A Method to Identify Black Spot Candidates in Built-up Areas

Sándor Szénási\textsuperscript{a}, Domokos Jankó\textsuperscript{b}

\textsuperscript{a}John von Neumann Faculty of Informatics, Óbuda University, 96/B Bécsi út, H-1034 Budapest, Hungary
\textsuperscript{b}Road Safety Engineering Bureau, 54/B Esztergomi út, H-1138 Budapest, Hungary

Abstract

One of the most important road safety interventions is the elimination of accident black spots (places with accident density higher than expected). The first step of this process is the identification of these locations. There are several methods for this purpose, but most of them are now outdated. The primary objective of our five-year project was to adopt data collection and analysis methods according to the related developments of the last decades: 1) the spreading of GPS technology 2) the appearance of tablets and smartphones, and 3) the existence of critical amounts of accidental data (enough for data mining approaches). This paper presents the final results of our examinations. We prepared several methods, regulations and computer software to integrate these advancements into the daily routine. From the year 2011, Hungarian police officers have used smartphones to record all data about accidents while on the scene, including the GPS coordinates, which leads to more accurate and reliable location information. This allowed us to develop a novel black spot searching algorithm (inspired by some data mining techniques), based on the GPS coordinates of accidents. We have implemented it as a web service, and the practical experience shows that it performs very well, especially in built-up areas. Compared to the sliding window method, the site consistency is significantly higher, while the method consistency and the rank consistency are quite similar.

Keywords: accident black spots, built-up areas, GPS, data mining, DBSCAN
1. Introduction

The designation of public road accident black spots is often used in the specialty field of road-safety without exactly defining its meaning. Various definitions are spread [1, 2, 3], which will not be discussed here. The officially accepted definition in Hungary is contained in the Governmental Order on handling the public road infrastructure according to the needs of road safety. As per the definition mentioned, black spots are named as “spots of high accident risk”, which are “those sections of the public road network, on which the frequency of accidents with personal injury is over the national average compared to the volume of traffic concerned”. There are several steps when handling the actual accident black spots from a road-safety point of view. Literature provides detailed presentation of these steps [4, 5]. The present article is rather focused on the part of the process in which data is gathered regarding the localization and identification of the accident spots on existing road networks. The decision whether the spots found are really black spots will need further professional analysis (e.g. to consider traffic volume, national average and examining accident types, etc.). Based on the above, the spots with accident densification identified by the method as described in the article may only be named “black spot suspected” or “black spot candidates” (“prospective black spot”). The precise presentation of these however will definitely need further traffic-safety examinations. The general definition does not distinguish between roads within and outside of settlements, however experience shows that the features of accident densification spots in big cities are different from those black spots explored in external sections of national public roads. That difference is primarily shown in the search criteria, while the proposed algorithm itself is principally the same.

Between 2009 and 2013, a total of 80,863 road accidents with personal injuries took place in Hungary. 70% of all these happened in built-up areas. The majority of the accidents happened in towns, among which the total number of accidents in the five years happening in Budapest was 15,800 (that is 28% of all accidents within settlements). It cannot be exactly stated what percentage of all accidents happened in black spots; however, according to the authors’ estimations, based on practical studies, this value would be between 5% and 10%, depending on identification criteria. This means that by identifying the actual black spots and introducing appropriate counter-actions, the road safety situation may be significantly improved. 41% of the accidents with personal injury occurred within settlements on cross-roads, at road intersections.

In a broad sense, road safety management involves several steps: data collection, data cleaning, data filtering, localization of suspicious areas, evaluation of these areas, prediction of future conditions, prediction of the effects of possible actions, making a decision that a black spot candidate is a real black spot or not, monitoring the already identified black spots, etc. In this paper, we are dealing only with localization and evaluation (and marginally with the data collection) especially in case of accidents in built-up areas identified by GPS coordinates. There are several alternative definitions of hotspots [6, 7, 8]; therefore, we do not strictly adhere to any of these. With our method, we are trying to find areas where the accident density is higher than a given threshold value. There are several techniques working together and several articles trying to adopt these to special circumstances. One of the most widely used methods is the well-known sliding window technique; however, it has some disadvantages. We have developed a new data mining based algorithm to eliminate these problems.

The organization of the paper is as follows. In Section 2, we give an overview of the existing black spot searching methods, including the issues within built-up areas. Then, in Section 3, we introduce the improvements of the GPS based data collection process. Section 4 shows a novel black spot searching algorithm based on data mining methods and GPS coordinates. Finally, in
the last two sections we compare this new method to another well-known one (Section 5), and
draw some conclusions and suggest possible directions for future study.

2. Related work

The domestic public roads operating bodies (according to legislation) regularly carry out
black spot discovery on their own road systems as part of their attempt to improve road safety.
The expert working material [9] set up by the working committee of the Hungarian Association
of Road Matters gives directives to that activity. Based on the said directive, black spots are
searched for and discovered following the stipulated criteria, which differ between in cases of
road sections within and without (numbered road sections) built-up areas [10, 11]. For the iden-
tification of the accumulation points of accidents, the directive recommends the sliding window
technique [12, 6] to be used.

There are several versions of this approach. One employs accident location reference systems
based on section numbering and kilometre signposts. It uses a window with a specified width
and slides it through the examined road. In the first case, we place the starting position of the
window to the accidents of the examined road. We collect the window positions, where the
number of accidents inside the window passes a critical number. Another way is to slide the
window without taking into account the position of accidents. We move the window with a
given distance, and collect the positions where the number of accidents is higher than a threshold
value. As Elvik [6] showed, the sliding window method is commonly used in Europe; however,
the chosen parameters are quite different in different countries.

