Analysis using NumPy and Pandas. *International Research Journal of Engineering and Technology* 7: 1334–46.
- Shorikov Andrey, Elena Butsenko. 2021. Optimization of adaptive network economic and mathematical modeling of business planning process control. *Economics and Mathematical Methods* 57: 110–25. [CrossRef]
- Yu, Cheng-Sheng, Chang-Hsien Lin, Yu-Jiun Lin, Shiyng-Yu Lin, Sen-Te Wang, Jenny L Wu, Ming-Hui Tsai, and Shy-Shin Chang. 2020. Clustering heatmap for visualizing and exploring complex and high-dimensional data related to chronic kidney disease. *Journal of Clinical Medicine* 9: 403. [CrossRef] [PubMed]