MOBILE LASER SCAN DATA FOR ROAD SURFACE DAMAGE DETECTION

Master Thesis Geoscience & Remote Sensing

B.B. van der Horst

TU Delft

MOBILE LASER SCAN DATA FOR ROAD SURFACE DAMAGE DETECTION

Master Thesis Geoscience & Remote Sensing

by

B.B. van der Horst

to obtain the degree of Master of Science in Applied Earth Sciences
at the Delft University of Technology,
to be defended publicly on Wednesday July 10, 2019 at 14:00.

Student number:	4256697	
Project duration:	September 3, 2018 – July 10, 2019	
Thesis committee:	Dr. R. C. Lindenbergh,	TU Delft, daily supervisor
	Dr. ir. C. C. J. M. Tiberius,	TU Delft, supervisor
	Ir. C. Kasbergen,	TU Delft, supervisor
	Dr. K. Anupam,	TU Delft, supervisor
	Ing. J. Assendelft,	Iv-Infra

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft

Abstract

Road damage detection is important for road safety and road maintenance planning. Road surface anomalies, like potholes, cracks and ravelling, affect driving conditions, such as driving comfort and safety, noise emission, load loss of trucks, increase of fuel consumption and traffic circulation. Localisation of these anomalies allows for targeted road maintenance, which contributes to the improvement of driver safety, comfort and the optimisation of road maintenance.

The current technique to detect road damage is that road inspectors determine road damage in road images. However, the results are susceptible to human subjectivity. An improvement on image based road damage detection is using LiDAR data, because the geometry of road damage is measured too. To mitigate the issue of human subjectivity, an automated method for road damage detection was developed for the profile laser scanner on the IV-Infra car. This laser scanner is mounted at the back of the vehicle so that its profile lines are perpendicular to the driving direction. The proposed method consists of: (I) feature extraction with a sliding window algorithm; (II) K-means clustering to create training data; (III) Random Forest classification and (IV) morphological operations to remove noise and identify larger damage patches.

This method was tested on an 800-meter long provincial road with different road defects and road types. Most occurring road damages are cracks, craquel and raveling. The results of this method were validated in two ways: using a road inspectors damage classification and a custom-made validation set based on orthophotos. An overall accuracy of 73% is achieved for the fully automated process. When training of the Random Forest was based on an improved, semi-automated training data, the overall accuracy was 58%, this gives visual clear results. This is explained by more noise are presented in the results based on the fully automatic method, which is overlapped with the coarse road inspector's data. Optical inspection shows that the semi-automated method identified almost all damages of the custom-made validation set, although a shift between the point cloud and the validation is found. Still, the method has some difficulties with detecting the transverse cracks. This problem can be solved by integrating the two other mounted laser scanners of the Iv-Car, but pre-processing is needed to organise the point cloud. Also, an improvement in georeferencing the validation data would help to optimise the method and training data. Nevertheless, promising results are achieved by this method.

Contents

1 Introduction

Road damage detection is important to determine road safety and road maintenance planning. The damage of the road surface, like potholes, cracks and ravelling, affects driving conditions such as driving comfort and safety and increases fuel consumption, traffic circulation and noise emission. Localisation of these damages can be used for targeted road management and maintenance, which contributes to an improvement of driver safety and comfort (Vittorio et al., 2014).

The traditional method for road condition surveying is that inspectors drive slowly on the road looking out for road surface damages and stop the vehicle when damage is found, do measurements on the damage and mark visually. This is dangerous, time-consuming and costly (Cheng and Miyojim, 1998; Yu et al., 2007). To improve the safety road inspectors evaluate road images. The results are, however, susceptible to human subjectivity.

Iv-Infra, a subsidiary of the Iv-Groep, has a mobile mapping system, shown in Figure 1.1, which is used on roads for various applications, like lamppost and road surface marking detection. A mobile mapping system is a (terrestrial) vehicle containing laser scanners and/or camera systems, (Vosselman and Maas, 2010). The mobile mapping system of Iv-Infra includes 3 laser scanners, 10 cameras for 360° photos, a GPS and an Inertial Measurement Unit (IMU). This thesis is an attempt to study the feasibility of using such a system for road damage detection. In this research a method for road damage detection is developed based on laser scan data of one of the three laser scanners of the car, a Z+F PROFILER 9012A. This laser scanner is mounted at the rear of the vehicle such that its profile lines are perpendicular to the driving direction. It measures the range and the intensity along the profile.

There are several advantages of such a system, for example no road closure for manual road inspection is needed, which increases safety and decreases costs. In addition it is an improvement on image base road damage detection, because with mobile laser scan (MLS) data a direct measurement of the geometry of the road damage is measured. With automated damage detection, there are not subjective differences in classification due to subjective judgement.

1.1. Research objectives

The main objective of this research is to use the mobile laser scanning data of the Z+F PROFILER 9012A, mounted on the rear of a car, for (automatic) road damage analysis. This leads to the research question:

- How to use the Iv-Infra mobile laser scan data for road damage detection?

To answer this question, the following sub-objectives were formulated:

1. Describe the characteristics of point cloud data from damaged and undamaged roads;
2. Develop a method for road damage analysis using mobile laser scan data;
3. Validate the quality of this method.

Figure 1.1: The mobile laser scan car of Iv-Infra include 3 laser scanners, 10 cameras for 360° photos, a GPS and an IMU system. The Z+F PROFILER 9012A is indicated with the red circle,the green and blue square indicates the Leica's P40 laser scanners. The black circle designates the GPS system and the 10 cameras for 360° photos.

1.1.1. Research questions

To meet the aforementioned objectives, the following research questions are formulated:

Sub-objective 1: Data analysis

- What is the accuracy of the laser scanner?
- What are the characteristics of undamaged pavement in point cloud data?
- What types of surface pavement anomalies and road unevenness (bumpiness) do exist and what are their characteristics in point cloud data?

Sub-objective 2: Method

- What algorithms can be used for damage detection?
  - Is it possible to make this algorithm automatic?
  - Is it possible to classify the different pavement anomalies?
  - Is it possible to use this method on different kind of roads like asphalt and brick roads?
- Which size of damage can be detected with this method?
- Is it possible to give a certainty value to the detected damages?

There are some requirements that the method must meet.

1. The method must be applicable on large data sets.
2. Implementation in the existing processing chain should be possible.
3. The processing of a new data set can be done in one night.

Sub-objective 3 : Validation

- What is the accuracy of this method and how can it be described?
- The distance between the profiles is dependent on the velocity of the car. Does this distance matter in the method? And can the other laser scanners which are installed on the car solve this problem?

1.2. Outline

The layout of this work is as follows. First, chapter 2 gives an overview of surface road damage and which techniques are already available for road surface damage detection. Also, this chapter contains a description about Iv's mobile laser scan car.

A detailed overview of the used laser scan data and research area will be given in chapter 3.

Chapter 4 explains the conceptual idea of the methodology for detecting road surface damage with a profile laser scanner. It describes in detail the different features which are used in the classification algorithm.

The results of the methodology are given in chapter 5. Also, the results are validated based on a road damage classification of an independent road inspector.

The conclusions of the research will be drawn in chapter 6. In this chapter recommendations for future research will be made as well.

2

Road damage detection

Several methods have been developed to collect data of a road surface and determine damage from such data. First, this chapter gives an overview of the types of road damage. Then, damage detection methods are provided. These methods can be classified based on how the road surface information is acquired. This can be vibration, image and laser scanning-based methods. As this study focuses on investigating the feasibility of laser scanning in detecting surface damages, a characterisation of it is presented, and then existing methods for damage detection are investigated. Further, this chapter describes the Mobile laser scan car of Iv-Infra, and gives a description of the used laser scanner. All this will be recapitulated in a small conclusion of this chapter.

2.1. Road anomalies

To distinguish several types of road anomalies the Long-Term Pavement Performance (LTPP) Program is used as guideline, (Miller and Bellinger, 2003). This project defines a common language to describe road surface damage which is widely used in different parts of the world. The LTPP categorises three types of pavements: asphalt concrete-surface, jointed Portland cement concrete and continuously reinforced Portland cement concrete, (Miller and Bellinger, 2003). Dutch asphalt is closest to the asphalt concrete-surface as defined in the LTPP Program. So the focus is on this type of asphalt.

According to the LTPP, asphalt damage is categorised into the following groups: cracking; patching; potholes; surface deformation; surface defects and miscellaneous distresses. In more detail:

- **Cracks** - can be formed due to traffic load or freezing of water in asphalt. Each type of crack is different in shape, position and origin. For this research crack shape is most relevant. The following types of cracks are distinguished:
  - *Longitudinal cracks* - are cracks parallel to the direction of the traffic;
  - *Transverse cracks* - are cracks perpendicular to the pavement centre line;
  - *Fatigue cracks* - are cracks which occur due to repeated traffic loads. They arises when small interconnected cracks develop into many-sided, sharp-angled pieces. These cracks are usually smaller than 30 cm on the longest side. In a later stage, they can look like a crocodile skin or to a chicken wire pattern.
  - *Block cracks* - are cracks that divide the pavement in an approximately rectangular pattern, where the cracks are parallel and perpendicular to each other. The blocks have sizes between 10 cm^2 and 10 m^2.
- **Potholes** - are bowl-shaped holes with various sizes involving one or more layers of the asphalt pavement structure. Size and depth increase due to water accumulation in the hole (Tedeschi and Benedetto, 2017). They arise due to freezing of water in the soil which results in expanding of the space. Thawing of the soil can weaken the road surface while traffic can break the pavement, which results in potholes.
- **Patches** - are pavement replacements after original construction greater than 10 cm^2. So the

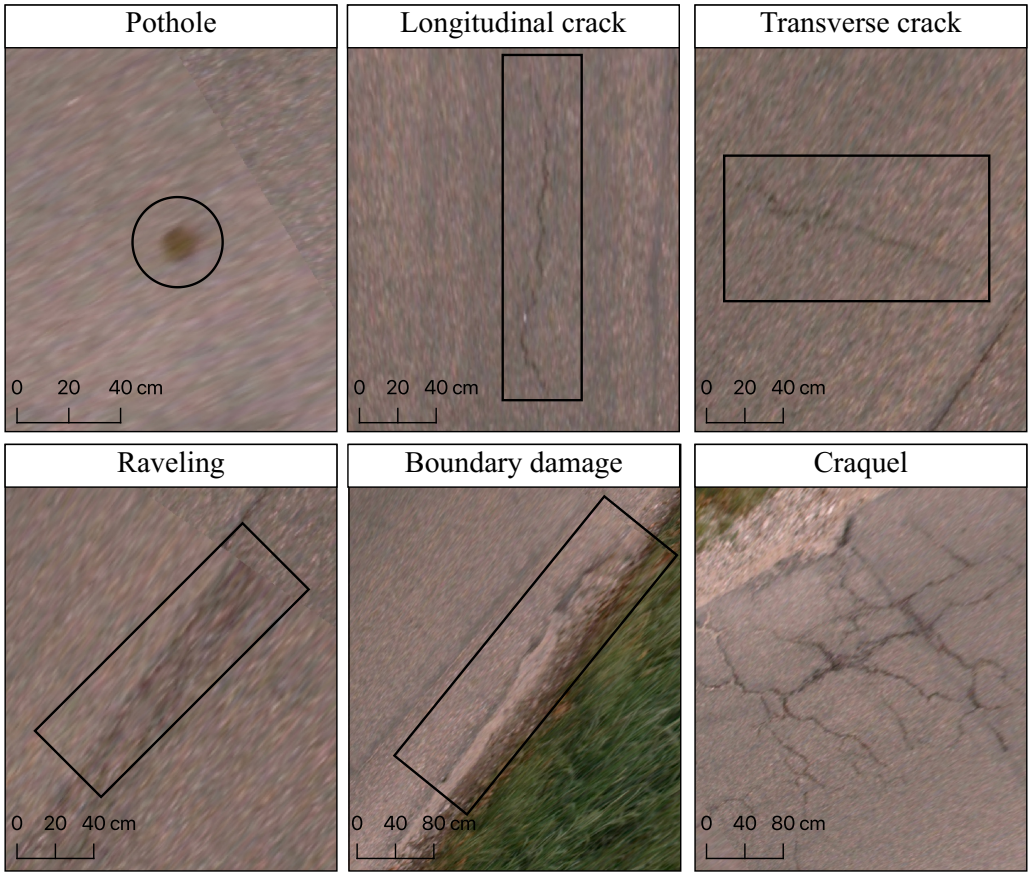


Figure 2.1: The six damage types in the research area found by a road inspector.

original construction is removed and replaced by new material. They can be recognised by different colour, texture and often have rectangular shapes.

- **Rutting** - is a longitudinal surface depression in the wheel path. The wheel path is lower than the surroundings due to traffic load.
- **Surface defects** are
  - *Bleeding* - is a film of bituminous material on the pavement surface that creates a shiny, glass like, reflecting surface that usually becomes quite sticky. Bleeding occurs during hot weather when bituminous material fills the aggregate voids and expands onto the pavement surface, (Pavement Tools Consortium, n.d.).
  - *Polished aggregate* - is the opposite of bleeding. It is the phenomenon where the surface binder is worn away to expose coarse aggregate.
  - *Ravelling* - are dislodging of aggregate particles due to influences of traffic, weather and obsolescence of the binder, (Kneepkens and Heesbeen, 2017; Tedeschi and Benedetto, 2017).

In this research the following damages are found on an asphalt test road following a road inspector: ravelling, potholes, boundary damage, craquel, longitudinal and transverse cracks. Craquel are cracks, which develop into many-sided, sharp angled pieces. This damage develops at the end of the structural life of an asphalt pavement, (Bouwend Nederland, 2016). Craquel at the outer 0.25 meter of the pavement is named as boundary damage. Examples of each type of damage can be found in Figure 2.1. Also there are road anomalies, which are no damage, like manhole covers, road marks and speed jumps.

Next, a survey of techniques for data capture and methods for processing data to determine road surface damage are presented.

2.2. Road surface damage detection techniques

There are several methods to detect road anomalies from manual inspections to fully automated methods, (Gavilán et al., 2011). Different kinds of sensors are used in previous work, in this section different techniques are described. First vibration measurement techniques are discussed, then some camera techniques are expound, to finish with laser scanning techniques.

2.2.1. Vibration based methods

Accelerometers, microphones and tire pressure sensors are used to measure vibrations caused by pavement elevation differences and roughness. Accelerometers in mobile phones can measure the relative movement of the car in three dimensions. Examples are the Pothole Patrol by Eriksson et al. (2008) and Wolfrine by Bhoraskar et al. (2012).

The Pothole Patrol by Eriksson et al. (2008) detects potholes with a simple 3-axis 380Hz accelerometer and GPS device. This kind of data collection is also done by Perttunen et al. (2011) who used a Nokia N95 smart phone with an embedded accelerometer and GPS. The processing approach is different here, while Eriksson et al. (2008) used filters and a simple machine-learning approach, Perttunen et al. (2011) combines the power spectrum and time domain data. Bhoraskar et al. (2012) used, with Wolverine, a smart phone for detecting congested traffic conditions and bumps on the road. Like Eriksson et al. (2008) they used machine leaning techniques to identify bump and breaking events, (Bhoraskar et al., 2012).

A disadvantage of this data acquisition method is that the relative movement of the car is only influenced by the small contact area between the road surface and the four tires. So only a small part, the wheel path, of the road surface can be analysed.

Another vibration acquisition method is the use of a microphone. Wang et al. (2015) used a microphone, located near the wheel, as one sensor of their multi-channel data of the Versatile Onboard Traffic Embedded Roaming Sensors (VOTERS). It is used as an addition to support other installed equipment, like optical and electromagnetic sensors.

RoADS from Seraj et al. (2016), SmartRoadSense from Alessandroni et al. (2014), and the automated sensing system of Vittorio et al. (2014) all use smart phones as well.

2.2.2. Image based methods

There are also methods collecting images from scanning, line-scan and video cameras of the road surface, which can be used for detecting the damage. An example is the automated detection system RoadCrack, created by the Australian Commonwealth Scientific and Industrial Research Organization (CSIRO, ca. 2000). This system is based on high speed cameras mounted underneath the vehicle. These cameras collect high resolution images of small patches of the pavement surface and they are consolidated into bigger images of half-metre intervals. CSIRO (ca. 2000) stated that the system can detect cracks in a millimetre order, while driving up to 105 kilometres per hour. This is done fully-automated with a combination of machine vision and artificial intelligence.

Another system based on a laser based imaging sub-system for shadow-free image collection is the Digital Highway Data Vehicle (DHDV) from Waylink (2015). They use their Automated Distress Analyzer (ADA) which produces crack maps in real time.

RoadCrack and DHDV are two commercial systems, which use cameras as one of their acquisition methods. There are several more commercial systems, most of which have not published details on their algorithm.

BAM Infra in combination with ICT Group created a technology to analyze automatic pavements. As acquisition method they use 360°digital images, combined with machine-learning as the detection method, (BAM Infra, 2018).

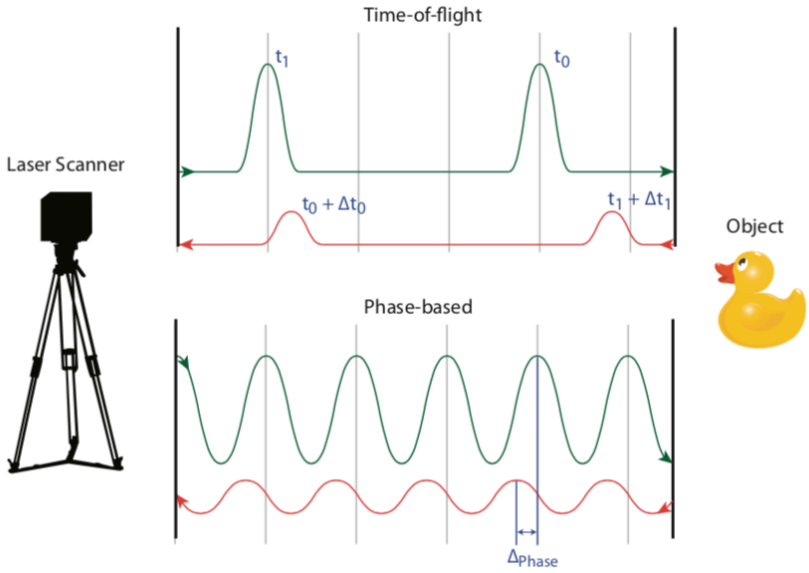


Figure 2.2: Distance measurement principle of time-of-flight (top) laser scanners and phase based laser scanners (bottom). The green signal is the emitted signal, while the red signal the received signal. In the time-of-flight a signal on t_0 and t_1 are emitted. The received signal is decreased in amplitude and the Δt_0 and Δt_1 are the time deference between transmission and receiving of the signal. With this time difference the distance to an object is determined. A phase-based laser scanner emits a continuous amplitude modulated wave and measured the phase shift of the reflected signal, Δ_{Phase}, to determine the distance to the object. Image copied with permission from Soudarissanane (2016).