Apart from the fact that the method has certain disadvantages, it also cannot be directly
applied to the network of roads within a settlement. A big disadvantage of the original sliding
window technique is that it performs poorly in the case of intersections, where the accidents
are dispersed over several roads. A two-dimensional equivalent of the previous method is also
usable, where the window is a rectangle or circle (or any shape) with a given diagonal/diameter
and we are calculating the number of accidents in this boundary shape. This gives an easily
implementable method and easily interpretable results; however, it is hard to set the ideal shape
and size of the window. The method presented in this paper is similar to this but both the window
size and shape dynamically conform to the groups of nearby accidents (see Section 4.5).

There are several methods to evaluate the previously established site. In the case of di-

3
in our opinion it is harder to interpret these results for a given black spot due to the differences in
the weights of the similar accidents based merely on their distances from that spot. In contrast to
this approach, in our developed method the location of the black spot is not a strict spot, but rather
a polygon that is determined by the inner accidents themselves (a minimal spanning polygon).
This way, there are no distinguished points that would affect the weighting process. This results
in a clearer measurement number (density = accidents per square meters) that is much easier
to interpret and that can be more efficient if the area of the accidents is not circular. Naturally,
in some cases the KDE method of starting from a specific point is clearly advantageous: for
example it is easier to draw a density map that is clearer to see through. Also, it is noteworthy to
mention the current improvements that include the network-based solutions [13]. In general, our
developed method stands for an alternative solution of this approach using a novel technique.

The traditional micro-level crash prediction models and black spot searching techniques have
proven successful; however, these are traditionally reactive in nature [17]. Therefore, it is also
worth mentioning the related work on macro-level safety performance functions and hotspot
identification techniques [18]. The main difference is that macro-level methods investigate safety
issues on an area basis (traffic zones, cities, etc.); in contrast, micro-level methods are based on
urban or rural areas for roadway networks, and specific locations (freeways, curves, intersections,
roundabouts, etc.). Macrosopic methods are more efficient at integrating zone-level features into
blackspot prediction models and identifying hot zones in large study areas [19]. Where the scope of
the safety application covers a neighbourhood (traffic zone), it is worth it to use the macro-level
models [20].

The final decision about sites is also a topic of research. The separation of safe and unsafe
sites is wholly arbitrary [21]. A simple ranking method is the most straightforward in most cases.
This means that we sort the black spot candidates using a scoring function (based on the crash
rate described above). The candidates at the beginning and at the end of the list are often obvious
from the safety perspective, but it is usually not possible to give an exact threshold value to
separate the entire list into positive and negative parts. Experts should check the list from the
beginning to the end and investigate the candidates one-by-one using all available, additional
information.

3. Improvement of Data Collection Process

3.1. Black spot searching in built-up areas

Public roads constitute a network, the basic elements of which are road sections and junc-
tions. Both the road sections and junctions differ in their geometric characteristics and traffic
technological design within and outside of built-up areas, therefore identifying the place of an
accident can be different in these cases [22, 23]. For example, in Hungary, out of the built-up
areas, lead the numbered roads, where “kilometre + metre” sections are allocated. In contract,
the road network within built-up areas basically consists of named streets. The place of an acci-
dent is identified by “street name + house number”, while in junctions by “street name + street
name”. Unluckily, however, several problems occur in practice making the spot identification
process uncertain.

Traditional black spot searching techniques within built-up areas (not in numbered public
roads but in the network of named streets) cannot be used efficiently. While searching for black
spots, the distance of accident spots from one another has special importance, which is rather
more complicated to calculate within a built-up area. Traditionally, in case of two accidents
on the same road, the difference of the two road sections is considered as the distance. Street numbers however (especially when it comes to different streets) do not give such an evident and fast result.

The aim of our project was to create a methodology for black spot searching in this heterogeneous environment. Fortunately, the accident database is reliable and complete because it is maintained by the police, using a country-wide, uniform format. The usage of GPS coordinates eliminates the issues caused by the different place identification methods. Defining the GPS coordinates in an urban environment may be blocked or made more difficult by the dense built-in structure, high buildings, and other construction (over-passes etc.). Despite this, tests have been carried out in Budapest with no significant measuring difficulties experienced. Based on these experiments, we have developed an algorithm, which can search black spots in built-up areas, based on these GPS coordinates and some major accident data.

3.2. Available databases

In Hungary, data regarding accidents with personal injury are collected by the police, while the detailed (and official) database reported on the accidents are handled by the Central Statistics Department (KSH). The accuracy and completeness of the database are ensured by legislations, in spite of which there are obviously inaccuracies, and to a certain extent, shortage of data (“underreporting”). Participants in all public road accidents (in cases with personal injury involved) are obliged to report it to the police. The police officer on the spot will start collecting the relevant data after the location of the accident has been secured and the injured persons are taken care of. The police, within the 30 day period as per the legislation, will then send the collected and systematized data to the KSH, where the data will be checked, set out, and made available for both national and international organisations. Besides the KSH database, a lot more databases may also be used for gathering information. For example, Budapest provides some additional databases indirectly linked to accidents:

- The cleaning database of the public roads of the capital. Even in cases of minor accidents (with property damage only) it happens that certain cleaning tasks are raised on the public road management’s side during the elimination of the consequences of the accident. Data regarding such tasks will be collected in a database with indications on times, places and causes.

- BKV accidents database: The Budapest public transport company (BKV) runs their own database regarding the accidents with their own vehicles where personal injury, as well as property damage, was involved.

Situations where the same accident is recorded in several different databases at a time need to be handled. This can be coordinated by linking different data. This can be done automatically (when the place and time match in the electronic system) as well as manually. Accidents indicated this way will obviously be considered only once in the calculations, independent from the fact of how many different databases they actually appear.

Similarly to all other databases, the query of accuracy and completeness is raised. The individual databases cover different subsets of the complete set of accidents. Although the KSH database seems to be the most complete, it only includes the accidents with personal injuries, and only if the participants involved reported to the police. The other two databases, due to their purpose and set up, do not contain all the accidents but rather only those where cleaning
was needed or BKV vehicles were involved. However, the most information possible is needed when searching for black spots, and therefore, during the black spot search in Budapest, the three databases mentioned will be used together, in order to present and approach the actual situation. Nevertheless, the combined database is still incomplete, but the area distribution of the missing accidents is expected to be constant within the city. Therefore, while completely accurate, absolute figures cannot be calculated based on the databases, the data can already be the basis of comparison. The objective of black spot searching is to discover spots with a density of accidents higher than in the environment. For that purpose, the collective database is suitable.

3.3. Using GPS coordinates

The above-mentioned databases contain large amounts of data, among which it is expedient to highlight the three more important ones: location, time and seriousness of accidents. The most critical one of the three is considered to be the place, because from the searching perspective (especially within built-up areas) even 10 meters of distance difference may be critical. By spreading the GPS localization method, it has currently been possible to exactly pinpoint the spot of an accident. This appears in more and more countries of accident spot identification [24, 25, 26, 27]. In Hungary since 2011, accident investigator police officers have been storing the GPS coordinates in the accident statistic sheet.

The introduction of this method had been preceded by a long preparation process, with the first pilot project launched back in 2007. In one of the counties, investigator officers were provided with Magellan eXplorist 100 type GPS receivers, with which the GPS data of the site of accidents was recorded. Along with the significant improvement in accuracy, technical issues were also raised: since the officers had to make notes of the GPS coordinates manually, this often happened only with errors. The coordinates themselves can hardly be linked to a physical place by the sight of human eyes, therefore it often happened that certain digits were recorded incorrectly, which led to several hundreds of meters difference in distance in certain cases.

Following the impacts of the experiences, several software (supporting and controlling data entry) have been developed; however, the final solution was managing to completely eliminate manual data entry. This was realised in 2009 the first time with a Windows CE based software run in a PDA, which already stored the coordinates, as well as in a small-scale map gave feedback to the officer defining the exact position of the event.

The final solution for this problem (introduced nationwide in 2011) was an Android based software [28, 29] (made by the commission of KKK and also developed by the authors). Reaching the spot of an accident, using the menu system of the application, besides a number of other functions (e.g. sending TMC messages on the fact of the accident) it also becomes possible to save the GPS coordinates (Fig. 1a). This is followed by giving further data on identifying the spot (street number, section number, street, house number etc.), then by filling in the accident statistic data sheet (Fig. 1b–1c).

The above mentioned software means a number of advantages for those working on the spot of an accident. Spot identification has become much faster for it is enough just to wait for the valid GPS signal without necessarily measuring the distances from the nearest reference points. Quite luckily, it has also brought significant improvement in accuracy since the receiver of the GPS, based on practical experience, can generally provide the exact position of the operator with 5-10 meters accuracy.
4. Clustering based black-spot search

4.1. Basic clustering techniques

In recent decades, due to the rapid evolution of information technology and accumulation of huge amounts of data, the possibilities of data analysis are greatly expanded. Digital accident data can be stored in large data warehouses, and the size of these becomes enough to allow the usage of various data mining algorithms efficiently.

Data mining is a kind of database application which can be used to discover hidden correlations in the data [30]. It is multidisciplinary field, combining database management, statistics and artificial intelligence. Its important role is the automation of certain processes and expert activities. This helps in processing large amounts of data, which needs disproportionately large efforts by human power. Various data storage, query and statistical functions have long been used on accident data. But these have limitations, and a few novel ideas can help with some problems which cannot be solved with the conventional methods, for example, black spot searching in built-up areas, which is a clustering task from the data mining point of view.

Clustering means that we create groups (clusters) of objects as follows: all objects in the same cluster are similar to each other (according to a pre-specified criterion), and objects in different clusters are different from each other. Generally, we have to classify the points from an N-dimensional space into groups, so as to minimize the distance between the points within each group. This distance corresponds to the similarity of two points, which can be defined by various metrics [31]. In our case, these objects are the accidents and the similarity is based on the planar location of them.

4.2. Black spot clustering using modified DBSCAN

The DBSCAN algorithm (Density-Based Clustering of Application with Noise) is one of the most efficiently usable density-based methods [32]. This algorithm is first used for black spot searching in WEB-BAL [33] system and a modified version (fine-tuned to find black spots in built-up areas) appeared in FÖ-BAL application. The main principle of density-based clustering
is the following: the density of the elements within a cluster is significantly higher than between separate clusters. This is how the elements that make up a cluster and the outliers (elements outside of any clusters) can be distinguished.

The basic DBSCAN algorithm needs the definition of a distance concept to calculate the distance between two elements. For this purpose, we use the Euclidean distance of two accidents. The clustering distance between two accidents is the planar distance between the two geographic GPS coordinates of them (Section 4.5 shows an advanced density calculation method based on this). This new approach differs from the common methods, where the distance between two accidents is usually based on the difference of kilometre section numbers. In practice, this new method has several benefits: in case of in-built area, no kilometre section numbers are used, but house numbers; and these are not suitable for accurate distance determination. Furthermore, we can easily use these planar coordinates to measure the distance between two accidents which occurred in different streets [34].