2.2.3. Laser scan based methods

Methods which use Mobile laser scanning (MLS) data exist too. Laser scanning, also called LiDAR (Light Detection And Ranging), is an optical method to measure a 3D surface, (Vosselman and Maas, 2010). A laser scanner emits light signals and these signals are received again thanks to the reflection of an object. There are two measurement principles: time-of-flight laser scanners and phase based laser scanners. For time-of-flight, the time between the transmission and receiving signal is measured and the distance is calculated given the known speed of light:

$$D = 0.5 * \Delta t * c \tag{2.1}$$

where D = the distance to an object from the laser scanner,
Δt = the time between transmission and receiving the signal,
c = the speed of light, approximately 300 000 000 meters per second.

Phase based laser scanners emit a continuous amplitude modulated wave and determine the distance to an object by measuring the phase shift between the emitted signal and the received signal (Soudarissanane, 2016). Both principles are given in Figure 2.2.

Mobile mapping systems consist of one or more laser scanners, a position (GNSS) and an orientation (IMU) system. Frequently they carry digital cameras or other sensors as well. The photo data can be used to add colour information to the laser point data. The position and orientation systems are used for geo-referencing the point data, so they can be projected on the right position on a map. To link all the systems a synchronised time reference is used, (Vosselman and Maas, 2010). One of the advantages of using laser scanning sensors is that 3D topography of the road surface can be captured highly accurately and quickly.

Guan et al. (2014) used MSL data to detect road markings. From MSL data, they create intensity images, which they use in a point-density-dependent multi-threshold segmentation method to recognise road markings.

Pavemetric Inc. developed the Laser Crack Measurement System (LCMS), which consists of two high performance 3D laser profilers and a camera as detector, in cooperation with government and research

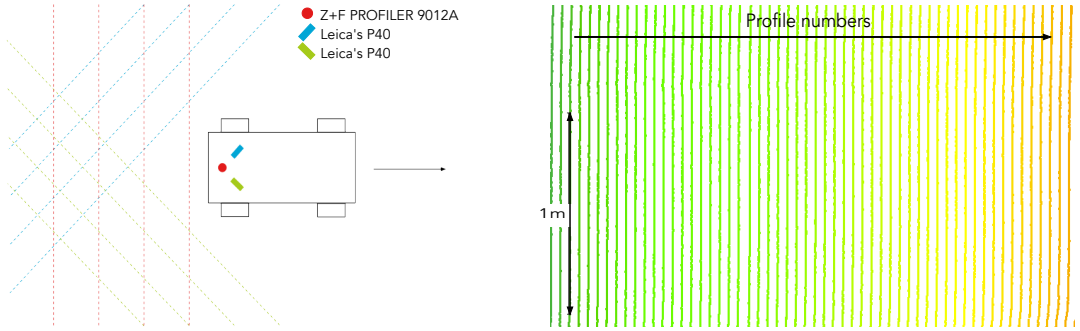


Figure 2.3: Left: the theoretic profile lines of the measurement car. In red the profile lines of the Z+F PROFILER 9012A, in blue and green the scan lines of the two Leica's P40.
Right: a small selection of the real profile lines of the measurement car. The points are coloured by profile number. The distance between two profile lines are around 4-5 cm. A regular pattern of distances is visible. These distances are dependent of the cars velocity.

partners (Laurent et al., 2014). This system measures range and intensities, and produces 2D and 3D data.

Yu et al. (2007) developed a system using a SICK LMS 200 laser scanner for reconstructing the 3D surface model, cracks in smaller regions can be identified from a variation of the 3D depth measurement.

Mertz (2011) used a low cost "laser line striper" to evaluate the unevenness of the road with a step-operator to detect road damage. Based on the number of the data points in one line, significant road damages are found. However, noise data can trigger the larger number of the points in the line, which lead to incorrect damage detection.

2.3. Iv's MLS car

The data collection is done by the Iv-Infra car, shown in figure 1.1. This car contains three laser scanners, two Leica P40's and one Z+F PROFILER® 9012A. In Figure 2.3 the theoretical and real laser scan profiles are given. The two Leica P40's are fixed in profiling mode and are mounted on an angle such that the profile lines intersect behind the measurement car. Furthermore 9 Blackfly® cameras, a GNSS and a IMU system that are mounted on top of the laser scanners, such that they are centered in the middle of the laser scanners. Eight of the nine Blackfly® cameras gives 360° photos of the surroundings and one of the cameras is directed to zenith to make a half sphere of photos. Also 3 HR cameras are mounted in the front bumper to make high resolution images of the road.

2.3.1. Z+F PROFILER 9012A

In this research the emphasis is on the Z+F PROFILER® 9012A, Figure 2.4. This is a profile scanner using the phase-shift method for measuring the range. An outgoing laser beam is intensity-modulated by a sine-wave signal. This signal is reflected by an object and the received intensity pattern is compared with the original transmitted signal. A phase-shift in the modulated signal is caused by the travelling time of light forth and back to the measured object. The phase measurement can be transformed directly into a distance/range,d:

$$d = \frac{c}{2 * f}, \tag{2.2}$$

where c gives the speed of light (with atmospheric corrections) in m/s,
and f the modulated frequency in Hz.

Accuracy of the Z+F PROFILER® 9012A

The Z+F PROFILER® 9012A has three measurement modes which differ in the spindle speed of the laser and amount of pixels per profile. A different spindle speed results in a different amount of pix-

Figure 2.4: Z+F PROFILER® 9012A Zoller + Fröhlich GmbH (2018)

Target distance	White (80%)	Gray (37%)	Black (14%)
1 sigma range noise, 2 m	0.2 mm	0.3 mm	0.4 mm
1 sigma range noise, 5 m	0.2 mm	0.3 mm	0.5 mm
1 sigma range noise, 10 m	0.2 mm	0.3 mm	0.5 mm
1 sigma range noise, 25 m	0.4 mm	0.6 mm	1.1 mm

Table 2.1: The accuracy depends on the colour of the reflecting object and the distance of this object to the scanner. A larger distance gives a larger range noise. Although the differences in range noise are minimal to ranges till 10 meter. (Zoller + Fröhlich GmbH, 2018)

els/points in one profile and the accuracy in range. In this research the 200Hz (12.000 rotations per minute) mode is used. This results in approximately 5120 points per profile. The corresponding noise factor is 2.8, (Zoller + Fröhlich GmbH, 2018). This noise factor changes the range accuracy. The nominal range accuracy varies with the distance and with the colour of the target, see Table 2.1. These numbers are determined at a 127 KHz data rate. In the case of the 200Hz mode the data rate is 1016 kHz and this results in a 2.8 times lower accuracy (Zoller + Fröhlich GmbH, 2018). So with the given values from Table 2.1 the sigma range noise varies between 0.56 mm and 3.08 mm.

Variation in angle

When the laser beam hits multiple “targets” of different heights, for example when the laser beam partly hits the road surface and partly falling into a crack (Figure 2.5), the laser scanner will detect a combination of multiple reflections, one for each target. Unfortunately phase-based ranging devices can never discern all the single vectors but only measure the resultant vector; the geometrical sum of all vectors. So the resultant range is a mixture of the distances to the surface and into the crack (Mettenleiter, 2019).

So the laser beam width defines which sizes of damages can be measured. A large beam is more likely to hits multiple “targets” which results in a resultant vector. The used laser scanner has a beam divergence of 0.5 mrad and it has a beam diameter of 1.9 mm at 0.1 m distance, (Zoller + Fröhlich GmbH, 2018).

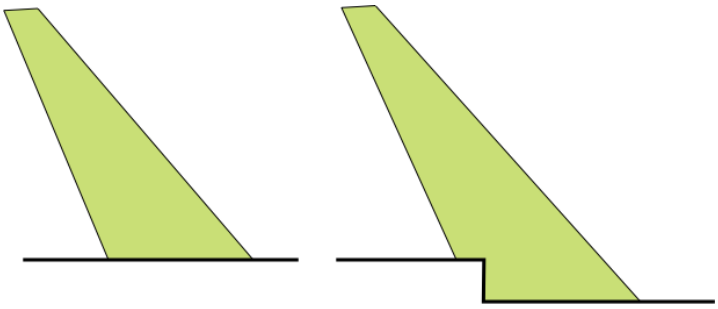


Figure 2.5: The left side shows a laser beam hitting a flat and horizontal surface, on the right side a laser beam hit an uneven surface (for example a part of a crack). The laser scanner detects a combination of multiple reflections in the laser beam.

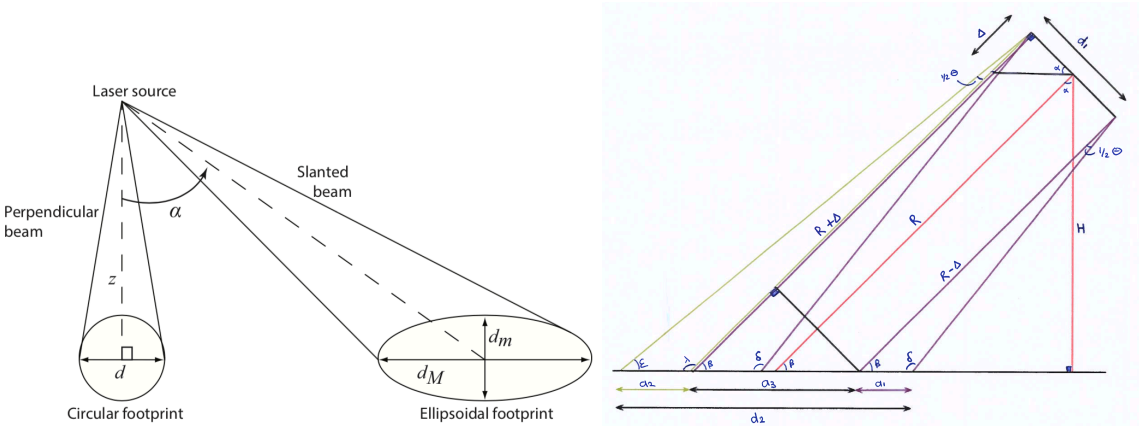


Figure 2.6: The technical shape of the laser beam.
Left: The shape of the footprint of a laser beam is circular with diameter d if it hits a perpendicular surface on distance z. The footprint change to an ellipsoidal footprint, defined by its major axis d_M and minor axis d_m if the beam hits a surface at an incidence angle α. Image copied with permission from Soudarissanane (2016).
Right: Increasing the angle α increased the beam width on the road. This is a technical draft of the approach of the divergenting laser beam. d1 is the beam diameter on a distance of 0.1 m, d2 is the beam diameter (the longest edge of the ellipse) on a horizontal surface. H gives the height of the laser scanner.

Beam width calculation

To calculate the theoretic beam width the sine rule and other trigonometric properties are used. The shape and a side view of a laser beam is given in Figure 2.6. In this Figure all the variables are given which are used in the beam width calculations.

First the part which is added or subtracted to the range, when the outsides of the laser beam are used, is calculated:

$$\Delta = \tan(\alpha) * \frac{1}{2} d_1 \tag{2.3}$$

Next, all the used angles, which depend on the changing angle α, are calculated:

$$\beta = 90° - \alpha\delta = 180° - \beta - \frac{1}{2}\theta\lambda = 180° - \beta\epsilon = 180° - \lambda - \frac{1}{2}\theta \tag{2.4}$$

The range, depends on the incidence angle α as well. When α increases the range with respect to a horizontal surface increases as well.

$$R = \frac{H}{\cos\alpha} \tag{2.5}$$

The road distance which is captured between $\alpha = 0$ and α can be calculated as:

$$\text{road distance} = H * \tan\alpha \tag{2.6}$$

To calculate the beam width on the horizontal surface, the laser beam is divided in three parts a_1, a_2, a_3, where a_1 and a_2 can be calculated with the sine rule and a_3 with trigonometry:

$$a_1 = \frac{R - \Delta}{\sin\delta} * \sin\frac{1}{2}\theta \tag{2.7}$$

$$a_2 = \frac{R + \Delta}{\sin\epsilon} * \sin\frac{1}{2}\theta \tag{2.8}$$

$$a_3 = \frac{d1}{\cos\alpha} \tag{2.9}$$

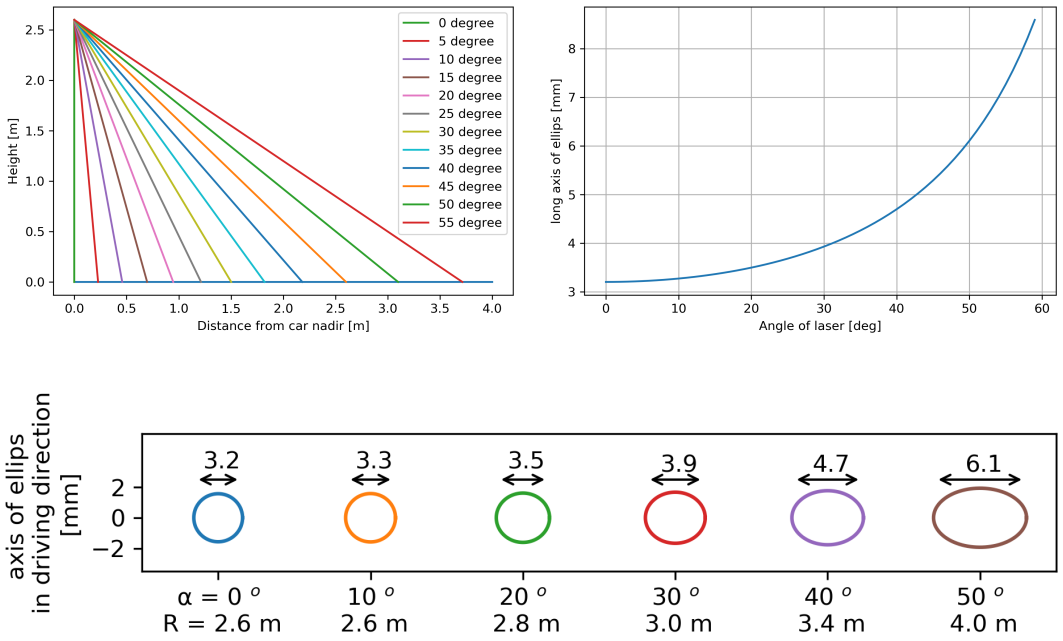


Figure 2.7: In the left top figure is seen how an increasing angle of incidence intersects with a horizontal and flat road on a larger distance from the car. The right top figure gives the theoretic beam diameter, d_M, when the beam hits a horizontal surface. And the bottom plot gives the footprint of the laser beam by an increasing angle of incidence. The footprint will be more ellipsoid increasing the angle. The diameter of the footprint is 3.2 mm on a distance of 2.6 meters (angle=0°). d_m increase from 3.2 mm (angle=0°) to 3.9 mm (angle=50°), while d_M (the axis of the ellipsoid in the transverse driving direction) increased from 3.2 mm (angle=0°) to 6.1 mm (angle=50°)

The beam width can be given by the sum of a_1, a_2, a_3:

$$d_M = d_2 = a_1 + a_2 + a_3 \tag{2.10}$$

In Figure 2.7 the effect on the beam width by the angle of incidence is given.

2.4. Conclusion

There are several types of road anomalies which can be found on an asphalt road. These are cracks, potholes, patches, rutting and surface defects like ravelling. But not every road anomaly is damage, like road marks, speed jumps and manhole covers. Although they have in common that they deviate in road flatness or in other words they have irregularities in the road surface.

To detect these road surface damages several techniques already exists. These techniques are divided by measurement technique, like vibration, image and laser scan based methods. An advantage of laser based methods is that the 3D topography of the road surface can be captured highly accurately and quickly.

In this research the Iv-Infra car is used for the road surface measurements. This car is all-round operable for detailed measurements. In this research the emphasis is on one of the three laser scanners, namely the Z+F PROFILER® 9012A. This is a profile laser scanner, which has a noise sigma that varies between 0.56 mm and 3.08 mm. The shape of the beam varies with the angle of incidence. An orthogonal falling laser beam gives a circular footprint, while a beam that hits the surface with an angle gives an ellipsoidal footprint. In this ellipsoidal footprint the beam width in the transverse driving direction varies for this mounting between 3.2 mm (0 °) and 6.1 mm (50 °). To determine road surface damage this beam width needs to be kept in mind, because with a larger beam width there is more change such that the laser scanner detects multiple reflections.

3

Mobile mapping data

This chapter gives an overview of the available data and the case study. It elaborates on 6 damage types and how they looks in laser scan data.

3.1. Road R106 near Haarlem - Netherlands

A road section of the R106 near Haarlem city, the Netherlands, was selected for a case study. It includes the Oudeweg and Penningsveer, together a length of 800 m. This is a quiet, touristic road where the driving speed is between 30 and 50 km/h. On this road, 36 road damages were found by a road inspector from a third party and were categorised as 2 ravelling; 7 craquelure; 2 potholes; 8 longitudinal and 11 transverse cracks and 6 boundary damages, (ARCADIS, 2019). This data is used as validation data. This road contains also speed bumps and a small part is brick-paved. Figure 3.1 shows the damage of the road as classified by a road inspector.

The laser scan data was provided by Iv-Infra, who measured the road on 17 October 2018 with their mobile laser scan car. It was a sunny day with a dry road.

The data was provided in 14 asc-files varying from 140 MB to 503 MB, 4.1 GB of point cloud data in total. The total number of points of these 14 files is around 37 million points. For each file number the amount of points are given in Table 3.1.

Figure 3.1: Research area, around 800 meters of the R106 near Haarlem, the Netherlands. On this road 36 damages were found by a road inspector. In this map, they are classified on damage type, and which shape (line, polygon or point) it was saved. Back ground map is from OpenStreetMap, (OpenStreetMap contributors, 2017)

File number	Number of points
1	1 235 152
2	2 950 313
3	2 989 903
4	3 335 533
5	2 320 074
6	2 958 343
7	2 111 276
8	2 148 947
9	2 162 486
10	4 095 931
11	4 430 715
12	3 229 729
13	1 522 918
14	1 745 274
total	**37 236 594**

Table 3.1: File number with the amount of points in that file. In total all files contains more than 37 million measurement points.

3.2. Available mobile laser scan data

The profile scanner produces measurement points with x, y, z coordinates. Each measurement point (x,y,z) is geo-referenced by the QINSy (Quality Integrated Navigation System) software (Quality Positioning Services B.V, 2018) such that the IMU, the GNSS locations, the vehicle odometer, the intensity and range are taken into account. This is done in the Dutch coordinate system, RD-coordinates. The z-component is given in Normaal Amsterdams Peil (NAP), the Dutch height reference. Each measurement point contains the following additional data fields: intensity, range, profile number and beam number. The intensity is the amount of reflected light, which has no clear unit. The range is the distance between the scanner and a reflection point on the object surface and is given in meters. A profile number is given to each new line which the profiler measures. A new profile starts nadir and the laser beam turns anticlockwise, see Figure 3.2. The beam numbers are given to each consecutive point in each profile. In this project, the laser scanner is configured such that each profile (360°) contains around 5100 points (beams), with a spindle speed of 200 rotations per second (profiles). When the car is driving, a spiral pattern is formed, illustrated in Figure 3.2. The distance between each profile depends on the car velocity and the spindle speed of the laser scanner. In this case, this results in a distance of 4 cm between the profiles while driving 30 km/h and 14 cm at 100 km/h. The point spacing along the profile is approximately 3 mm on the road in nadir direction and does not depend on driving velocity, but on incidence angle.