The original DBSCAN method uses two parameters:

- $\epsilon$: A radius type variable (the unit of this parameter is metres).
- $\text{MinPTS}$: A lower limit for the number of elements in a cluster (the unit of this parameter is weighted accident number).

There are some definitions based on these parameters. The space within a radius of $\epsilon$ of an element is called the $\epsilon$-environment. If the $\epsilon$-environment of an element contains at least $\text{MinPTS}$ number of elements, this element is called an internal element. For a given domain of elements, one element is directly densely accessible from another internal element, if it is in the first $\epsilon$-environment of the first element. The definition of dense reachability is similar; it is permitted for one element to be accessible from another only through a chain of directly densely accessible elements. Two elements are densely connected if there is an element from which both are densely reachable with the given parameters. And finally, a density-based cluster is a domain of densely connected elements that shows maximum accessibility of density. Our goal is to find domains of accidents in the database in which all elements are densely connected and no further expansion is possible (based on the parameters specified).

The input of the DBSCAN algorithm is the set of all accidents and one starting accident from this set. This latter is considered as the initial region. We check all neighbouring accidents in its $\epsilon$-environment (Fig. 2a). If there are one or more accidents in this range, we choose the most promising of them (based on the expected score of the extended region, see Section 4.4) and extend the region with it (Fig. 2b). In the next and further iterations, we check the $\epsilon$-environment of this extended region and perform the expansion again if it is possible; where the $\epsilon$–environment of a region is the union of the $\epsilon$–environments of its elements (Fig. 2c). If it is not possible to select more accidents in the given state, the iteration ends. The final region (set of accidents) is a black spot candidate (Fig. 2d). Algorithm 1 shows this process (some details of the algorithm will be discussed later).

### 4.3. Spatial indexing

This is a computationally intensive method, and the most time-consuming part is to check the $\epsilon$–environment for each iteration of the loop. In this phase, we have to calculate the distance between the actual region and all accidents outside this. To solve this issue, we use a spatial indexing technique to reduce the computing needs. The basic idea of this solution is to split the whole problem space into smaller parts (called tiles). For this, we create a matrix (A) to store
The $\epsilon$ environment of first accident (the red one).

Extend the candidate with the most promising accident. Check the $\epsilon$ environment of the extended cluster (consists of the red accidents).

Extend the candidate with the next accident. There are enough points to calculate the area of the cluster (green region).

It is the last extension. There are not any new accidents in the $\epsilon$ environment.

*Figure 2:* Main steps of the modified DBSCAN algorithm.
Algorithm 1 Black Spot Candidate search

1: function DBSCAN(ACC : set of accidents)
2:    RES ← ∅
3:    for all start ∈ ACC do
4:        bsc ← {start}
5:        localMax ← {start}
6:        NEAC ← Neighbours(start)
7:        repeat
8:            best ← ∅
9:            for all x ∈ NEAC do
10:                if min \(d(y, x) \leq \epsilon\) then
11:                    if best = ∅ \(\lor\) Score(bsc ∪ best) < Score(bsc ∪ y) then
12:                        best ← y
13:                    end if
14:                end if
15:                bsc ← bsc ∪ best
16:                if Score(bsc) > Score(localMax) then
17:                    localMax ← bsc
18:                end if
19:            NEAC ← NEAC ∪ Neighbours(best)
20:        end for
21:        until best = ∅
22:    if Score(localMax) > MinSCR then
23:        RES ← RES ∪ localMax
24:    end if
25: end for
26: return RES
27: end function

Used variables and functions

- \(d(acc_1, acc_2)\): Euclidean distance between accidents \(acc_1\) and \(acc_2\).
- \(Neighbours(acc)\): The result is the accidents in the current and neighbouring tiles of the given accident (see Section 4.3).
- \(Score(bsp)\): Score of the given black spot candidate (see section 4.4).

Figure 3: Algorithm of Black Spot Candidate search.
these tiles, using the following dimensions (Eq. 1–2, where ACC is a set of accidents, $a$ is an accident, $a.Lon$ is the longitude of the accident, $a.Lat$ is the latitude, and $\epsilon$ is the parameter of the DBSCAN method):

$$\text{Number of columns} = \left\lceil \frac{\max_{a \in \text{ACC}} a.Lon - \min_{a \in \text{ACC}} a.Lon}{\epsilon} \right\rceil + 1$$ (1)

$$\text{Number of rows} = \left\lceil \frac{\max_{a \in \text{ACC}} a.Lat - \min_{a \in \text{ACC}} a.Lat}{\epsilon} \right\rceil + 1$$ (2)

In the next step, we distribute all accidents based on their location coordinates. Every X accident will be placed in the tile identified by row $R$ and column $C$, where (Eq. 3–4)

$$C = \left\lfloor \frac{X.Lon - \min_{a \in \text{ACC}} a.Lon}{\epsilon} \right\rfloor + 1$$ (3)

$$R = \left\lfloor \frac{X.Lat - \min_{a \in \text{ACC}} a.Lat}{\epsilon} \right\rfloor + 1$$ (4)

We can use this matrix effectively in the $\epsilon$-neighbouring check phase. When the algorithm collects the accidents in the $\epsilon$-neighbouring of the current region, it does not need to calculate the distance to all outsider items. It is enough to check the accidents in the tiles where the already selected elements are located, and in the neighbouring tiles of these.

4.4. The score function

We need a score function to evaluate the black spot candidates. This function plays an important role in the whole process, because the algorithm uses it to determine what to consider as a black spot and what not. The number of accidents cannot be used for this purpose, because it leads to a trivial result: all known accidents can be considered as one large black spot. Therefore, we should use some relative indicator. The goal of the black spot search method is to find areas with high local accident density. Therefore, it’s advisable to take into account the location of accidents and their relationship to each other.

The evaluation of black spots plays an important role in two cases:

1. Even during the search: the algorithm has to make a decision about which accident to select to expand the actual cluster. If there are multiple points within the distance of $\epsilon$, we have to examine all of them on the basis of the expected increase in cluster score.

2. At the end of the search: after a finite number of steps, there are not any new accidents to expand the actual cluster. In this phase, we have to make the decision, that the actual cluster is a black spot or not.

The second requirement often requires human assistance. Moreover, in some cases, experts cannot even make an unambiguous decision. For this reason, it is advisable to use a scoring system instead of the binary logic system (the cluster is a black spot or not). This score value shows how likely it is that the algorithm has found a real black spot.

Black spots are defined as an area where the density of accidents is high. In this case, density can be interpreted quite naturally: the ratio of the number of accidents and the area they occupy (Eq. 5).
It is worth using different weighting factors to calculate the loss of accidents [35]. Based on this, the method of calculating the density is the following (Eq. 6):

\[
WScore = \frac{\text{Weighted number of accident}}{\text{Area occupied by accidents}}
\]  

(6)

A variety of weighting factors can be used to calculate the value of accidents. For example, we may take into account the average estimated national economic losses for each injured person [36]. However, this is not a recommended calculation method in this case. These factors can significantly distort the results, because the weighting factor for a fatal accident can be up to 300 times larger than for a slightly serious accident. The weighting factors are usually determined by specialists (dealing with accident reason research) according to the main goals of the analysis. The score can be based on the number of accidents or on the number of injured persons.

Section 4.5 contains several considerations about the area designated by the accidents. Here, we only mention that it is worth introducing a minimum region size constraint. Practical tests showed that the score of a black spot candidate is disproportionately high in case of a few but very close accidents (this is because the divisor is too low in these cases). For this reason, it may be appropriate to introduce a minimum area constant (\textit{MinAREA}).

Accordingly, we use the following, fairly simple weighting method (Eq. 7–8):

\[
WScore_{\text{ACC}} = \frac{WA_f \times ACC_f + WA_s \times ACC_s + W\text{A}_I \times ACC_I + W\text{A}_p \times ACC_p}{\max(\text{Area occupied by accidents}, \text{MinAREA})}
\]  

(7)

\[
WScore_{\text{PERS}} = \frac{WP_f \times PERS_f + WP_s \times PERS_s + WP_I \times PERS_I}{\max(\text{Area occupied by accidents}, \text{MinAREA})}
\]  

(8)

Where:

\begin{itemize}
  \item \textit{WA_f} = Weight factor for accidents with fatal severity.
  \item \textit{ACC_f} = Number of accidents with fatal severity.
  \item \textit{WP_f} = Weight factor for killed persons.
  \item \textit{PERS_f} = Number of killed persons.
  \item ... 
\end{itemize}

The values of these weight parameters can specify what we are looking for. There are specific weighting factors for the accidents and casualties; this allows the user to specify which aspects he or she wants to take into consideration. These factors also make it possible to fine tune the black spot search parameters (for example, we can run a search that is specifically directed towards fatal accidents). It is also a common requirement that we want to work only with accidents with personal injury. We can handle all of these cases with selecting the values of some weighting factors to zero.

Considering injury severity in black spot evaluation gives different results than the original crash rate. The weighting can be based on the overall accident severity (which usually means the worst severity of all injuries of occupants) or on the severity of the injuries suffered by the participant persons one-by-one. Both methods can be used effectively; therefore, we assign weighting factors to all accident outcomes and personal injury severity.

In a literature review, the following weighting principles should be distinguished [37]:
• Same weight of all accidents (no weighting).
• Only the most severe accidents included.
• Weighting by number of vehicles.
• Weighting by accident type.
• Weighting by injured persons.
• Any combination of these.

To enable maximal flexibility, we suggest the last method, giving weight factors to all levels. Setting some factors as 0, or any relatively small number, can exclude the given accidents/participants from the examination. The goals of the search process should also influence the weighting: for example, if goals are set for fatal and severe injuries, the weights should be adjusted accordingly. There are several papers and manuals about suggested weighting numbers, like [35, 38, 39, 40] and there is a good overview about these in [37].

It is worth noting, that the usage of weighting factors can give a higher random variation of the results [39]. Higher differences in weights lead to higher randomness. It is also important to note that the density value of the black spot candidates depends on the weighting factors. Changes on the weighting numbers imply the need of density threshold alignment.

Thus, the algorithm uses the above formula during the clustering process (which direction do we want to exclude the cluster?); and after it (is the black spot candidate a real black spot?). To fulfil the latter, we need to introduce a threshold value for the minimal density ($MinSCR$). We consider candidates as real black spots, where (Eq. 9)

$$Score \geq MinSCR$$

Unfortunately, we cannot determine an exact value for this constant, which would be used universally in all examinations. On the one hand, this is because the density significantly depends on the number of accidents. For example, examination of the last 10 years gives different results than of the last two years. Perhaps, we might try to develop some scaling method to map these score values into a uniform range, but complex accident filters (for example, the user wants to work only with accidents caused by young drivers in the weekends) would make it too difficult and unreliable. One the other hand, there is not a sharp border between black spot and non-black spot clusters. Between the two, there is a very wide subjective area [41]. For this reason, we always let the user to choose the correct $MinSCR$ value (to indirectly decide that a black spot candidate is acceptable or not).