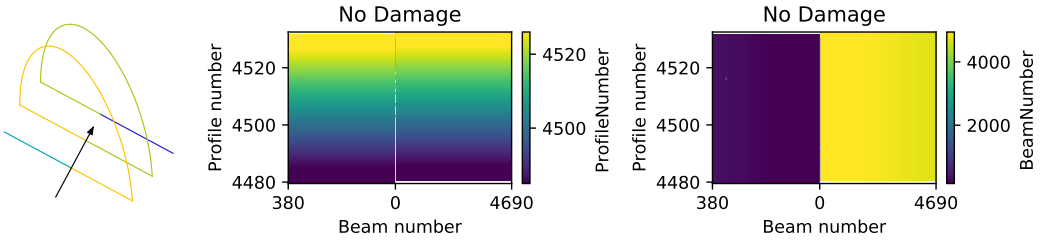


Figure 3.2: Scanning pattern of the laser scanner.
Left: the technical scanning pattern. Each new colour represents a new profile number. A new profile number starts nadir. The driving direction is marked with an arrow.
Middle: this plot represent the profile numbers for the ”No Damage” area. It includes profile numbers 4480 till 4530.
Right: the beam numbers of the ”No Damage” area. Each new profile number starts with a new range of beam numbers. For this damage set the dark blue colours represents the beam numbers from 0 till 380. The yellow colours represent the beam numbers 4690 till 5100. Beam number 5100 is the last beam number of the profile and is next to beam number 0.

Damage type	Profile numbers	Beam numbers
No damage (0)	4480 - 4531	4690 - 380
Craquel (1)	19420 - 19535	4800 - 810
Raveling (2)	18050 - 18155	4446 - 115
Boundary Damage (3)	18290 - 18435	4215 - 4900
L. Crack (4)	17960 - 18055	4470 - 340
T. Crack (5)	18055 - 18100	4625 - 830
Pothole (6)	19515 - 19540	4715 - 5075

Table 3.2: For each damage type one location is used for further research. In this table the location in beam numbers and profile numbers are given. The number next to the damage type corresponds with the location number in Figure 3.3.

Figure 3.3: The research area with the locations of the damage of Figure 2.1. Number 0 represent an area with no damage, 1: craquel, 2: raveling, 3: boundary damage, 4: longitudinal crack, 5: transverse cracks and 6: potholes.

3.3. Data selection

As mentioned in Section 2.3.1 the laser beam width varies with the angle of incidence. A large beam is more likely to hit multiple “targets” and produce false range measurements. But when taking the beam width smaller, the scanning area decreases as well. So a comparative assessment needs to be made between scanning area and beam width. With the knowledge that some damage is at the boundary of the road, 800 beams on each side of the nadir are used. This corresponds with an incidence angle of approximate 56 degrees. Then the beam width is around 7 mm at the sides of the road.

3.3.1. Cases

In Figure 2.1 six types of damage are given. Five damages are marked by a road inspector as damage. The pothole damage area which is used was not marked as damage, but is used because the other two potholes found by the road inspector are on the edge of the road outside the first 800 and last 800 beam numbers. In this project, these six locations are used to explore how damage looks in the input data. Also an area without damage is used. Both damage and non damage locations are given in Figure 3.3. Also they are given by beam numbers and profile numbers in Table 3.2.

3.4. Road damage in laser scan data

To distinguish road damage in laser scan data it is important to know what are the differences in input data of the damaged road relative to the non damaged road. In this section the six damage locations of Figure 2.1 it is discussed how they look in the mobile laser scan data.

3.4.1. Height measurements

In Figure 3.4a the height values of each point of the damage locations are given.

Raveling: The raveling arises on the connecting stripe and is located around beam number 4740. It shows up due the lower elevation, although it is difficult to see due the curved road.

Craquel: The craquel is clearly seen in the height data due the lower heights. Boundary damage (left in the plot) is seen due lower height values than the surroundings. A pothole (marked with a circle) is as well seen by lower elevations. Smaller cracks are visual seen, but they are not so clear.

Longitudinal crack: The longitudinal crack is located around beam number zero and from profile number 17960 till 18000. In the height value it has a lower height than the surrounding points. Although in this data section a curved surface is measured; on the higher beam numbers the surface is much lower than on the lower beam numbers. This difference is around 10 cm. Also another crack is seen, but this crack is not marked by the road inspector. In this section two asphalt sections are connected to each other and the connection location (around beam number 4760) a lower stripe is seen as well in the input data.

Boundary damage: The boundary damage is difficult to distinguish in the height data. This due the large height difference of the road boundary and the road itself. The verge is visual by higher elevations. Also a part of the data is not visible, because in pre-processing the data, the beam numbers between 800 and 4300 are removed from the data set.

Transverse crack: In the height input data the transverse crack is not clearly seen. Probably the transverse crack is just between two profiles and the input data did not contain a measurement of this data. Although in this section the road boundary (around beam number 700) is clearly seen due to the higher elevation. Also the connecting stripe is seen due to the lower elevation in comparison with the surrounding area.

Pothole: The pothole is located around profile number 19525 and has a lower height than the surrounding non damaged points.

No damage: The height of the no damage area is smooth. Although there is a small curve between the middle of the road and the boundaries. This curve in the road is made for the drainage of water.

3.4.2. Intensity measurements

In Figure 3.4b the intensity values of each point of the damage locations are given.

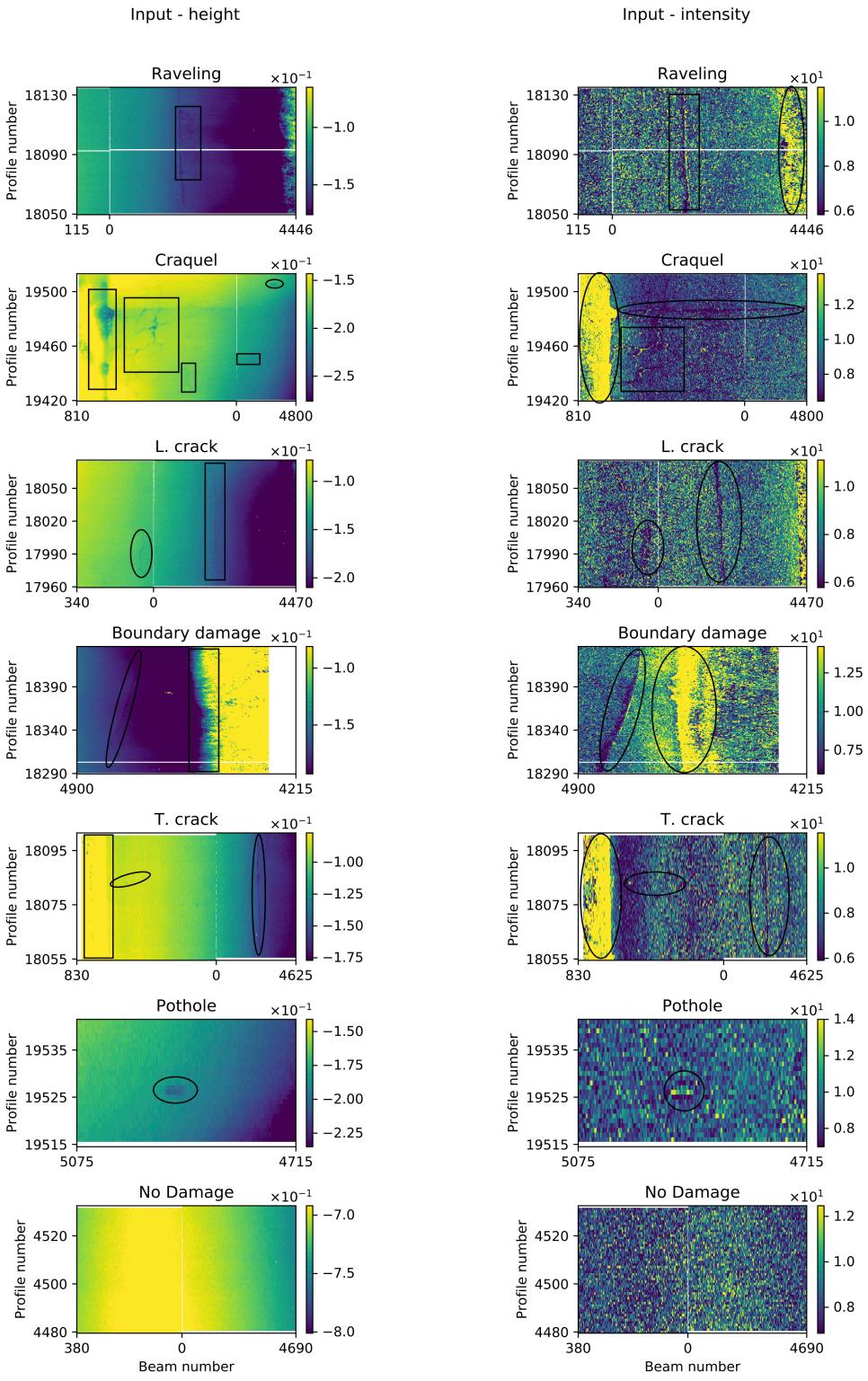
Raveling: The raveling is not so clearly visible in the intensity data as in the height data. Nevertheless there is a lower intensity in comparison with the surrounding area. On the right side of this data set (high beam numbers) the roadside is visible due a higher intensity than the road.

Craquel: The craquel in this set is clearly visible by the intensities values. The boundary of the road is well visible due the high intensity values. Also in this area a transverse stripe is visible by a lower intensity. This is a line where two types of asphalts meet each other.

Longitudinal crack: The longitudinal crack is not clearly visible in the intensity data. Although the connecting stripe between the two parts of asphalt is on the other hand clearly visible due a significant lower intensity.

Boundary damage: Although in the height value the boundary damage was not so clearly given, in the intensity values it is. In this data, the road side gives a higher intensity, while the asphalt of the road gives a lower intensity. It is clear that the boundary between the asphalt and the roadside is not a straight line. Also in this data the connection stripe is also visible due the lower intensity.

Transverse crack: The transverse crack is not visible in the intensity data, just like in the height data. In this data set the roadside has higher intensity values in comparison with the road. This is probably caused by the colour differences of the green grass and white-yellow sand in the roadside and the dark gray asphalt of the road.



(a) The height measurements (b) The intensity measurements

Figure 3.4: Left the height values and right the intensity values for each damage of Figure 2.1.

Pothole: There are less intensity changes where the pothole occurs. In this set the intensity is more or less the same.

No damage: In the no damage area the intensity values are equally distributed with an emphasis on the relative lower intensities values.

3.5. Conclusion

For this research mobile laser scan data of the Z+F PROFILER® 9012A laser scanner is used. This laser scanner measured intensity and ranges. This measurements are obtained for a x,y,z point. This point has a beam number and profile number as well.

Because of the use of a profile laser scanner, the point density in beam direction is much higher than in profile direction (driving direction). This can be a problem in damage detection if a damage lies between two profiles. When driving faster the gaps between each profile increase, so the possibility that damage lies between two profiles will increase.

In this project, the emphasis is on seven smaller areas of the road. Each area includes a type of damage found by a road inspector. Also a no damage area is used. Damage locations can be found by a deviation in intensity and height in comparison with the surrounding road area. Undamaged pavement is smooth and has no deviations in height or intensity.

4

Methodology for road damage detection from profile data

To identify damage on the road surface from MLS data, the proposed workflow consists of (I) feature creation, (II) K-means clustering to create training data, (III) Random Forest classification and (IV) mathematical morphological operations to remove noise and identify larger damage patches.

In the first step, different features are created to characterise the road surface. Classification will be done with the help of these features, to distinguish “damaged” surfaces from “non damaged” surfaces. The first idea was to use Random Forest classification only and use the classification of the road inspector as training data. But then it was found that the road inspectors classification was too coarse to use them for training. So K-means clustering is used to create training data for the more robust Random Forest classification. The final steps are to remove points classified as damage but not connected to other damage points and connect damage points to form damage patches. This last step is done using mathematical morphological operations.

In this chapter each step will be described. The results of each step are shown in chapter 5.

4.1. Step I: Feature creation

Feature creation is done to extract the characteristics of the road surface. Road surface damage is distinguished from non damaged surroundings based on deviations in height and intensity.

4.1.1. Sliding window algorithm

Various independent features are made with a sliding window algorithm.

In a sliding window algorithm, calculations are only applied to the points within the current window. The window is “sliding” over the data points, see Figure 4.1. In this project, the window slides along the profile, see Figure 3.2. A sliding window algorithm was chosen because this algorithm is fast and scalable. Also data processing can be done in the order the data was acquired.

The window size is defined by length N, where N are the the number of points. In this project the results of the calculation of a window are assigned to the centre point of that window. The centre point of a window, y_c, is defined as:

$$y_c = \begin{cases} y(\frac{N}{2}+1) & \text{if } N \text{ is even} \\ y(\frac{(N+1)}{2}) & \text{if } N \text{ is odd} \end{cases}, \tag{4.1}$$

where y_c = value of the centre point of the window,
N = the number of points in a window.

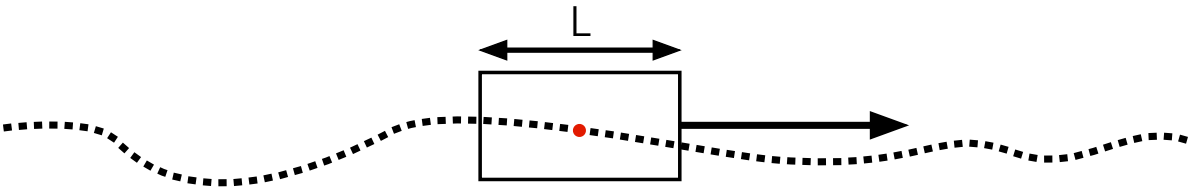


Figure 4.1: Sliding window principle. The points represent the data points in a profile. A window with length L slides along the profile in one direction (in this case L=15). In this project, the value of a feature is assigned to the centre point (red) of the window.

Note that when using this sliding window algorithm, only the centre points of a window receive a value for this function. For a window with length L it means the first and last L/2 points in a profile do not receive values, when evaluating one profile. When evaluating more profiles, the first L/2 points of the first profile and the last L/2 points in the last profile get NaN-values (Not a Number) representing the lack of values. The sequence of the points changes by pre-processing the data, such that the beam numbers run from roadside to middle of the road to roadside. In this case, L / 2 points from the road side mean nothing, because two road ends are connected to each other. Increasing the window length L results in more points with NaN-values at the roadside. This is shown in Figure 4.2.

An overview of the six different features and their calculations is given:

- Absolute deviation from the mean;
- Deviation from the mean;
- Deviation from a slope line;
- Standard deviation of the deviation of expected range;
- Standard deviation of profile point density;
- Sum of different window lengths.

4.1.2. (Absolute) deviation from the mean

A damaged road surface is distinguished from a non damaged surface based on deviation in elevation and intensity. To extract this deviation characteristics a derivative of the road surface roughness is used. Surface roughness is defined as the irregularities in the surface texture which are inherent to production process and wear (Taylor Hobson Limited, 2003). The arithmetic average roughness, R_a, is the most used roughness parameter. This is defined as the average of the absolute deviation from the mean line over a sampling length L, (Gadelmawla et al., 2002).

The surface roughness evaluates the roughness of a certain surface part with length L. In this project, the feature values are assigned to individual points. Therefore instead of surface roughness, the absolute distance from a centre point of a sliding window to the arithmetic average height of that window is assigned to the individual centre points.

In formula:

$$f_{1,i} = \mid y_c - \bar{x} \mid, \tag{4.2}$$

where $f_{1,i}$ = the absolute deviation from the window mean for the ith point in a profile,
$\bar{x} = \frac{1}{N} \sum_{i=1}^{N} y_i$, the mean value of the data within the window,
y_c = value of the centre point of the window.

Also the deviation from the centre point of a sliding window to the arithmetic average height of that window is used:

$$f_{2,i} = y_c - \bar{x}, \tag{4.3}$$

where $f_{2,i}$ = the deviation from the mean of window length L for the ith point in a profile,
$\bar{x}$ = the mean value of the data within the window,
y_c = value of the centre point of the window.

To estimate the mean, least-squares can be used. The least-squares principle offers a solution for a system of observation equations. Such system is typically redundant (more equations than unknowns:

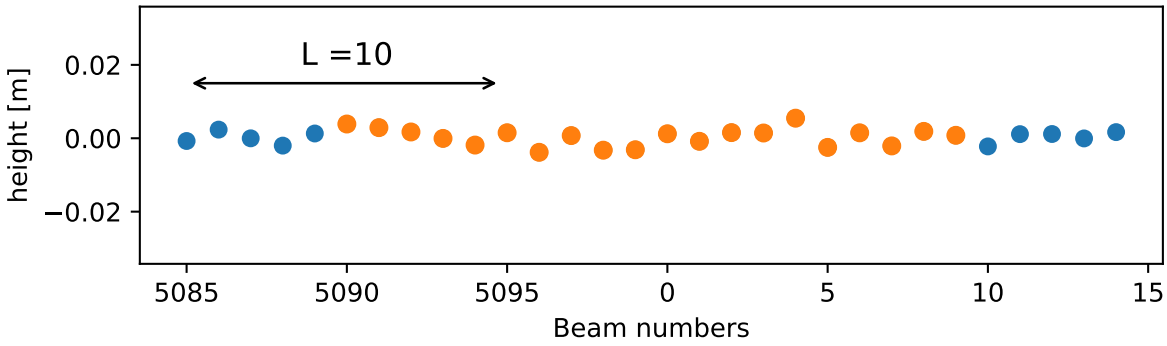


Figure 4.2: An example of a profile with 30 points. When a window with a window length of 10 points slides over this profile the blue first and last 5 points did not get a value, while the orange points got a value.

$m > n$) and inconsistent ($y \neq Ax$ due to measurement error) (Tiberius, 2015). The inconsistent system can be made consistent by introducing an $m \times 1$ error vector e :

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} a_{11} & \cdots & a_{1m} \\ a_{21} & \cdots & a_{2m} \\ \vdots & \ddots & \vdots \\ a_{N1} & \cdots & a_{Nm} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}}_{x} + \underbrace{\begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix}}_{e} \tag{4.4}$$

The least squares estimate minimises $e^T e = (y - Ax)^T (y - Ax)$, the sum of squared errors. Then the least square estimate of x, $\hat{x}$, is computed from y and A as:

$$\hat{x} = (A^T A)^{-1} \cdot A^T y \tag{4.5}$$

In the case of the arithmetic mean, the system is defined as:

$$\underbrace{\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}}_{A} \underbrace{[x]}_{x}. \tag{4.6}$$

Then the least squares solution is:

$$\hat{x} = (A^T A)^{-1} \cdot A^T y = \left(\begin{bmatrix} 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = \frac{1}{N} \sum_{i=1}^{N} y_i \tag{4.7}$$

Then $f_{1,i}$ is the absolute value of each least squares residual per measurement and $f_{2,i}$ the least squares residual. In this project, a built-in mean function of Python is used to make it computational efficient.

For both features the elevation of the road is irrelevant, as only the relative differences between a measurement point and the mean is used. Therefore, a road on a higher altitude with the same roughness gives the same values as a lower altitude road.