Fortunately, the absence of the universal threshold value does not pose a problem in practice. We cannot clearly define the score value, above which we consider the clusters as black spots (and vice versa). However, we can use the score values to compare the clusters. We can sort all the candidates by this score; the result is a list, where the black spots with higher probability are at the beginning of the list. This allows the user to give a score limit, which selects the candidates which he or she wants to deal with in the further steps.

4.5. Area calculation method

For the score calculation, a divider is needed, which represents how far from each other the selected accidents are. In the case of conventional black spot search methods, this is easy
to implement. These methods usually examine only one road, and in this case, the difference between section numbers can be used as the distance between accidents. This method is not suitable in the case of accidents in built-up areas. In the one hand, these accidents have not got road number and section number attributes. There are only street name and house number attributes, but these are not suitable for fast and accurate distance calculation. We might operate with the difference of house numbers, but it is hard to implement a reliable distance calculation method based on this. On the other hand, black spots are usually located near intersections. In these cases, the accidents belonging together have been assigned to different streets. This can make the distance estimation more complex and inaccurate.

Our recommended method is based on the area of the shape spanned by the accidents. In this case, accidents are considered as a point cloud, using the two-dimensional location of them (for this purpose, we can use the GPS coordinates directly). Operations with GPS coordinates (distance calculation, etc.) are quite computationally intensive due to the spheroid base of this system. Therefore, we convert all coordinates from the WGS’84 system (used by most GPS devices) to the EOV system. The EOV (Uniform National Projection system) is a plane projection system used for the Hungarian civilian base maps and for spatial informatics. The latter is a conformal cylindrical projection in transversal position. The big advantage of this conversion is that we can simply calculate the distance between two points using the Pythagorean Theorem. In practice, we work using distance squares to speed-up the calculation by omitting the root. However, this is only an implementation nuance, so in this article we will continue to refer to the use of GPS coordinates and real distances.

The next step is to find the convex shape with minimal area, which contains all points (all of the points must be inside the shape or in the contour of the shape). Subsequently, we can evaluate the black spot candidates as follows (Eq. 10):

\[
Density = \frac{Number \ of \ accidents}{Area \ of \ minimal \ bounding \ shape}
\]  

(10)

The dimension of the result is accident/m². This is not the usual dimension, but it provides an easily interpretable metric. This is the two-dimensional equivalent of the “crash rate”. We can extend the score function with a normalization by the traffic volume; however, this information is not always available in cases of in-built areas. Our work primary focuses on these areas; therefore, we use the basic crash rate equivalent.

We have to approximate the area of the black spot using a minimum polygon. To calculate the area of a convex polygon, we can use the Shoelace formula (also known as Gauss’s area formula). This requires a specific sequence of two-dimensional coordinates. As is visible in Fig. 4, the order of the points according to the clockwise direction. We have to search accidents inside this polygon and accidents which represent the corner points. Due to the effective running time, our modified DBSCAN algorithm continuously builds this polygon. After every new point addition, it checks its location. If the new point is already inside the shape, there is no additional operation to do. Otherwise, we have to modify the boundaries of the minimum polygon. Simultaneously, it is advisable to calculate the area of the shape continuously too. Each time a new point (\(P_{new}\)) is added to the cluster, the following steps must be performed:

- In the case of clusters with one or two accidents, the area concept cannot be interpreted. We just save these points as the corner points of the polygon. The area equals to the minimal area given by a constant parameter (\(MinAREA\)). This is not a real issue, because we usually do not want to find clusters with one or two accidents.
• For clusters with three items, the area of the cluster is equal to the area of the triangle spanned by the three points. We have to ensure that the order of points is applicable for further processing (clockwise direction). The \( P_{\text{new}} \) point must be on the right side of the vector connecting the first and second points. We can check this using a scalar multiplication. If the order of the points is not appropriate, we reverse the vectors to ensure this constraint.

• In the case of every additional point (4\textsuperscript{th}, 5\textsuperscript{th}, ...), we have to check that the point is inside the actual minimum polygon or not. Based on this, we modify the polygon if it is necessary:
  
  - We have to check that the new point is on the right side of the boundary vectors of the actual polygon. If it is true for all vectors, the point is inside the polygon (Fig. 4a). In this case, we do not have to modify the shape.
  
  - If the point is outside of the current polygon, there must be some vectors with the point at the left side (Fig. 4b). Due to the convexity of the shape, it is easy to see that there will always be a \( k \) and \( l \), where \(( \, P_k, P_{k+1}, ..., P_{l-1}, P_l, P_l \) border points form a path, where the \( P_{\text{new}} \) is at the left of each vectors in this path, and at the right of each vectors out of this path. To solve this issue, we have to remove the \(( P_{k+1}, ..., P_{l-1} \) points from the border and add the new \( P_{\text{new}} \) point. This means that we replace the \(( P_k, P_{k+1}, ..., P_{l-1}, P_l \) path of the contour with the \(( P_k, P_{\text{new}}, P_l \) path. This step changes the area of the shape, so it must be recalculated. It is not necessary to recalculate the entire polygon area, but enough to subtract the area of the eliminated part (polygon spanned by \( P_k, P_{k+1}, ..., P_{l-1}, P_l \) points, see Fig. 4c); and add the area of the new part: the polygon spanned by the \(( P_k, P_{\text{new}}, P_l \) points (Fig. 4d). Fig. 3 and 4 show this process. We can use the shoelace formula (Eq. 11) to calculate the area of the polygon.