Both height and intensity values are used as input data for this feature. This results in 4 different features for fixed length L. The mean of a smaller window length L is more dependent to larger deviations than the mean of a larger window length. This results in that a single damage point is easier detected with a large window, than with a shorter. In Figure 4.3 the effect of different window sizes and locations is given.

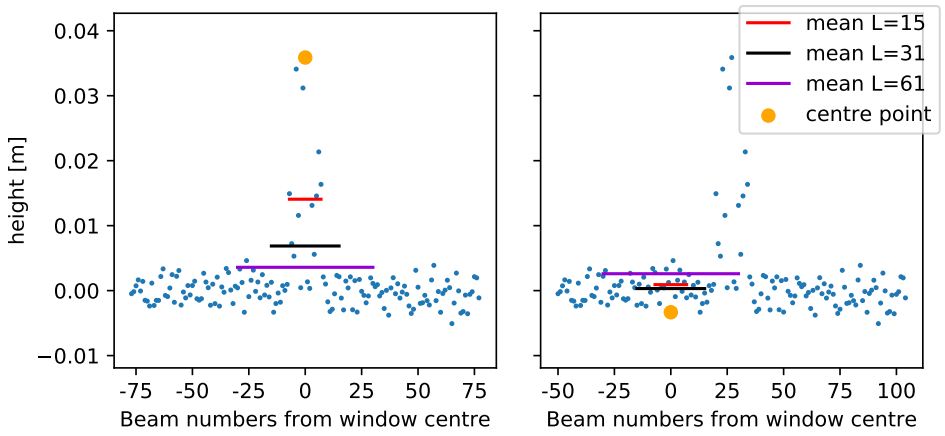


Figure 4.3: The effect on different window lengths on the mean.
Both plots contains the same data point, with in the centre data points which resembles to possible damage. The location and size of the sliding window has effect on the distance to the centre point.
In the left figure the windows are taken in the centre of the damage, and than the larger windows are less influenced by the damage points and the damage points are easier to detect.
In the right figure the window locations are shift to the non damage area. The large window are influenced by some damage points, while the smaller windows are not. This effects in possible larger distances to the larger window, although they are relative small.

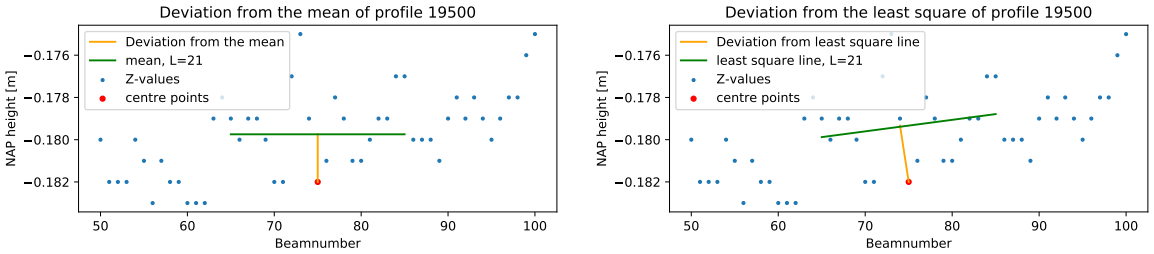


Figure 4.4: The features deviation from a linear least square fit and deviation from the mean. Left: An example of calculating the deviation from the mean of profile 19500. In red the centre point of the window (size 21) is given, in green the window mean and in orange the deviation from the mean to the centre point. Right: An example of calculating the deviation from a least square line of profile 19500. In red the centre point of the window (size 21) is given, in green the least square line and in orange the deviation from this line to the centre point.

4.1.3. Deviation from a (sloped) line

The idea behind this feature is very similar to the earlier described (absolute) deviation to the mean feature. The line fitting for this feature is again done using ordinary least squares. The idea is that when the road has a slope in the transverse driving direction (cant), the least squares line accommodates this slope when the window length is large enough.

The system for least squares line fitting is written as:

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} 1 & b_1 \\ \vdots & \vdots \\ 1 & b_m \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} h \\ \alpha \end{bmatrix}}_{x} \tag{4.8}$$

where b_i are the beam numbers,
y_i = surface value of the i^{th} point in an evaluation length,
α = slope of line in m/beam number
h = the intersection of the line with the y-axis.

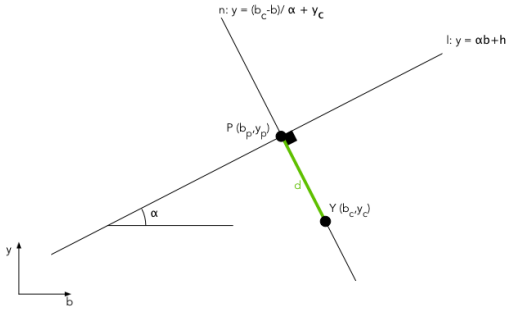


Figure 4.5: The orthogonal distance from the centre point of a window to the slope line is d.

Then the shortest distance from the centre point of the window to this line is taken. To calculate the shortest distance, which is the orthogonal to the line, of the centre point Y with coordinates (b_c, y_c) to a line the following steps are used. First, the normal of the line l: $y = \hat{\alpha}b + \hat{h}$ through point Y is given by line n:

$$y = \frac{b_c - b}{\hat{\alpha}} + y_c. \tag{4.9}$$

Then, the intersection of the two lines, the normal line n and slope line l gives intersection point P. This point P is the closest point on line l to the point Y.

$$\hat{\alpha}b + \hat{h} = \frac{b_c - b}{\hat{\alpha}} + y_c. \tag{4.10}$$

The b-coordinate of point P is given by:

$$b_p = \frac{b_c + \hat{\alpha}y_c - \hat{\alpha}\hat{h}}{\hat{\alpha}^2 + 1}. \tag{4.11}$$

The y-coordinate of point P is found by substituting the b-coordinate of point P in the original line l:

$$y_b = \hat{\alpha}(\frac{b_c + \hat{\alpha}y_c - \hat{\alpha}\hat{h}}{\hat{\alpha}^2 + 1}) + \hat{h}. \tag{4.12}$$

Then, the orthogonal distance between point P and Y is calculated by:

$$d(P,Y) = \sqrt{(b_P - b_Y)^2 + (y_P - y_Y)^2} = \frac{|\,\hat{h} + \hat{\alpha}b_c - y_c\,|}{\sqrt{1 + \hat{\alpha}^2}}. \tag{4.13}$$

where b are the beam numbers on the x-axis,
b_c is b-coordinate of the centre point of the window,
y_c = is y-coordinate of the centre point of the window,
$\hat{\alpha}$ = the slope of the least square line,
$\hat{h}$ = the intersection of the line with the y-axis,
b_P = is the b-coordinate of the closest point on line l yo point Y,
y_P = is the y-coordinate of the closest point on line l yo point Y.

In Figure 4.5 a small sketch is given with an example and the variables.

The orthogonal distance from the least square line to the centre point of that window is in fact the total least squares error,(Van Huffel and Vandewalle, 1991). With total least squares, the error is not only taken parallel to the y-axis (as done for ordinary least square fitting), but in b and y direction, to account for errors in both height and beam direction. In this feature a combination of the total least squares and ordinary least squares is taken. First an ordinary least square fitting is used, then the orthogonal distance is calculated. When the variances in x and y are equal (when the units of the axis are the same), this is an orthogonal line. Although it is unsure if the variances in the b and y are equal, it is

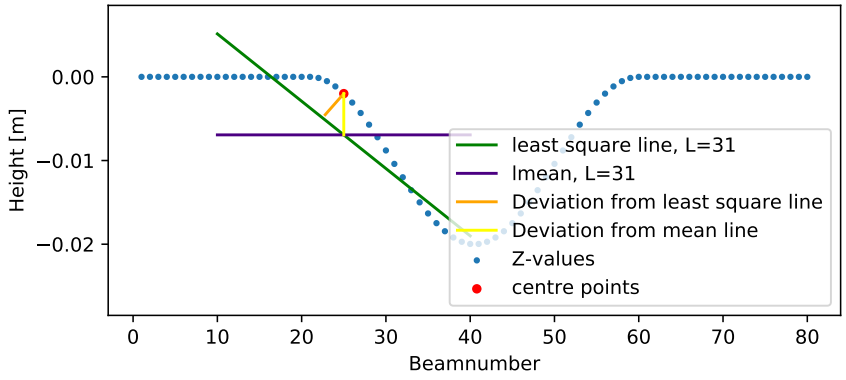


Figure 4.6: The difference between the least square line and a mean line of theoretical data. It is seen that the deviation from the least square line is smaller than from the mean line.

assumed that both are equal. In Figure 4.4, a draft of the orthogonal deviation is given. The difference between the deviation from the mean line and from a least squares is given in Figure 4.6.

When the least squares equation of equation 4.5 is expressed in the Big-O-notation, to express the running time behaviour by increase the input size, the matrix multiplication, $A^T A$, dominates with $O(m^2 n)$ the others steps, where m are the columns of the matrix and n the rows.[1] This feature is applied in Python and to make the calculation of the least squares fit faster, the distance variation between beam numbers is neglected. Then model A is the same for each centre point. The first part of equation 4.7, $(A^T A)^{-1} \cdot A^T$, is the same for all points with the same window length and has to be calculated only once.

4.1.4. Standard deviation of the deviation of expected range

Range is what the laser scanner measures. In this feature the variations in a measurement are used.

Even on a flat and horizontal road, the range changes for each beam number. Further away from nadir (higher angle of incidence), the range increases, see Figure 2.7.

In this feature the range is simulated for each beam number for a horizontal and flat road. This is done by calculating the angle which each beam number should have, by taking the fraction of the beam number by the maximum beam number times 360°:

$$\alpha_i = BN_i \times \frac{2\pi}{BN_{max}} \tag{4.14}$$

where BN_i is the ith beam number in a profile,
BN_{max} = number of beams within the profile,
α_i = the angle for each ith beam number in radians.

This angle α_i is used to estimate the range for each beam number, $R_{flat,i}$, assuming that the road is horizontal and flat:

$$R_{flat,i} = \frac{H}{\cos \alpha_i} \tag{4.15}$$

where α_i = the angle for each ith beam number in radians,
H = the height of the laser scanner above the road surface,
$R_{flat,i}$ = the range for a flat and horizontal surface for the ith beam number.

[1] $A^T A$ is a matrix multiplication with as input a $n \times m$ and $m \times n$ matrices and as output a $m \times m$ matrix. This gives a complexity of $O(m^2 n)$. The next step is to get the inverse matrix $(A^T A)^{-1}$, which has a complexity of $O(m^3)$. The matrix multiplication of $A^T y$ has a complexity of $O(mn)$ and the last matrix multiplication $(A^T A)^{-1} A^T y$ has a complexity of $O(m^2)$. The $O(m^2 n)$ dominates $O(mn)$ and the number of rows n is larger than the number of columns m, which means that $O(m^2 n)$ dominates $O(m^3)$.

This validation range for profile number 19500 is given in blue in the top figure of Figure 4.7.

With this validation range, a range difference between the range which is measured and the validation range is calculated for each beam number:

$$\Delta R_i = R_i - R_{flat,i}, \tag{4.16}$$

where ΔR_i = the range difference between the measured range and the validation range for the ith beam number,
R_i = the actual range for the ith beam number ,
$R_{flat,i}$ = the range for a flat and horizontal surface for the ith beam number.

An example of ΔR_i is given in green in the middle figure of Figure 4.7.

The ΔR_i is not a self-contained feature to determine deviations in the road surface. So for a rolling window with length 20 points the sample standard deviation of the ΔR_i is calculated:

$$f_{4,i} = s = \sqrt{\frac{\sum_{i=1}^{N}(\Delta R_i - \Delta \bar{R}_i)^2}{N-1}}, \tag{4.17}$$

where $f_{4,i}$ = s = the sample standard deviation,
ΔR_i = the range difference between the measured range and the validation range for the ith beam number,
$\Delta \bar{R}_i$ = the mean of the range difference,
N = the number of point within a window.

When a least square model is used for the arithmetic mean of the range difference between the measured range and the validation range (equation 4.7), the feature can be written as:

$$f_{4,i} = \sqrt{\frac{e^T e}{N-1}}, \tag{4.18}$$

where $e^T e$ is the sum of least squares errors.

The sample standard deviation is given in sea-green, Figure 4.7. In this Figure it is clearly visible in the range differences that the road in this profile is not horizontal and flat, but a bit curved. Also, the ΔR_i deviations are clearly visible, but an independent number is not given to these deviations. That is, a higher value of ΔR_i does not necessarily mean that there is damage. Using the sample standard deviation, an independent value is given to the points. When there is a curve in the road along the profile, the sample standard deviation will be more or less flat. Although, the deviations are then visible due to a higher sample standard deviation, while parts with less deviations get a sample deviation around zero.

4.1.5. Standard deviation of profile point density

In Figure 4.8 the theoretic point density over a pothole along a profile is given. It shows an approximately equal point distribution over the flat road surface, while the distance between points increases (decrease of point density) as the angle of incidence increases. This is done gradually and is not taken into account. In Figure 4.9 this effect is shown.

When there is a pothole/crack/hump this will disturb the equal point distribution. Generally, going down in a pothole will give less points within a certain distance, and going up in a pothole gives a higher point distribution along the profile.

The point density, the number of neighbouring points inside a sphere of radius R is estimated for each point with CloudCompare (Girardeau-Montaut et al., 2017). A radius of 0.02 meters is used. In this case, for most points only the neighbouring points in a profile are taken, because the distance between profiles is around 4 to 5 cm. So in this case no sliding window is used for estimating the point density.

For profile 19500 the number of neighbours is given in Figure 4.9.

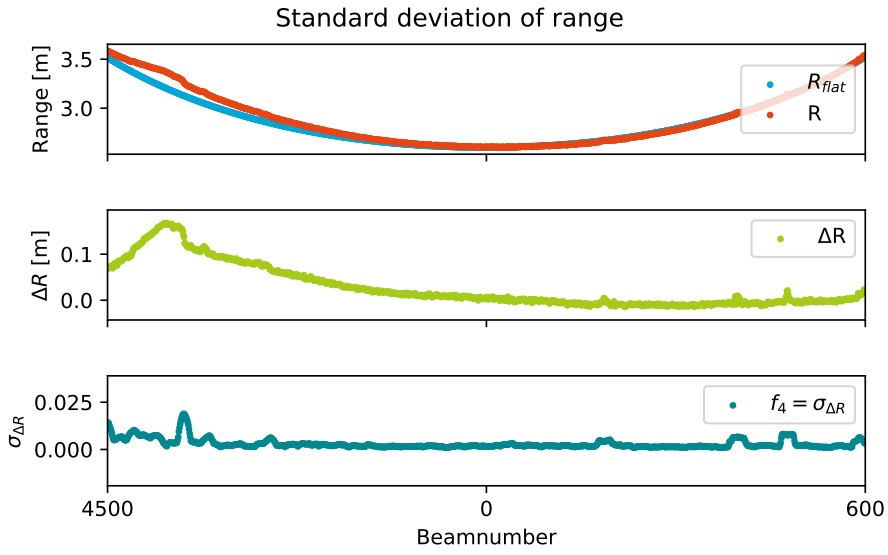


Figure 4.7: From top to bottom the three steps for creating feature, f_4. In the top plot the orange line is the measured range, R, of profile number 19500, while in blue the validation range, R_{flat}, for a flat and horizontal profile is given. In the middle plot the range difference, ΔR, is given. In the bottom plot the sample standard deviation of ΔR is given.

When the sample standard deviation is taken over a window length of 20 points (like equations 4.17 and 4.18), differences between the spacing of points (caused by potholes/cracks/humps) should appear.

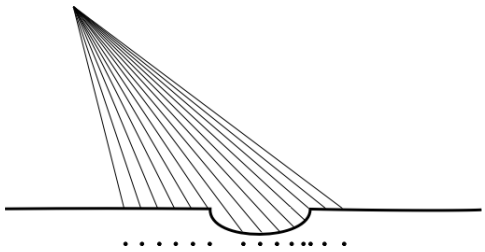


Figure 4.8: The point distribution change when there is a pothole/crack/hump.

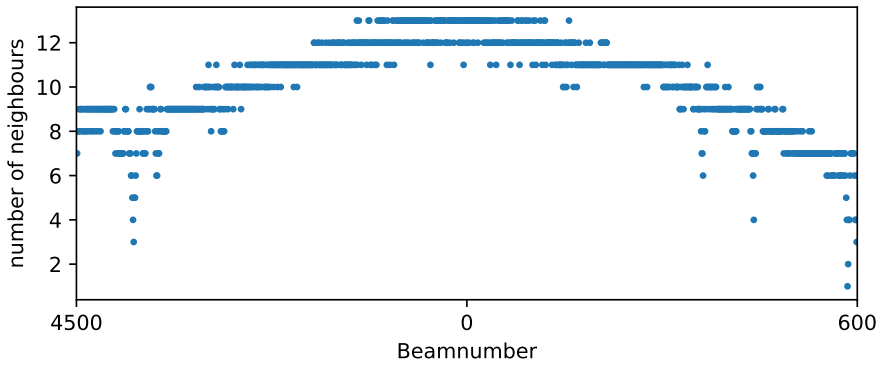


Figure 4.9: The number of neighbour points for each beam number in profile 19500. Below the car (around beam number zero) the point density is largest. Further away from nadir the point density decreases due the fact of that the distance between points increased. Deviations in the number of neighbours is an indicate of road surface defects.

4.1.6. Sum of different window lengths

For the (absolute) deviation from the mean and the deviation from the least square line, different window sizes can be used to calculate each feature. The results of different window lengths are summed and added as a separate feature, given in Figure 4.10. The intention of this concept is to create larger differences in values between non-damaged areas and damaged areas. Adding this features with different window sizes improves distinction between a flat road surface and an anomaly.

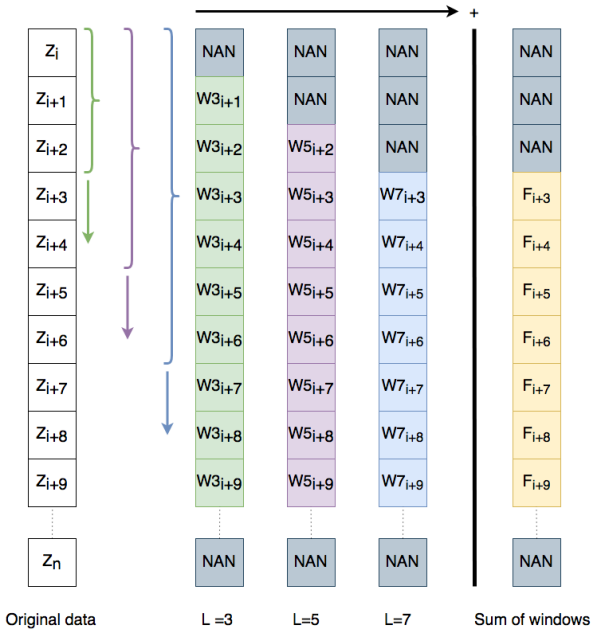


Figure 4.10: The concept of the sum of different windows. In this case a feature is calculated for three different window sizes: L=3 (green), L=5 (purple) and L=7 (blue). The results of these different windows are summed and stored in a new feature (orange).