The previously mentioned shoelace formula is the following (Eq. 11):

\[
\text{Area} = \frac{1}{2} \left| \sum_{i=1}^{n-1} x_i y_{i+1} + x_n y_1 - \sum_{i=1}^{n-1} x_{i+1} y_i - x_1 y_n \right| = \\
\frac{1}{2} \left| x_1 y_2 + x_2 y_3 + ... + x_{n-1} y_n + x_n y_1 - x_2 y_1 - x_3 y_2 - ... - x_n y_{n-1} - x_1 y_n \right|
\]

where

- \( \text{Area} \): The area of the polygon.
- \( n \): The number of sides of the polygon.
- \( x_i, y_i \): The coordinates of the corners of the polygon \((i = 1, 2, ..., n)\)

Using the algorithm described above, the size of the cluster area is continuously known due to the running of the clustering algorithm. Based on this information, we can calculate the score of the black spot candidate.

4.6. Post processing

The result of the traditional DBSCAN algorithm is not satisfactory in our case, we need an additional post-processing phase, which consists of two main steps:

• We have to find the clustering state with maximum value.
The new (red) point is on the right side of the boundary vectors.

There are some vectors with the new point at the left side.

Subtract the polygon spanned by $P_k, P_{k+1}, P_{l-1}, P_l$ points.

Add the polygon spanned by the $(P_k, P_{\text{new}}, P_l)$ points.

Figure 4: Extending the area of the black spot candidate.
We have to eliminate duplications and overlapping candidates.

The main goal of the classical DBSCAN method is to get the largest set of items, according to the densely connected criteria. However, in the case of accident black spots, we would like to see only accidents that strongly belong to one group and this is often not the same. Fig. 5 shows an example for this. If we strictly follow the steps of the DBSCAN algorithm, all of the accidents should be seen as a big black spot (Fig. 5a). But in fact, there are two independent black spots and some less relevant accidents between them (we can call these noise, according to the terminology of data mining), as it is visible in Fig. 5b. To handle this additional criterion, the search should be supplemented to include a maximum score (see Section 4.4) monitor function. After every iteration, a fitness function is evaluated that reflects the accident density of the black spot. The process goes until the cluster reaches the maximum size (there are not any new accidents in the $\epsilon - environment$), and its result is the (final or intermediate) state where the maximum fitness was reached.

The result of the previously described algorithm is the density-based cluster built from the starting accident. Different starting points usually result in different clusters; therefore we have to run the clustering started from each accident. This additional step causes the result of the modified DBSCAN algorithm to consist of several overlapping black spot candidates (for instance, all execution started from the accidents of the same black spot candidate will probably return the same cluster). To give a clear and easily interpretable result, we have to eliminate this redundancy. The principle of this post-processing method is that it sorts the black spot candidates by the already calculated score value, placing the items with higher score to the beginning of the list. After this, it processes these items one-by-one, and removes all clusters that have any overlapping accidents with any clusters earlier in the list. The result of this additional filtering is a list of non-overlapping black spot candidates with the highest score values.

Several papers about comparison of hotspot identification methods suggest using some crash prediction model to estimate the number of expected accidents. The empirical Bayes estimation of the already presented values (crash rate) seems a good way for better black spot identification [42, 43]. Our future plan is to extend the already developed score function with this prediction model. Unfortunately, in the case of built-up areas (due to the lack of some information) we have to find a good definition for the “similar areas” needed by these methods.
5. Evaluation of the new method

It is very hard to evaluate a black spot searching algorithm, because there is not any exact definition for black spots. Because of this, it is not possible to clearly classify the black spot candidates into the common classes (true-positive, false-positive, true-negative, false-negative) to determine the accuracy. Chang and Washington [21] introduced three tests that have been used in various papers [44]. These are the following: the site consistency test, the method consistency test and the total rank differences test. We had to adopt these tests to our task, because the original tests are based on road intervals, but our results are based on areas. Each of these is described in this section. All of these tests are based on the comparisons of the same method in two different time periods.

5.1. Site consistency test

We assume that a site identified as a black spot during a time period $t_1$ should also reveal high risk in a subsequent time period $t_2$. The examined method is as consistent as many accidents occurred in the $t_2$ time period in the area of black spots identified by the accidents of the $t_1$ period (Eq. 12).

$$T_1 = \frac{\sum_{i=1}^{n} C_{i,t_2}}{\sum_{i=1}^{n} \sqrt{A_i}}$$

Where $n$ is the total number of black spots identified in time period $t_1$, $A_i$ is the area of $i^{th}$ hotspot identified in time period $t_1$, and $C_{i,t_2}$ is the number of accidents of time period $t_2$ in the location of the $i^{th}$ hotspot identified in time period $t_1$.

5.2. Method consistency test

We assume that a black spot area identified by the method in time period $t_1$ will also be identified as a hotspot by the same method in the subsequent time period $t_2$. The method is consistent, if the number of hotspots identified in both periods is great and the number of hotspots identified only in one of the periods is relatively small. We use the following formula (Eq. 13) to calculate the consistency:

$$T_2 = \frac{|B_{t_1,1}, B_{t_1,2}, \ldots, B_{t_1,n} \cap B_{t_2,1}, B_{t_2,2}, \ldots, B_{t_2,m}|}{|B_{t_1,1}, B_{t_1,2}, \ldots, B_{t_1,n} \triangle B_{t_2,1}, B_{t_2,2}, \ldots, B_{t_2,m}|}$$

Where $n$ is the number of the black spots identified in time period $t_1$, $m$ is the number of black spots in the time period $t_2$, $B_{t_1,i}$ is the $i^{th}$ hotspot identified in the time period $t_1$, $B_{t_2,i}$ is the $i^{th}$ black spot in the time period $t_2$, $|A|$ means the number of items in the set $A$, $A \cup B$ is the union of $A$ and $B$, and $A \triangle B$ is the symmetric difference of $A$ and $B$.