4.1.7. Window lengths

For the above described feature different window lengths can be used for calculating the feature. In this research the following window lengths are used: 5, 21, 41 and 101 points. These correspond with approximately 1.6 cm, 6.7 cm, 13.1 cm and 32.3 cm respectively. The choice to use these lengths is to have a distribution of smaller window lengths to larger window lengths to fit different damage scales.

4.2. Step II: K-means clustering

For the classification of the road in damage and non damage points the robust Random Forest algorithm is used. Random Forest algorithm is a supervised learning algorithm, which means that training data is needed. In this project first the road inspectors classification was taken as training data. But later was found that this data was too coarse for using them as training data. So another method was needed to create training data for the Random Forest algorithm.

In this study, K-means clustering is used to classify a small selection of the data with known damage into two clusters (“no damage” and “damage”). Although, K-means clustering is already a classification method, it is here used for creating training data for the Random Forest algorithm. This step can be removed in the long run by using (validated) outputs of this method.

K-means clustering divides M points in N dimensions into K clusters so that each point belongs to the cluster with the closest centroid,(Hartigan and Wong, 1979). First K cluster centroids are randomly located. Then the Euclidean distance between a point and a cluster centroid is calculated. The points are assigned to the nearest cluster centre. New cluster centres are calculated by taking the mean of the average coordinates of the points in each cluster. It is an iterative process to find the optimal location

of the cluster centroid. An illustration of K-means clustering is given in Figure 4.11.

Each feature is scaled before the clustering is done. This is done by first subtracting the mean value, and divide it by the standard deviation of the feature. Scaling is important to create equal “distances” between feature values. Both scaling and clustering are done with the Python scikit-learn module (Pedregosa et al., 2011). The average complexity is given by O(k n T),(Pedregosa et al., 2011),

where k = the number of clusters
n = the number of samples
T = the number of iterations.

So increasing the numbers of samples it decreased the processing speed.

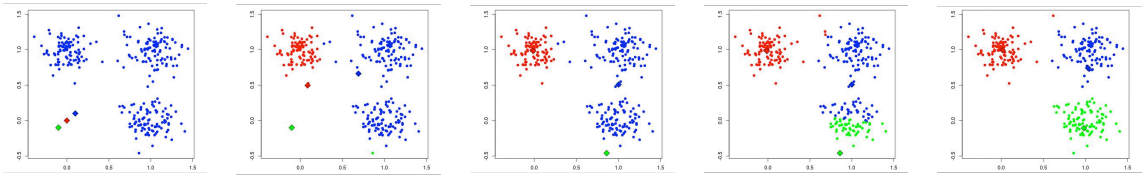


Figure 4.11: K-means clustering in steps. The first step is to chose the number of clusters, in this case 3, and give initial cluster centres (diamond shape). Than each point is assigned to the nearest cluster centre. The new cluster centres are calculated by taking the average coordinates of the points within each cluster. Iteratively the three clusters and their cluster centres are calculated, (Mubaris, 2017)

4.3. Step III: Random Forest classification

K-means results are used to train a supervised classification. Random Forest is a fast classification method and has excellent results (Belgiu and Drăguț, 2016). Another advantage of this classifier is that it can be used to generate feature effectiveness rank on discrimination (Belgiu and Drăguț, 2016; Yan and Xia, 2019). This can help to decide which features are most important and useful for the classification (Belgiu and Drăguț, 2016; Yan and Xia, 2019).

Random Forest Classification is a supervised classification method, based on an ensemble of decision trees (Liaw and Wiener, 2002) that grows through training towards a best combination (Pirotti and Tonion, 2019). A classification tree is a multistage approach which breaks up a complex decision into a union of several simpler decisions (Safavian and Landgrebe, 1991). Each node in a tree makes a binary decision using one feature, and subsequent decisions in a tree lead to a class label. A binary decision can be for example if a feature value of a point is higher than a certain value. To classify a data point, an input vector (consist of feature values) run down each decision node of the trees in the forest, (Kulkarni and Sinha, 2012). Each tree of the forest classifies this point to a class and thereby votes for that class. The forest determined the class for the point by the majority votes over all trees in the forest, (Kulkarni and Sinha, 2012). For example, if a forest consists of three trees and two trees predicts a point as damage the random forest results classifies that point as damage. An illustration of a random forest is given in Figure 4.12.

In this project, each decision tree contains 3 decision nodes and the random forest has 100 trees. Each tree uses only a selection of the available features to create unique trees. Per tree the number of used features is $\sqrt{n}$, where n is the total number of features. Training the algorithm is done by dividing the training data set and use only a small part to avoid overfitting of the system. Overfitting is “the production of an analysis that corresponds too closely or exactly to a particular set of data, and may therefore fail to fit additional data or predict future observations reliably”, (Lexico powered by Oxford, n.d).

In this project, the RandomForestClassifier from the Python scikit-learn module is used (Pedregosa et al., 2011). After training, the whole data set is classified by this random forest classifier.

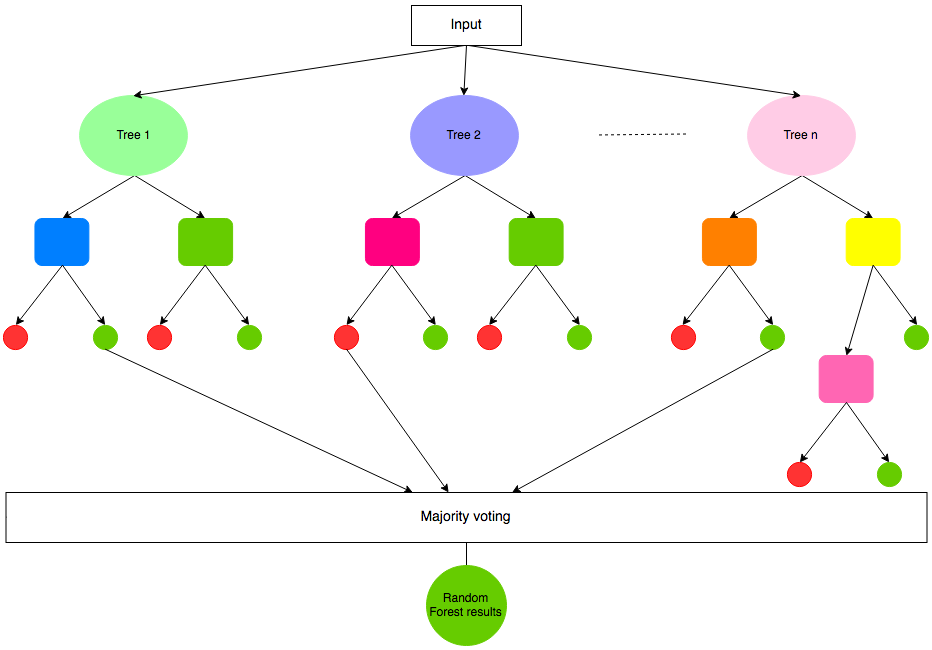


Figure 4.12: Illustration of an random forest algorithm with n decision threes. The red and green circles are the possible predictions of a decision tree. In this example this random forest has n trees. When the majority of the predictions of the decision threes is green the Random Forest will label the point with the green class. Image based on Redesign of the original inspired figure of Koehrsen (2017).

4.4. Step IV: Mathematical morphological operations

After classification some noisy points can occur. To filter this points out, mathematical morphological operations are used. Also this technique is used to connect damage areas to each other when one non-damage point are between two damage planes.

This is done by the morphology module of the Python scikit-image package, (van der Walt et al., 2014). Mathematical morphological operations assigns pixels in an image based on the values of neighbouring pixels. Which neighbouring pixels are used is defined by the morphological operator. In this project, a diamond shape (+-shape) of 1 centre point is used, given in Figure 4.13. This morphological operator is moving over the pixels of the input data, like a sliding window, and assign values to the centre point based on the values of the input data of the shape of the operator. Here four types of mathematical morphological operations are considered: opening, closing, dilation and erosion. Dilation is an operation, which changes a “no damage” pixel into a “damage” pixel when neighbouring pixels are classified as “damage”. Erosion is the opposite operation of dilation. Erosion gives “damage” pixels a “no damage” value when the neighbouring pixels are classified as “no damage”. Erosion decreases the size of objects, while dilation increases the size of objects, and can merge multiple objects into one (Smith, 1997). Mathematical morphological closing is a combination of dilation followed by an erosion operation (Smith, 1997). Mathematical morphological closing removes gaps in connected damage pixels. Examples of the different results of the mathematical morphological operations on a letter B are given in Figure 4.13.

In this project, first objects smaller than 3 points are removed. This is done, because it is assumed that areas smaller than 3 points (approximately 1 cm) are to small to be serious damage. The size of objects is determined by the adjacent “damage” points, see Figure 4.14. After removing small objects, morphological closing is used. Morphological closing is used to connect damage areas which are lying close together. This is done by projecting the data as a matrix with the number of profiles as rows and the number of beams as columns. An example of these steps is given in Figure 4.15.

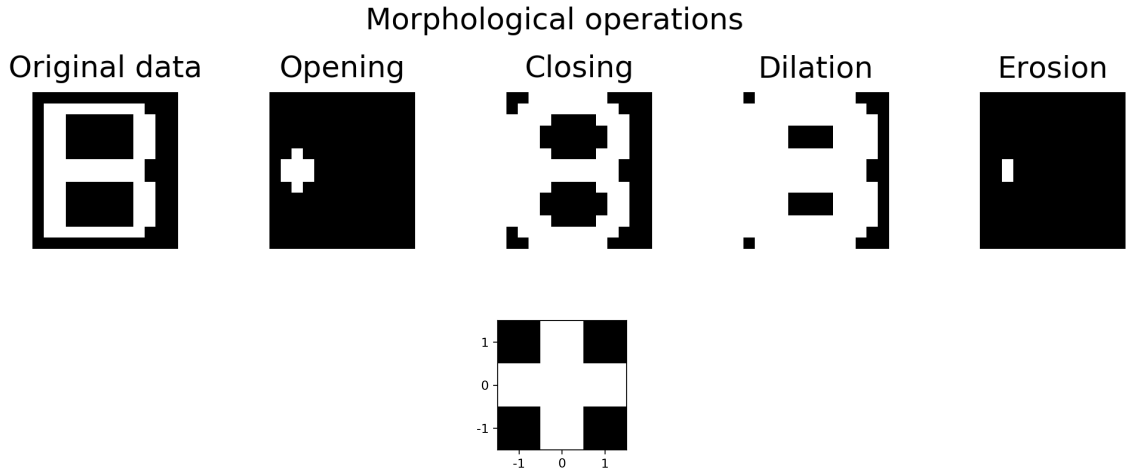


Figure 4.13: Mathematical morphological operations and operator. The top images give the mathematical morphological operations on the letter B. The white cells are filled with ones, black cells corresponding with zeros. For neighbouring pixels a diamond-shape around 1 centre point is used. This shape is given in the bottom figure.

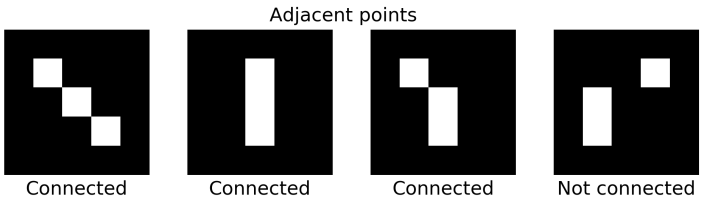


Figure 4.14: Matrices with three damage pixels. The left three matrices contain connected points and will not removed, while in the right matrix the points are not connected to each other. All these points will removed, since the objects are smaller than 3 points.

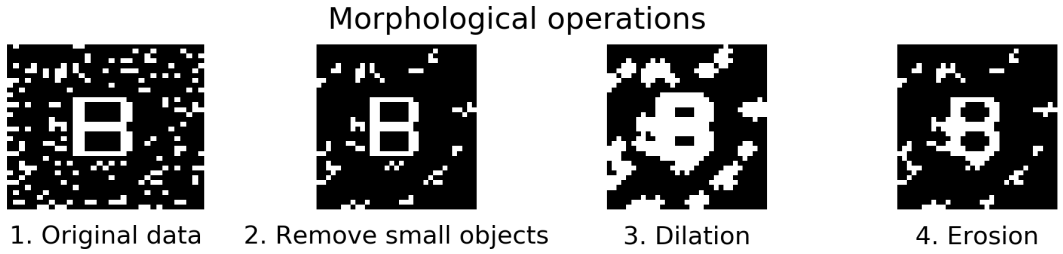


Figure 4.15: The morphological operations on a matrix with the letter B, and some noise values. The left image (1.) gives the original data, a letter B with some random noise. In image 2 is seen that objects smaller than 3 pixels are removed. In step 3 dilation is used to make the damage areas bigger, so that image areas lying close together are connected to each other. In the final step (4), erosion is applied to make the damaged areas smaller, so that the damaged areas get back their original dimensions. Combining step 3 and 4 is the same as using morphological closing.

4.5. Validation

The validation of the above described method is done with the help of damage shapefiles of a road inspector from a third party. The shapefiles are three files with point, line and polygon data. These data files are converted to raster data with the GDAL (GDAL/OGR contributors, 2018) tool gdal rasterize. This tool rasterizes the shapefile (vector geometries) with a pixel size of 0.05 × 0.05 meter. Then the three raster files are combined to one large raster file. This validation data is projected to the point data, such that each data point gets a ground truth damage value.

To describe the quality of this method the overall classification accuracy is used. The overall accuracy is defined as the total number of correctly classified points divided by the total number of points:

$$\text{overall accuracy} = \frac{\text{correct classified points}}{\text{total points}} \tag{4.19}$$

Also a confusion matrix will be made. A confusion matrix is a visualisation of the performance of a classification algorithm, (Brownlee, 2016). The columns of the matrix represent the true values, while the rows represent the predictions by the classification algorithm. The diagonal (from upper left to lower-right) elements in the confusion matrix represent the number of correctly classified points. The other values in the matrix are points mislabelled by the classification algorithm. The commission error is the fraction of points which is predicted to be in a class but does not belong to that class. They are falsely attributed to a class and are given in the rows of the confusion matrix (except for the correctly classified points). The omission errors are the fraction of points that belongs to a class but were predicted to be in a different class. They are falsely omitted from a class and they are given in the columns of the confusion matrix (except for the correct classified points). Higher diagonal values indicate a better classification algorithm.

In this project, a confusion matrix with the types of damage will be made. Than it will become clear which types of damage are better classified than other types. Although the confusion matrices give a good indication of the method's performance, there is also a drawback of this validation method. The drawback is that the validation data are shape files. The damage shapes are projected to the points and only the points in the origin data get validated values. So damage which fall between two profiles are not counted in with this validation method.

4.6. Conclusion

In this chapter the proposed methodology was given. It is based on the fact that damage has deviations in height and/or intensity in comparison to the surrounding pavement without damage. To detect damage several features based on a sliding window algorithm were made. A comparison of the centre point and surrounding points (in a profile) in a window was made.

After feature extraction K-means clustering is used to create a training data set. This is done to make the whole classification method automatic. In this case, K-means clustering is done for two clusters, "damage" and "no damage". It is also possible to make multiple clusters, one for each type of damage. But this will have to be investigated. In the future this K-means clustering can be removed from the workflow and already classified damages can be used as training data for the Random Forest classification.

The following step is to use a quarter of K-means clusters to train the Random Forest classification algorithm. This is done to give the possibility to make a decision on different features, while in K-means clustering all features in the clustering process are used. Also, when the Random Forest algorithm gives good results no further training is needed, and only classification can be done.

The last step of the method is to remove single, noise, points and connect "damage" pixels to each other.

Validation is based on damage classification shapefiles of a road inspector that is transferred to the individual points. With confusion matrices and the overall accuracy the accuracy of this method is described.

With this workflow it is not yet possible to give a “damage” value to a point. To make this possible, an addition of additional classes is needed. Then each class represents a “damage” value e.g. from small to large damage and for each type of damage.

5

Road damage classification results

The results of the proposed method are presented in this chapter based on the craquel introduced in Figure 2.1. First the features will be described. Second, the results of the K-means clustering and Random Forest will be given. Next, the results will be validated with data of a road inspector from a third party. Finally, a discussion on the results is given.

5.1. Features

For this research 32 features are calculated as described in Section 4.1. Four different window sizes (5, 21, 41, 101 points) are used for the features “(absolute) deviation from the mean” and “the deviation from a line”. The results of each feature will be discussed.

5.1.1. Deviation from the mean

The deviation from the mean is calculated for 4 different window sizes (L=5, L=21, L= 41 and L=101 points), and the height or intensity are used as input data. First the results made with the height values will be given and next the results made with intensity values are described.

Height values

The results of the 4 different windows of the height values shown in Figure 5.1. With the small window size of 5 points, damage is invisible, although in the centre of the image, there are deviations between positive and negative values. In the lower left corner, higher values are visible. This corresponds to the road boundary. By increasing the window size the craquel appears The boundaries of the craquel appear as positive deviations from the mean, while the core of the craquel has negative values. The window size with 41 point shows the best results, as even the smaller longitudinal cracks are visual on the left of the plot.

Intensity values

The same procedure is used with the intensity values as input to calculate the deviations from the mean. The results for different window lengths are given in Figure 5.2. The road boundary is in all figures visible as a negative deviation. But with a larger window it is more pronounced. The craquel is not visible in the results of all window sizes, although the two different asphalt types (the boundary can be found around profile number 19450) are detectable. This difference is visible due to the smoothness of the values. The newer asphalt (the top of the figure) is more homogeneous, while the older asphalt (lower part) is more heterogeneous.

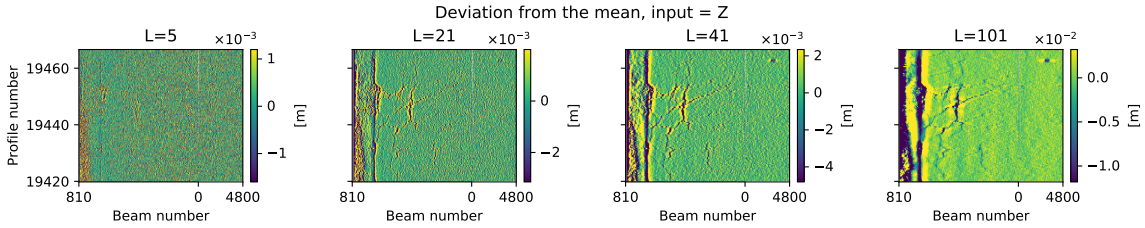


Figure 5.1: The deviation from the mean of elevation values for the damage type craquel. From left to right the window size increases: 5, 21, 41 and 101 points. With the smallest window the craquel is not visible, while with the other three window length the craquel appears due the lower feature values.

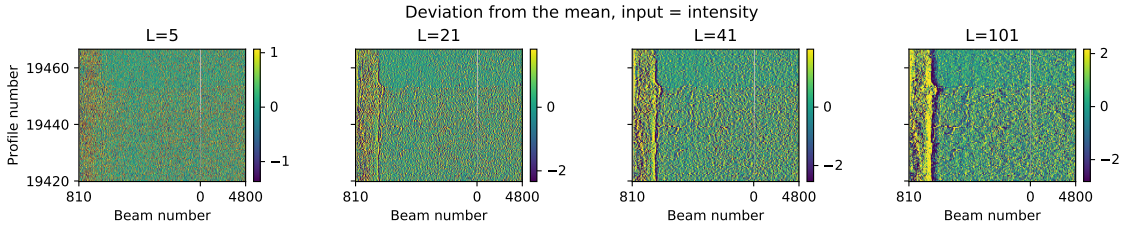


Figure 5.2: The deviation from the mean with the intensity values as input for the damage type craquel. From left to right the window size increases: 5, 21, 41 and 101 points. The difference in homogeneity corresponds to two different asphalt layers (on top of the image a newer one). The boundary of the road is clearly visible (left of each plot) due to a change of sign at the road boundary.

5.1.2. Absolute deviation from the mean

As with the deviations from the mean the absolute deviation from the mean is calculated for 4 window lengths (5, 21, 41, 101 points) for both height and intensity values.

Height values

The results of the absolute deviation from the mean of the height values are given in Figure 5.3. They resemble to the deviation from the mean, although there is no difference in sign for the core and the boundary of the craquel. Again, the smallest windows give no clear distinction between damage and no damage points. But by using the largest window size, some details disappear which are found in other results. The craquel with some details is best visible with window length 41.

Intensity values

In Figure 5.4 the absolute deviation from the mean is given when using intensity values. The difference between the two asphalt types is more clear in the absolute deviation from the mean than the deviation from the mean. But the craquel is not well detectable.

So to detect craquel the intensity-values are not useful, while the height values give good visible results for both (absolute) deviation from the mean.

5.1.3. Deviation from a line

The results are given for the height and intensity values respectively in Figures 5.5 and 5.6. These results are similar to the results of the deviations to the mean. The calculation of the deviation to a line takes more time than for the deviation from a mean. The calculation for the deviation to a mean takes around 7 seconds, while the calculations for the deviation to a line takes around 25 minutes on a laptop with a 2.4 GHz Intel Core i5 and 8 GB RAM for a 120 meter road section (around 350 0000 points). This can probably be explained by inefficient programming. There is a limited theoretical difference between the two features in computational complexity.

This feature does not provide the expected additional value with respect to the deviation from the mean. So with a view to the processing speed, the advice is not to use this feature.

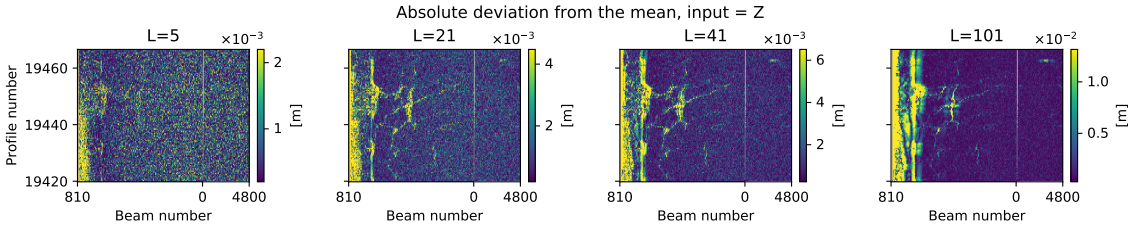


Figure 5.3: The absolute deviation from the mean with elevation values as input. From left to right the window size increases: 5, 21, 41 and 101 points. The craquel is distinguishable by the higher values. The craquel with some details is best visible with window length 41.

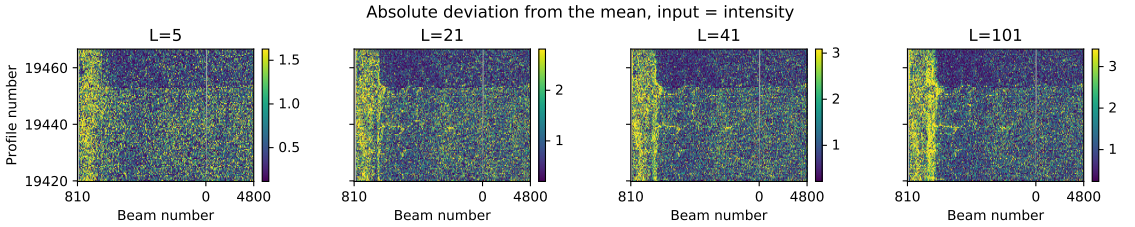


Figure 5.4: The absolute deviation from the mean of intensity values for the damage type craquel. From left to right the window size increases: 5, 21, 41 and 101 points. The difference in asphalt is clearly distinguishable by the difference values between the two asphalts. The boundary is visible due the higher feature values, but the craquel is not detectable.

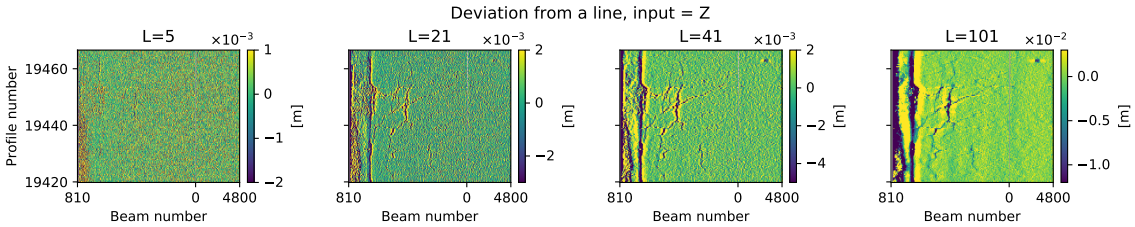


Figure 5.5: The deviation from a linear least square line of the elevation values for the damage type craquel. From left to right the window size increases: 5, 21, 41 and 101 points. The smallest window size gives no clear damage. Increasing the window, the damage is visible due the negative values.

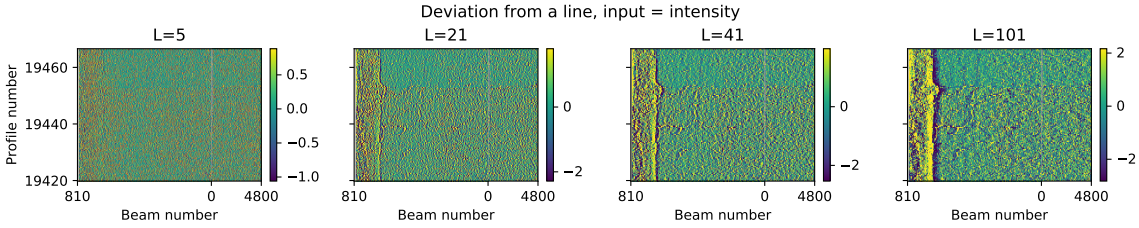


Figure 5.6: The feature deviation from a linear least square line of the intensity-values for the damage type craquel. From left to right the window size increases: 5, 21, 41 and 101 points.

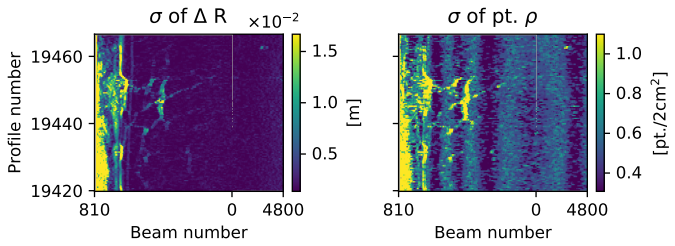


Figure 5.7: Left: The standard deviation of the range difference of the area of the craquel. The higher values of the standard deviation represent the craquel, while standards deviations close to zero represent undamaged road.
Right: The standard deviation of the point density. A higher deviation corresponds to the craquel. This result is more rough than the standard deviation of the range difference. Vertical stripes with lower feature values appear in the data. The source of this vertical stripes is unknown.

5.1.4. Standard deviation of range

The left-side of Figure 5.7 shows the results of the standard deviation of the range for a window lengths of 20 points. The window length of 20 points is empirically defined. The higher standard deviations represent the craquel and the boundary of the road. With this feature, smaller details of the craquel are visible as well.

5.1.5. Standard deviation of point density

The standard deviation of the point density shows notable results, given in the right pane of Figure 5.7. Again the window size is 20 points. The craquel appears in these results via a higher standard deviation. But regular low deviations are also visible as stripes. Probably this is an artefact of the laser scanner.

Although the craquel is visible, the details which are visible in the standard deviation of the range are not visible in this feature.

5.1.6. Sum of different windows

The idea of the sum of the different windows is that the values are increasing and that in this way the difference between damage and undamaged pavement is more visible. The results of the sum of the different window of the (absolute) deviation to the mean (height and intensity input) and the deviation to the line (height and intensity input) are given in Figure 5.8 and 5.9. The craquel is clearly visible in the summed features based on heights. The craquel, boundary of the road and the pothole are distinguished due the lower values relative to the surroundings. Also just next to the damage higher values are seen.

For the sum of the different windows of the features “the deviation to the mean” and “the deviation to the line” with the intensity values as input, only the boundary of the road is clearly visible. Also in this section two types of asphalt are visible due the homogeneity of the values. On top of the images the feature values are smooth and homogeneous, while the bottom section is heterogeneous.

5.1.7. Correlation

The correlation between the described features is given in Figure 5.10. A high correlation between different window lengths for the same feature is indicated. There is also a high correlation between the deviation to the mean and the least squares line. So an advice is not to use the deviation to the least square line, because the calculation for this feature is more computationally demanding than the deviation to the mean. However, the “standard deviation of the range” and the “standard deviation of the point density” are uncorrelated.

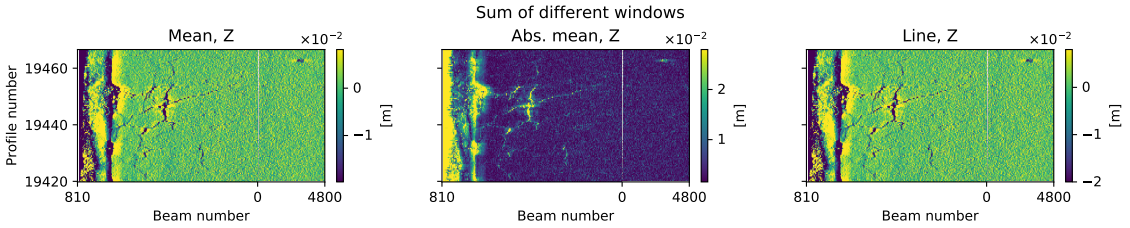


Figure 5.8: Sum of different windows for the deviations to the mean, the absolute deviations to the mean and the diversions to a line with as input the height values respectively. The sum of the different windows of the deviation to the mean feature with as input the intensity values. The craquel is distinguishable by the different feature values compared to with the surrounding road.

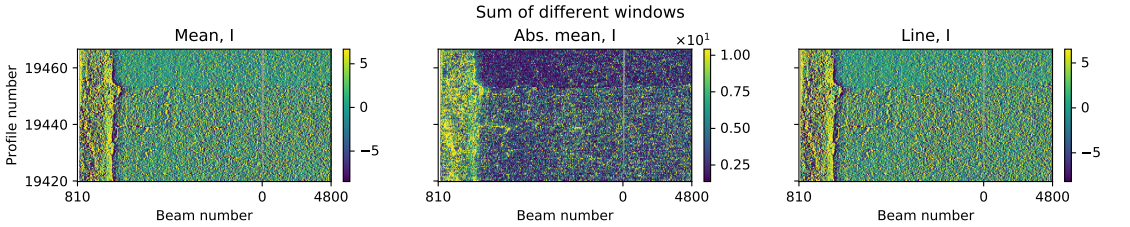


Figure 5.9: The sum of the different windows of the three features intensity based. The different types of asphalt can distinguished by the homogeneity of the road surface. This surface difference is best visible in the absolute deviation to the mean clearest visible.

As there is strong correlation between the window lengths of each feature, it is the question if the different window sizes have an added value for the classification. To make the algorithm more efficient only the essential window length of a feature could be used. Identifying the essential window length of a feature can be done visually or to make a very detailed damage validation set and the overall accuracy and a confusion matrix.

5.2. K-means clustering

K-means clustering (K=2) is done on a known damaged area to create training data for the Random Forest classification algorithm. The training area is from profile number 19400 till 19550 and includes a pothole and craquel.

The results of the K-means clustering, done with all features, are given in the left plot of Figure 5.11. Here it can be seen, that the K-means clustering, did not make clear distinction between damage and non-damage. It gives a more or less uniform distribution of damage points. Although after profile number 19480 there are less points belongs to the class “damage”. This corresponds to newer asphalt type and associated colour difference.

These results are disappointing. It would be expected that the craquel would become clearly visible. For example, when setting a threshold for the $\sigma\Delta R$ on 0.003 meters and all points above this threshold get the class damage, the training data would be much better - based on a single feature only. This is given in the right plot of Figure 5.11.

Although K-means clustering returns disappointing results for this section, the advantage of using K-means clustering is that it can be used automatically on different sets of road data, while a hand-made threshold it needs to be set for every section again. Because, although for this set (Fig. 5.11 right) these boundary values give better training data, this does not mean that this will also be the case for another damage type on another road. Varying with the features where K-means clustering is applied on, makes for different results.

The processing time for K-means clustering was around 40 seconds for 241696 points. So this is a relatively fast process.

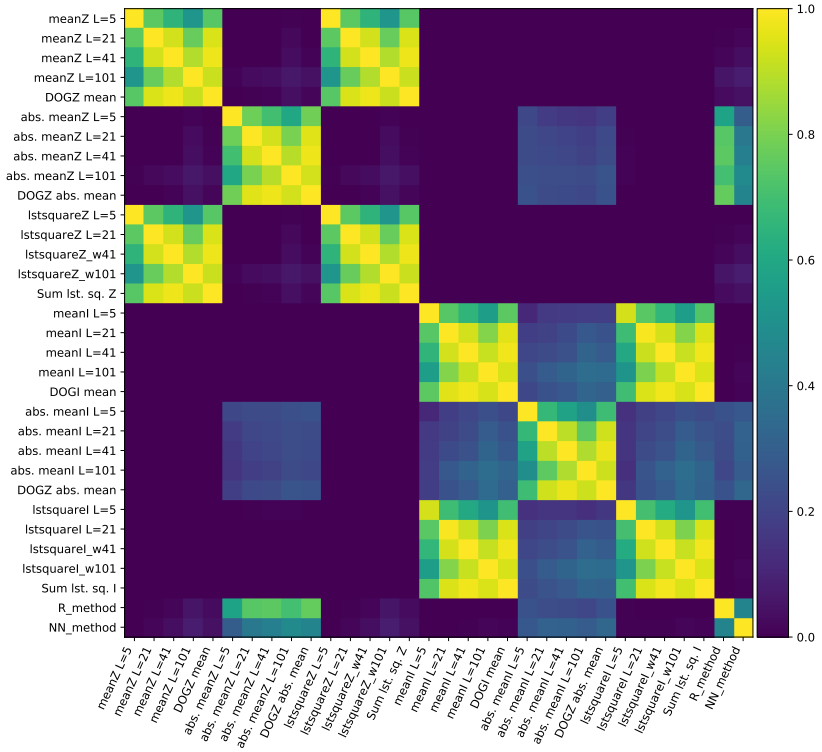


Figure 5.10: The correlation between each feature. There is less correlation between the features calculated with the height values and the intensity values. The standard deviation of the point density and the range are independent features, they have little correlation with other features.

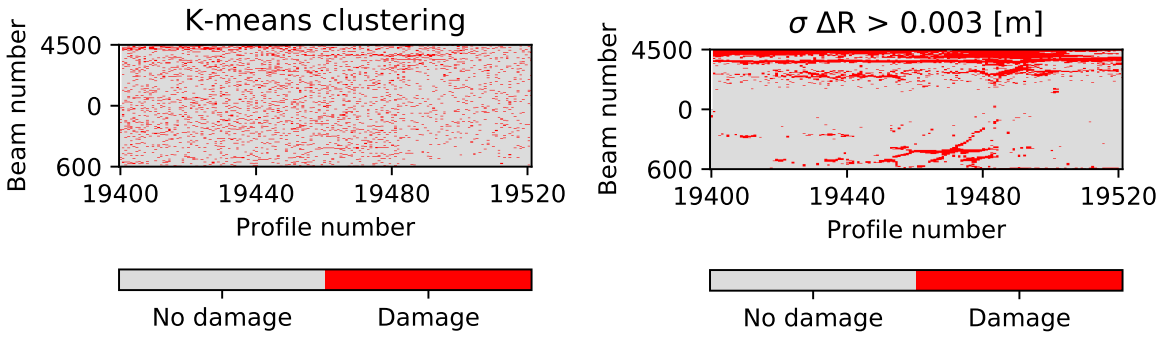


Figure 5.11: Left: The result of K-means clustering on the training area (profile numbers 19400-19550). This result is based on all features.
Right: When a boundary is set for $\sigma\Delta R$ on 0.003 meters the training data are improved a lot. The training data is based on only one feature.

5.3. Random Forest Classification

Now training data is available from the K-means clustering and is used to train the Random Forest. To avoid overfitting a subset of 25% of the K-means clustered data set is used in the Random Forest training. It takes 1 minutes and 2 seconds CPU time on 241 696 points to train the Random Forest on the damage set. This training was done with a 2.4 GHz Intel Core i5 and 8 GB RAM.

The results after training and the morphological operations are given in Figure 5.12a. The damage outline are not clearly distinguished in these results. Boundary damages (left in transverse crack plot, right in the raveling plot, right in the boundary damage plot and left in the craquel plot of Figure 5.12a) are marked as damage.

When training the Random Forest algorithm with the training data using a threshold (the right plot in Figure 5.11), the results are much clearer than using the K-means training data. It shows less noisy results. When the results of Random forest trained with the K-means data set (Fig 5.12a) are compared with the results of Random forest trained with the “threshold” data set similarities of damage locations can be seen (5.12b). For example for raveling, these are clearly found in the middle of the image in Figure 5.12b and this vertical stripe is found in Figure 5.12a, but then not so thick and clear. These examples using two different training data sets show the importance of good training data. Good training data will show better and clearer results (with less noise).

5.3.1. Feature relevance

The feature relevance is calculated for the above two Random Forest algorithms and is given in Figure 5.13. When training is done with the K-means training data set, classification is predominately based on the features with intensity values as input data, given in Figure 5.13a. While classification is based on elevation values features, point density and the range, the threshold data set is used for training the Random Forest, Figure 5.13b. The effect of training with the threshold data set is that the feature where the threshold was set has the largest influence on the classification as well. Also, the features with intensity values as input are not part of the classification decisions. Probably this can described by that the range is more correlated with the Z-values than with the intensity values.

Also, it should be noted that another ranking is applied after every new training.