5.3. Rank difference test

We checked the total ranking differences of black spots identified in the two periods. The smaller the value, the more consistent the examined methods are.

$$T_3 = \sum_{i=1}^{n} |\text{Rank}(B_i, t_1) - \text{Rank}(B_i, t_2)|$$

Where $B$ is the set of black spots identified in both periods. $\text{Rank}(B_i, t)$ is the rank of the $B_i$ black spot in the time period $t$. 

18
5.4. Evaluation results

We chose the following parameters for the new DBSCAN based black spot searching method: $\epsilon=60$ m, MinSCR=0.004 weighted accident/m$^2$, MinPTS=3 accident, MinAREA=200 m$^2$. The accidents weights were: $WA_f = 10$, $WA_s = 3$, $WA_l = 1$, $WP_f = 0$, $WP_s = 0$, $WP_l = 0$. We compare the new method to the well-known sliding window method, where the size of the window (width and height) is 100 m, and the minimum number of accidents is 3. All black spot searching methods are sensitive to the parameter values. For example, in the case of the sliding window method, we can choose a relatively large window width to have less large black-spots, and choose a relatively small window width to have more small black-spots. There is no unambiguously recommended value. According to this, we try to select the parameters according to the following:

- In all tests, the result of the tested methods are quite similar.
- The number of identified black spots is suitable for further practical processing.

Table 1: Results of the comparison of the “spatial window” and the “DBSCAN based black spot clustering” methods. Precision is the ratio of the number of confirmed black spots (identified in both intervals) and the number of all black spots (identified at least in one of the intervals). Where $t_1 =$ accidents in Budapest from 2008 to 2009, $t_2 =$ accidents in Budapest from 2010 to 2011

<table>
<thead>
<tr>
<th>Value</th>
<th>Spatial window</th>
<th>DBSCAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black spots in interval $t_1$</td>
<td>141</td>
<td>130</td>
</tr>
<tr>
<td>Black spots in interval $t_2$</td>
<td>114</td>
<td>113</td>
</tr>
<tr>
<td>Black spot identified in both $t_1$ and $t_2$</td>
<td>169</td>
<td>163</td>
</tr>
<tr>
<td>Black spot identified in $t_1$ but not in $t_2$</td>
<td>53</td>
<td>49</td>
</tr>
<tr>
<td>Black spot identified in $t_2$ but not in $t_1$</td>
<td>33</td>
<td>31</td>
</tr>
<tr>
<td>Precision</td>
<td>0.6627</td>
<td>0.6708</td>
</tr>
<tr>
<td>$T_1$ test result (accident/m)</td>
<td>0.0023</td>
<td>0.2003</td>
</tr>
<tr>
<td>$T_2$ test result</td>
<td>0.4956</td>
<td>0.5046</td>
</tr>
<tr>
<td>$T_3$ test result</td>
<td>3417</td>
<td>3165</td>
</tr>
</tbody>
</table>

The detailed results are visible in Table 1 and Table 2. As we can see, in the case of $T_2$ and $T_3$, the results of both methods are similar. In the case of $T_3$, the DBSCAN based method is slightly better, but the difference is not significant. The method consistency check ($T_3$) shows that the new method is more consistent, and the difference between the two results is remarkable. Finally, it is worth noting that the $T_1$ results are significantly better in the case of the DBSCAN method. More or less, both methods find the same locations (however the new method is slightly better, as the $T_2$ and $T_3$ results show), but the designation of the black spot location is more precise in the latter case; this leads to higher accident density ($T_1$). This lower bounding area means lower noise level, which leads to more consistent results.

6. Conclusions

The primary objective of our project was to adopt new data collection and analysis methods in addition to the significant related developments of the last decades. These are as follows:
The spreading of GPS technology (due to decrease of the price of the receivers). This has allowed the objective and accurate location identification of accidents.

- The appearance of tablets and smartphones. Using these devices, accident data recording becomes easy and more reliable (especially the location information).

- The existence of critical amounts of accidental data. Nowadays, accidental data warehouses contain all information about road accidents from the last 10-15 years. This enables the usage of data mining techniques to find black spots.

In the last five-years, we have developed several methods, regulations and some computer software to integrate these advancements into the daily routine of road safety experts. The data collection process worked well for a few years; as a result, we have a well usable accidental database for further processing.

The final step of this project was the development of a novel black spot searching algorithm based on the GPS coordinates of accident locations. The algorithm is an adaptation of the classic DBSCAN data mining method. As it is visible in Section 5, we have implemented this algorithm as a web-based application, and the practical use has shown that it performs very well. It is usable outside built-up areas; however, the main step forward is noticeable inside built-up areas.

The identification of black spots is not the final step of the road safety investigations. We have to examine the accidents of the black spots one by one and determine the underlying causes. This is essential to make the right and cost-effective decisions. For this reason, our further development primary focuses on the deep analysis of black spots.

Acknowledgements

The authors would like to thank the Hungarian Transport Administration for providing accident and road characteristics data as well as the Centre of Budapest Transport providing maps and local accident databases. The content of the paper however does not necessarily express the views of these companies and the authors take full responsibility for any errors or omissions. The authors also wish to thank Gabriella Kovács, Tibor Mocsári, Gyula Orosz, Imre Pásti for
their constructive comments and suggestions. The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the paper.

References