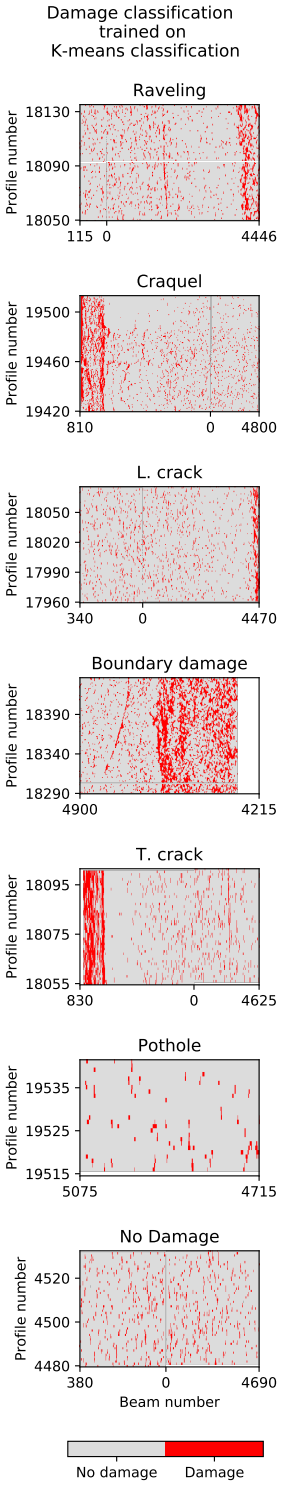
5.4. Validation with a road inspectors classification

A road inspector marked road damage on the 360°photos. This was done in ESRI shapefiles. These shapefiles (point, line, polygons) are converted to a raster of 5 by 5 cm. Through rasterising the shapefiles, some pixels are no longer connected, therefore the number of damage segments is increased. The raster data is projected to the point data, such that a point to point comparison can be made. A problem with this method is that not all damage is projected to the point data, for example damage between two profiles is omitted. Therefore only damage that could have been detected is validated.

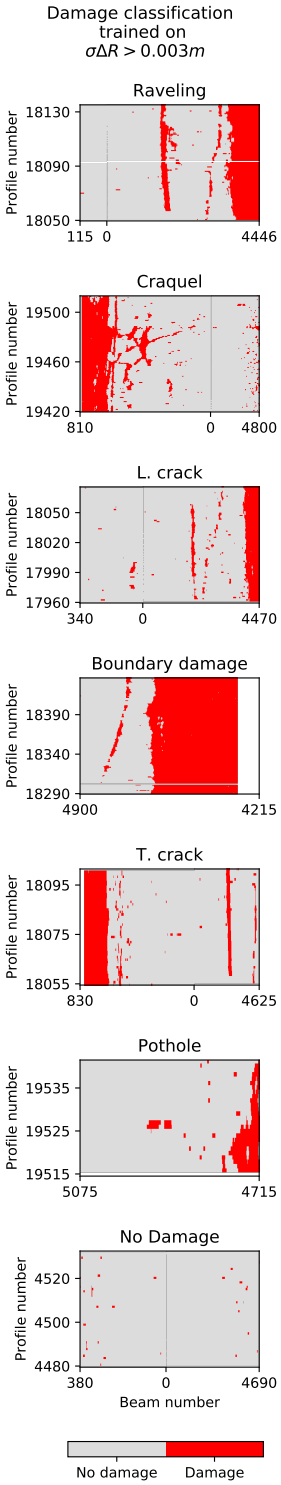
In Figure 5.14a, the road inspectors damage classification is given for each damage type of Figure 2.1. Here it is seen that the craquel is rough and the pothole is classified as craquel as well. Also a custom detailed damage classification is made based on the road inspectors classification and orthophotos. This is done for a small section, profile numbers 13480-19700, where all different damage types are present. This gives a much more detailed classification, especially for the craquel, given in Figure 5.14b. This detailed classification is used for an optical validation.

5.4.1. Overall accuracy

The overall accuracy is calculated for two combinations: the road inspectors validation versus the Random Forest classification based on K-means clustered data and the road inspectors validation versus the Random Forest classification based on a threshold for clustering the data. The detailed validation is leaving aside in calculating the overall accuracy and the confusion matrix, as the validation data is misaligned with the point cloud data.

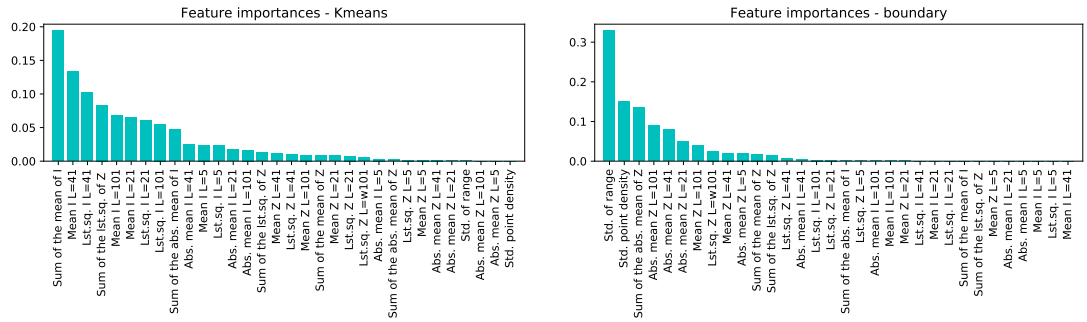


(a) Damage classification based on K-means clustering. The results are noisy, but some damages are visible when is known where the damage is located.



(b) Damage classification based on $\sigma\Delta R > 0.003$ meter clustering. The results are less noisy, and damage is clearly visible.

Figure 5.12: Damage classification done by Random Forest.



(a) Training with K-means clustered data

(b) Training with setting a threshold clustered data

Figure 5.13: The features ranked on importance in the Random Forest classification. Here is seen that with the K-means training data of Figure 5.11 the classification is based on the intensity values. Using the threshold data set for training the Random Forest classification the features based on elevation values are more important in the classification. The threshold on the standard deviation of the range has most influence on the classification.

The overall accuracy is for the Road inspector validation - K-means combination 73 % and for the Road inspector validation - boundary combination 58%, although the data trained with the boundary clustered data shows less noise and clearer results it has a lower overall accuracy. This can explained by the noisy damage classification based on K-means clustered data, where the overlap with the rough and large damage patches is larger.

5.4.2. Confusion matrices

For each method results (with K-means clustered data and the threshold clustered data) a confusion matrix is made with respect to the road inspectors validation data Figure 5.15. In these confusion matrices the normalised values are used. This is done by dividing the number of points in the confusion matrix by the total number of points. The numbers in a confusion matrix are than between zero and one and a higher number corresponds to a better match. These confusion matrices shows that craquel and boundary damage is more easily found with the threshold training data. Then 32 % of the craquel is found and for boundary damage it is 28%. While for the K-means clustered data it is around 10 % and 4 % for craquel and boundary damage respectively. Also with the threshold clustered data more non-damaged area is classified as damage. This can be explained by the fact that the road verges are also classified as damage, see Figure 5.16.

5.4.3. Optical validation

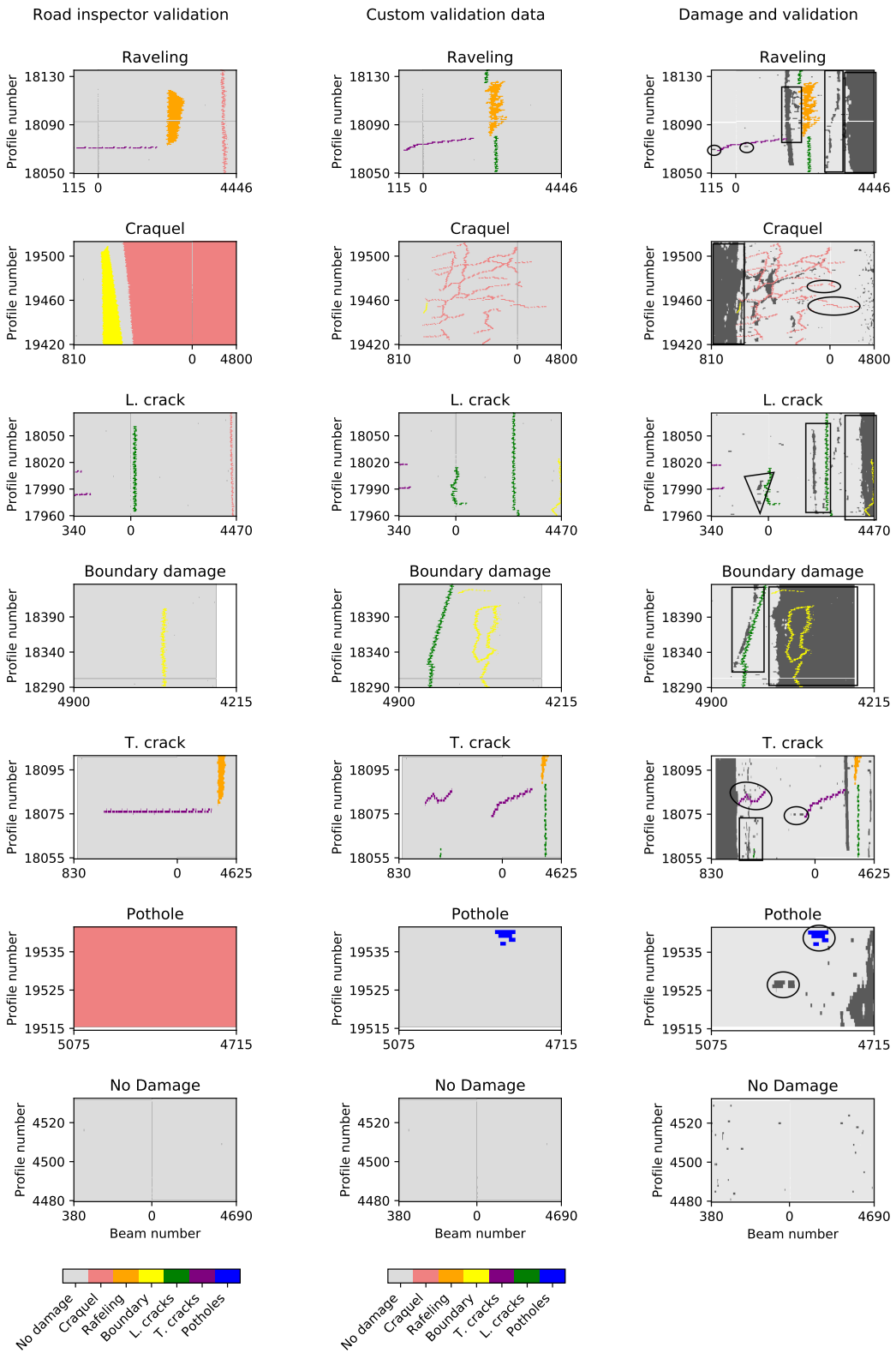
When the clear results of the Random Forest trained by the threshold data is compared with the custom validation data it is clear that these two data sets are not overlapping each other, Figure 5.14c. There is a shift between the detailed validation and the point data. For example, the pothole is shift in beam and profile direction.

This offset between the validation data and the point cloud data can have multiple reasons:

- The camera distortions are not correctly accounted for.
- The accurate camera height is needed to generate orthophotos. The exact camera height was unknown, and with trial-and-error, the best fit was created. An erroneously high camera location gives wider orthophotos, while a lower camera height gives smaller orthophotos. A small difference in camera height could be the cause of the offset in the orthophotos.
- Another option could be that the camera and the point cloud data are not exactly georeferenced when the data acquisition was done.

To cirumvent the offset a manual, optical validation is done with help of Figure 5.14c. Each damage set is described below.

Raveling: Some parts of the raveling are detected by the method. Also the longitudinal cracks are



(a) The road inspectors damage classification.

(b) Custom damage validation data set

(c) Damage classification and the detailed validation data.

Figure 5.14: Left: the rough road inspectors damage classification. Middle: the custom refined damage classification based on photos and the classification of the road inspector. Right: the damage classification and the detailed validation data. Here it can be seen that there is an offset between these two data sets, what make validation of the method difficult.

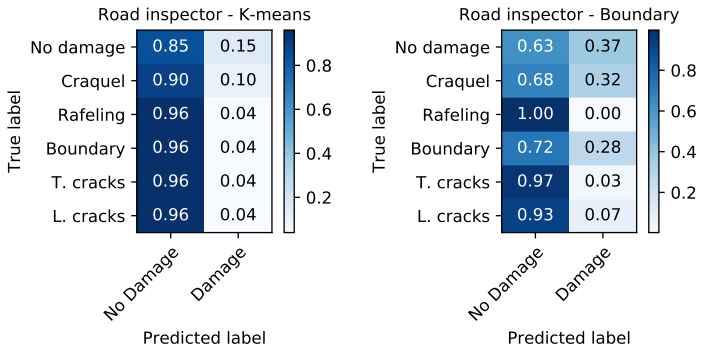


Figure 5.15: The confusion matrices for the two different training data sets and the road inspectors validation. The most overlap with damage found by the road inspector and the method is when the Random Forest is trained on data refined by a threshold. However, more non-damaged area is misclassified as damage in this case.

clearly seen. The whole transverse cracks are not seen, only small parts of them. The road verge is classified as damage on the right side of the plot. Also, a vertical line is seen on the right side of the plot. This damage is not visible in the orthophoto, but when looking for it is present in the intensity data. This part has a slightly lower intensity than the surrounding points.

Craquel: In the craquel plot the large and longitudinal cracks are detected by the method. The transverse cracks of the craquel are not clearly visible. Also the road verge (left in the plot) is detected as damage.

Longitudinal crack: Both longitudinal cracks are classified as damage. The boundary damage (right lower corner) is detected as well. Only small parts of the transverse cracks (left) are detected.

Boundary damage: The boundary damage is detected and the whole road verge as well. The longitudinal crack is clearly visible with the method.

Transverse crack: The transverse cracks are difficult to detect with the method. Only small parts of the crack are detected. The road verge (left) is classified as damage.

Pothole: The pothole is recognised by the method, but in two parts. The road boundary on the right side of the plot is detected as well as damage.

No damage: In the no damage area only small locations are detected by the method as damage. It is difficult to validate with the photos if these small areas are damage or not.

Overall damage is detected where damage is found by the road inspector. The road verges and longitudinal cracks are detected as damage. Only small parts of the transverse cracks are distinguished by the method. With the current data it is impossible to verify if the small regions identified as damage by the method are in fact small damages or are misclassified.

5.4.4. Detectable damage size

It is important to know which sizes of damage can be detected with this method. Cloud Compare (Girardeau-Montaut et al., 2017) is used to measure the sizes of the damage found. This is done for the Random Forest classification trained with the boundary clustered data. Some parts of a longitudinal crack of 1 cm width are found. Smaller damages are not found with this method, because it is assumed that 3 points or less connected pints are noise.

The pothole, with a diameter of 10 cm, is detected as well. Remarkable is that not the entire hole is found as one damage area, but in two, see Figure 5.12b. This is probably due to the window lengths used. A shorter window length can “fall” completely in the pothole and no large deviations to the mean are found at the centre of the pothole/damage.

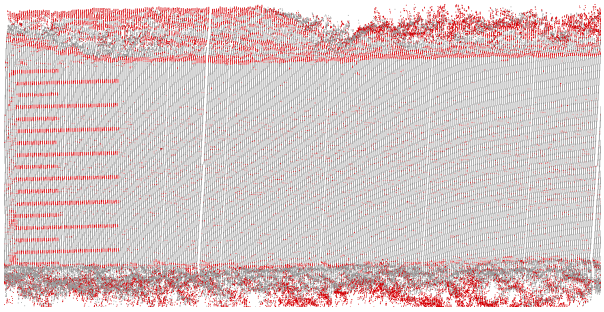


Figure 5.16: The small section of the road where the the classification is done with the K-means clustered data. Here is seen that the road markings (left) are marked as damage.

5.4.5. Unexpexted results

When the classification is done with the K-means clustered data, Figure 5.16, the road markings of the speed bump are marked as damage. This could be explained by the high intensity of the road markings and that the Random Forest used mainly intensity based features to do the classification. Pre-processing the point cloud data to remove high intensity values should reduce this effect.

5.5. Discussion

First, validation data is discussed. Second, the pros and cons of adding two available laser scanners are given. Third, the K-means clustered training data is reviewed. Next, the relevance of different features and the use of different window sizes are discussed as well. And the use on other road types are discussed. Finally, the accuracy of the point cloud is reviewed.

5.5.1. Validation

Validation data

When the results are validated with the validation data from the road inspector, the results are disappointing. For the road inspectors validation data it can be explained by the rough polygons marking the damage locations. To give a better validation of the method the point cloud and orthophotos need to be georeferenced exactly such that each point corresponds to the same location in the point cloud as well as on the photo. The use of orthophotos instead of the 360° photos is, in my opinion, easier to select damage and to draw them. A disadvantage of using orthophotos for selecting damage is that they are in 2D, while the point cloud is in 3D. Projecting the selected damage onto the point cloud is an extra error possibility.

An addition to the validation data would add depth and width of the damage. But this require fieldwork. In this way the sizes of possible detectable damage can be defined. When knowing this, the necessary beam width can be determined. For example, when smaller damage sizes can be detected right below the car (where the beam width is minimal) and not x meters next to the car, this may be due to the fact of the beam size or the incidence angle.

Other validation methods

In this project, validation is done point by point. Another method could be to count the amount of large damage. Another option could be to create a bounding box around each damage in the validation data and each detected damage. The intersection of union of each damage could used to evaluating the classification algorithm.

5.5.2. Improvement of spatial resolution

With this single scanner acquisition method (scanning in profiles perpendicular to the driving direction), the possibility that a transverse crack falls between two profiles is large, and increases further when driving faster. On the mobile mapping car two additional laser scanners are mounted, such that the profiles intersect each other behind the car, Figure 2.3. By adding these scanners to the point data, the spatial resolution will increase. The disadvantage of adding these two laser scanners is that they are of a different type (with another accuracy) and the orientations are different (relative to each other and the profile laser scanner) so a sliding window can not be implemented concurrently for all three scanners. Also it is important that all three laser scanners are georeferenced exactly the same. The advantage of using these two extra laser scanners is that the spatial resolution increases. This makes it logical to use the sliding window algorithm in different directions, such as horizontal, vertical and diagonal. Transverse cracks and bumps created by tree roots are then theoretically easier to detect. With a single scanner, data in driving direction is too sparse for the application of a sliding window, as the distances between the profiles are too large.

5.5.3. Training data

The automated way to create training data in this project was with K-means clustering. K-means clustering is not a robust clustering method. A single outlier may spoil the classification results an give different clusters. To optimise this step, the input features should only include the most relevant features. Also, initial and predefined cluster centres can be supplied to the K-means algorithm to speed up clustering and to create a higher possibility on more of the same clusters. When distinguishing different types of damage, K-means clustering did not give the best training data. K-means clustering “chooses” the clusters automatically, and it is not sure that these will exactly match the different types of damage. For example, it could also subdivide the core of craquel and the border of craquel.

Hand-made training data makes it possible to distinguish different damage types. This can be done for the initial training of the Random Forest Algorithm. When the Random Forest Algorithm is used on a next road, the classification can be checked and new training data extracted for a next survey. In this case, the training database will grow with multiple detailed damages. A possible problem of this could be that the different road types are given different feature values. This requires a lot of training to mitigate effect of different road surfaces. which result finally in a Random Forest algorithm that does not distinguish damage.

5.5.4. Feature relevance

This project focused on detecting all types of damage on a test road. But no distinction was made between different types of damage. The confusion matrices of Figure 5.15 shows that boundary damage and craquel are detected best by this method. Other damage types are more difficult to detect. For example for rutting another window length is needed to detect deviations with the surrounding area.

To make a detection algorithm for a specific damage type, more training data is needed (with a specific damage type) on different roads. A specific window length of a feature can be useful to detect one type of damage, but for another type it is useless. Despite the fact of the high correlation between window sizes, these different sizes are useful. For example, larger window lengths are useful to detect wide cracks, while smaller window lengths can be used for detecting small cracks. To increase the efficiency only the features and window lengths relevant to the damage of interest have to be evaluated. This decreases the calculation time.

5.5.5. Different road types

Brick roads have lower heights between the bricks at regular intervals. These lower heights can be seen as damages due the deviations of the surface height in comparison with the bricks. However, no problems are to be expected on asphalt with different colours.

The test road contained a part of brick road as well. Due the noncontinuous data there, some profiles where missing (Figure 5.17). Using the workflow on this part give problems.

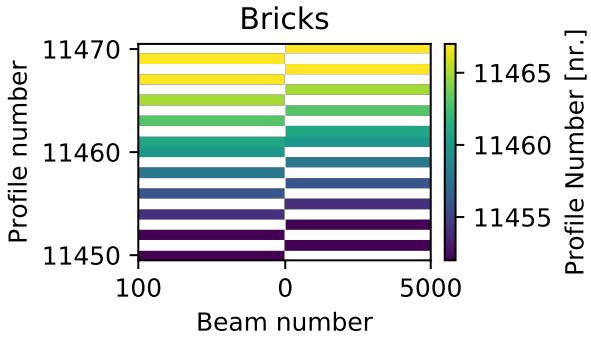


Figure 5.17: For a section of the point cloud (profile numbers: 11450-11460) profile numbers missing. The sliding window algorithm does not work on these sections.

5.5.6. Accuracy of point cloud

As mentioned in Chapter 2 the accuracy of the laser scanner varies between 0.56 mm and 3.08 mm. But in this project inaccuracies of the IMU and GPS accuracy are neglected. The empirical user range error of a single receiver GPS system depends for example on the satellite clock, the atmospheric propagation model and the receiver noise, (van der Marel, 2017). But actual values depend on the receiver, atmosphere models, time and location. When precise orbits and clocks are used from a GNSS Service or a differential GNSS set-up are used, cm level accuracy is obtained. This is much larger than the mm level of the laser scanner. For inside a measurement the errors would be correlated, but between measurements (for example driving another day) these can not combined only on the GPS locations.

Also the measurement accuracy decreases with a larger beam width due to interference of multiple object in the beam footprint. It is needed to define an optimum distribution between the beam size and the measured area.

5.6. Summary

In this chapter, the results of the method were given. First, the features were described. Here was seen that the smallest window length of 4 points, has no added value for all features. Overall with larger window lengths the craquel appears.

The standard deviation of the point density had remarkable results. Lanes with low feature values appear regularly.

Using the intensity as input gives a different feature values for two asphalt sections.

Creating training data was done in two ways: using K-means clustering as an automatic method and using a boundary value for a specific feature. With K-means clustering a noisy training set was created, while using a threshold it gave a clear training data set. Training the Random Forest with the K-means clustering resulted in decision trees based on features with intensity values as input. While a Random Forest trained with the boundary training data set created decision trees based on height values and the standard deviations of the range and point density.

Although the Random Forest classification trained with the K-means clustered data was noisy, damage locations was detected. This noise was not present when the Random Forest was trained with the boundary value clustered data. The damage locations were more clear than using the automatic K-means clustering.

Validation was attempted based on a damage classification by a road inspector, but this was too rough. So a custom validation data set was made based on the orthophotos and the classification of the road inspector. With this detailed classification it became clear that the location of the validation data did not match the point cloud data. Probably this offset has been caused by creating the orthophotos with the incorrect camera height. This resulted in very low validation values.

6

Conclusion and Recommendations

In this project a method was developed to detect road surface damage with the mobile laser scanner of IV-Infra. The emphasis of this project was on the profile laser scanner, mounted on the rear of the car. To answer the main question “How to use the Iv-Infra mobile laser scan data for road damage detection?” this final chapter will be divided in three parts: data analysis, method and validation.

6.1. Conclusion

6.1.1. Data analysis

In this project the Iv-Infra car was used for the road surface measurements. This car is all-round operable for detailed measurements. The emphasis was on one of the three laser scanners, namely the Z+F PROFILER® 9012A. This profile laser scanner measures the intensity and range. This measurements were converted to x,y,z points, but had the original beam number and profile number as well. The laser scanner range noise varied between 0.56 mm and 3.08 mm. For darker colours the noise is larger than for brighter colours. So for asphalt measurements, the noise will be larger.

The distance between profiles depends on the driving velocity of the car. When driving faster the distances will increase. Driving 30 km/h the profile spacing is around 4 cm while driving 100 km/h the spacing is already 14 cm. This can be a problem for damage detection if a damage lie between two profiles. The point density along the profile does not change when the range is constant and the points lies around 3 mm with respect to each other.

In this project an 800-meter long road section of the R106 near Haarlem, the Netherlands was used as a case study. Cracks (longitudinal and transverse), raveling, craquel, boundary damages and potholes have been found by a road inspector on this road section. In total 36 damages were marked. But on this road other road anomalies have been found as well, like road marks, speed bumps and manhole covers.

In this study, one damage was used for every damage type (found on the test road). In general, damage causes deviations in road flatness. In other words they have irregularities in road surface. In point cloud data the damages can be distinguished by higher deviations in height and intensity relative to non-damaged road surfaces. A non-damaged area is relative flat in height and has a homogeneous distribution of intensity values.

6.1.2. Method

To detect road surface damage several techniques already exist. These techniques can be categorised by the measurement technique, like vibration, image and laser scan based methods. An advantage of laser based methods is that 3D topography of the road surface can be accurately and quickly captured.

In this project profile data was used. An advantage is that the data is already organised in profile and beam numbers. This will be an advantage to the data processing speed, as no sorting is needed. That is why a sliding window is so efficient for creating the features. In this project, 6 different types of features are used:

- Absolute deviation from the mean;
- Deviation from the mean;
- Deviation from a line;
- Standard deviation of the deviation of expected range;
- Standard deviation of profile point density;
- Sum of different window lengths.

For the first three features, four different window lengths are used for the sliding window: 5, 21, 41 and 101 points. These correspond with approximately 1.6, 6.7, 13.1 and 32.3 cm. For the standard deviations a fixed window length of 20 points (6.4 cm) is used. When the height and the intensity were used as input for the first three features, this resulted in a total of 32 features.

For training the Random Forest algorithm training data is needed. The road inspectors validation data was not detailed enough, so training data was made by K-means clustering. A disadvantage of K-means clustering is that every time different training data is generated. This has an effect on the final results. Also training data was made on a non-automatic way; selecting a threshold value of a feature based on visual observations.

For the Random Forest algorithm 100 trees were used and each tree had 3 layers. To create independent trees, for each tree only a section of the features was used. After classification noise was filtered out with morphological operations and nearby damage areas were connected to each other. Using K-means clustered data for training the Random Forest, the workflow is fully automatic.

At this moment there is no distinction between different types of road damages. A possibility would be to create training data which distinguished the different types of damage and having a severity of the damage. With this training data the random forest algorithm can be trained to distinguish multiple damage types and give a value to the damage (for example low, moderate and high). But then the workflow is no longer fully automatic.

Each road type, such as asphalt road and brick roads, has its own properties. The sliding window can be used on all road types. But damage classification will be different. For example, on brick roads there are regular height deviations between the bricks. These regular deviations may have characteristics similar to damage, introducing problems in the classifications.

6.1.3. Validation

Validation of the method was done with help of the road inspectors damage classification. But when the overall accuracy and the confusion matrix were calculated, the numbers where low for a classification method, namely 73%. This was explained by the coarse validation data. The damages detected by the method are smaller than the validation data. A lot of damage from the road inspector data was not identified by the method. Also small, new damages were found by the method while the road inspector did not mentioned them. So another custom, more detailed validation set was made with orthophotos and the damage locations of the road inspector. But here was the problem that the validation data did not exactly overlapped the point cloud data due a offset between the data sets. This can explained by the used orthophotos. The orthophotos were not made with the exact camera height. But it was used for an optical validation. Visual inspection shows that the 6 damage cases were detected, but transverse cracks were difficult to detect, due to the low profile density. The space between profiles is around 4 cm while driving 30 km/h, while the point density in beam direction is around 3 mm. A transverse crack can easily fall between two adjacent profiles. Adding the two other laser scanners mounted on the rear will reduce these gaps. But these laser scanners have a different orientation and other accuracies. The georeferencing needs to be exactly the same for all scanners.

Different sizes of damage where detected with this method. These where measured in Cloud Compare (Girardeau-Montaut et al., 2017) and the smallest width was 1 cm for a longitudinal crack. Smaller

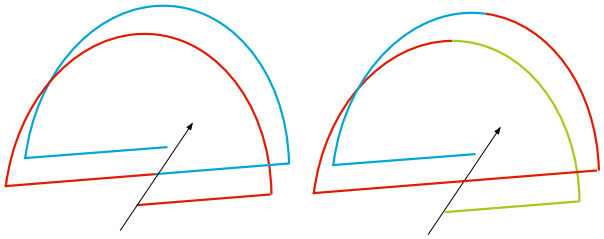


Figure 6.1: Left the current situation where a new profile is starting nadir. One colour corresponds with the same profile number. For using a sliding window, one half of the profiles is shifted one profile number. Right the recommend situation. No shift in profile numbers is needed due the fact that a profile number on the ground is connected to each other.

damages are not found with this method due the filtering after classification. Another problem occurred when the damage width is larger, then the detected damage was divided in multiple parts. It would also be interesting to know to what depth detected damage can be.

6.2. General Conclusions

Using the profile scanner of the Iv-Infra mobile laser scanner for road damage detection a sliding window algorithm is used to extract features. Then K-means clustering was used to create training data for the Random Forest classification. This is a fully automated and efficient workflow that shows promising results, but the quantitative validation of this workflow remains difficult due the offset in the validation data.

6.3. Recommendations

The proposed workflow can be used for damage detection. Although there are some recommendations apart from the topics discussed in the discussion (section 5.5).

First, an improvement on the validation data is suggested, such that the validation data is aligned with the point cloud. Then a meaningful point to point confusion matrix and overall accuracy can be calculated.

Secondly, the accuracy of the point data should be validated. This can be done by taking an exactly flat road surface and verify flatness of a known flat surface. Also different types of asphalts can be used to see the differences between them. When the noise is known, test data can be simulated and there can be experimented with the depth and width of the damage.

Finally, a small recommendation. At this moment the scanner starts a new profile nadir, while there road is located. Now half of the profiles are shifted one profile. When starting the profile zenith, the connected profiles on the road have the same number, see Figure 6.1. This can be achieved by turning the scanner 180°.

Acknowledgements

I would like to thank Iv-Infra for the opportunity to work with them on the data of the Iv-Car.

Furthermore, I would like to thank: Roderik Lindenbergh, Christian Tiberius, Cor Kasbergen, Kumar Anupam, Joost Assendelft and Sander Puister for their support during this thesis project. Thank you to Cor and Kumar for feedback and suggestions during the writing process.

Also, I am grateful to my parents for the possibility to stay in Haarlem during my internship, their love and support.

Finally, I would like to thank Adriaan van Natijne, my brothers, my friends and fellow students for their support and great time during my entire studies.

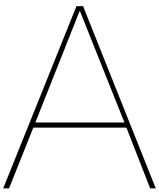
Software:

CloudCompare v2.9.1 (macOS 64-bit) is an open source software developed by Daniel Girardeau-Montaut.

GDAL (Geospatial Data Abstraction software Library) is a translator library for raster and vector geospatial data formats that is released under an X/MIT style Open Source License by the Open Source Geospatial Foundation.

PDAL (PDAL Point Data Abstraction Library) is a C++ BSD library for translating and manipulating point cloud data.

Python 3 is developed under an OSI-approved open source license, making it freely usable and distributable, even for commercial use.



ISPRS Conference Paper

The following paper was presented during the ISPRS Geospatial Week (GSW), 10-14 June 2019, at The University of Twente, the Netherlands.

MOBILE LASER SCAN DATA FOR ROAD SURFACE DAMAGE DETECTION

B. B. van der Horst[1], R. C. Lindenbergh[1], S. W. J. Puister[2]

[1] Department of Geoscience & Remote Sensing, Delft University of Technology, Delft, Netherlands - B.B.vanderHorst@student.tudelft.nl, R.C.Lindenbergh@tudelft.nl
[2] Iv-Infra, Haarlem, Netherlands - S.W.J.Puister@Iv-Infra.nl

Commission II, WG II/10

KEY WORDS: Infrastructure Monitoring, Road Pavement, Potholes, Mobile Laser Scanning, Damage Detection

ABSTRACT:

Road surface anomalies affect driving conditions, such as driving comfort and safety. Examples for such anomalies are potholes, cracks and ravelling. Automatic detection and localisation of these anomalies can be used for targeted road maintenance. Currently road damage is detected by road inspectors who drive slowly on the road to look out for surface anomalies, which can be dangerous. For improving the safety road inspectors can evaluate road images. However, results may be different as this evaluation is subjective. In this research a method is created for detecting road damage by using mobile profile laser scan data. First features are created, based on a sliding window. Then K-means clustering is used to create training data for a Random Forest algorithm. Finally, mathematical morphological operations are used to clean the data and connect the damage points. The result is an objective and detailed damage classification. The method is tested on a 120 meters long road data set that includes different types of damage. Validation is done by comparing the results to a classification of a human road inspector. However, the damage classification of the proposed method contains more details which makes validation difficult. Nevertheless does this method result in 79% overlap with the validation data. Although the results are already promising, developments such as pre-processing the data could lead to improvements.

1. INTRODUCTION

Road damage detection is important to determine road safety and road maintenance planning. The damage of the road surface, like potholes, cracks and ravelling, affects driving conditions such as driving comfort and safety and increases fuel consumption, traffic circulation and noise emission. Localisation of these damage can be used for targeted road management and maintenance, which contributes to an improvement of driver safety and comfort (Vittorio et al., 2014).

The traditional method for road condition surveying is that inspectors drive slowly on the road looking out for road surface damages and stop the vehicle when damage is found, do measurements on the damage and mark visually. This is dangerous, time-consuming and costly (Cheng & Miyojim, 1998; Yu et al., 2007). For improving the safety road inspectors can evaluate road images. The results are, however, susceptible to human subjectivity.

Iv-Infra has a mobile mapping car, shown in Figure 2, including 3 laser scanners, 10 cameras for 360° photos, a GPS and an Inertial Measurement Unit. This system has been implemented successfully for lamp post identification. This paper is an attempt to study the feasibility of using such a system for road damage detection. In this research a method for road damage detection is developed using laser scan data of one of the three laser scanners of the car, a Z+F PROFILER 9012A. This laser scanner is mounted at the rear of the vehicle such that its profile lines are perpendicular to the driving direction. It measures the range and the intensity, along the profile.

There are several advantages of such a system, for example no road closure is needed for manual road inspection, which increases safety and decreases costs. When the damage detection can be done automatically no differences due to subjective judgement are obtained.

This paper is structured as follows. In the following section advantages and disadvantages of some of the popular alternatives to manual road condition survey will be discussed. Some details about the measurement car and research area will be given in section 3. The methodology will be explained in section 4. In section 5, results of this method will be given and finally the conclusion is presented in the last section.

2. BACKGROUND

Several methods have been developed to collect data of a road surface and determine damage from such data. The methods can be classified based on how the road surface information is acquired. This can be vibration, image and laser scanning-based methods. As this study focuses on investigating the feasibility of laser scanning in detecting potholes, ravelling, crack and craquelure (Fig. 1), these definitions of these damage are first presented, and then existing methods for damage detection are investigated.

Potholes are bowl-shaped holes with various sizes involving one or more layers of the asphalt pavement structure. Size and depth can increased whenever water accumulates in the hole (Tedeschi & Benedetto, 2017). They arise due to freezing of water in a soil, which results in expanding of the space. Thawing of the soil can weaken the road surface while traffic can break the pavement resulting in potholes.

Ravelling is dislodging of aggregate particles due to influences of traffic, weather and obsolescence of the binder (Kneepens & Heesbeen, 2017; Tedeschi & Benedetto, 2017)

Due to traffic load, freezing and expanding of water in asphalt, cracks can be formed. Two types of cracks (longitudinal and transverse cracks) were considered in this study. Longitudinal cracks are cracks parallel to the road, while transverse cracks are perpendicular to the road.

Craquelure are cracks, which develop into many-sided, sharp angled pieces. This damage develops at the end of the structural life of an asphalt pavement, (Bouwend Nederland and emulsie asfaltbeton, n.d.). Craquelure at the outer 0.25 m of the pavement is named as boundary damage.

Figure 1. Examples of road surface damage.

Next, a survey of techniques for data capture and methods for processing data to determine road surface damage are present.

2.1 Vibration based methods

Accelerometers, microphones and tire pressure sensors are used to measure vibrations caused by pavement elevation differences and roughness. Accelerometers in mobile phones can measure the relative movement of the car in three dimensions. Examples are the Pothole Patrol by Eriksson et al. (2008) and Wolfrine by Bhoraskar et al. (2012). Filters and machine-learning approaches are used to detect road damages. A disadvantage of this data acquisition method is that the relative movement of the car is only influenced by the small contact area between the road surface and the four tires. So only a small parts of the road surface along the wheel paths can be analysed.

2.2 Image based methods

There are also methods collecting images from scanning, line-scan and video cameras of the road surface, which can be used for detecting the damage. An example is the automated detection system RoadCrack, created by the Australian Commonwealth Scientific and Industrial Research Organization (CSIRO, n.d.). This system is based on high speed cameras mounted underneath the vehicle. These cameras collect high resolution images of small patches of the pavement surface and they are consolidated into bigger images of half-metre intervals. CSIRO (n.d.) stated that the system can detect cracks in a millimetre order, while driving up to 105 kilometres per hour. This is done fully-automated with a combination of machine vision and artificial intelligence (CSIRO, n.d.). Another system based on laser based imaging is the Digital Highway Data Vehicle (DHDV) from Waylink (n.d.). They use their Automated Distress Analyzer (ADA) which produces crack maps in real time.

RoadCrack and DHDV are two commercial systems, which use cameras as one of their acquisition methods. There are several more commercial systems, most of which have not published details on their algorithm.

2.3 Laser scan based methods

One of the advantages of using laser scanning sensors is that 3D topographic of the road surface can be captured highly accurately and quickly. Guan et al. (2014) used mobile laser scanning (MSL) data to detect road markings. From MSL data, they create intensity images, which they used in a point-density-dependent multi-threshold segmentation method to recognise road markings.

Pavemetric inc. developed the Laser Crack Measurement System (LCMS), which consists of two high performance 3D laser profilers and a camera as detector, in cooperation with government and research partners (Laurent et al., 2014). This system measured range and intensities, and produced 2D and 3D data.

Yu et al. (2007) developed a system using a SICK LMS 200 laser scanner for reconstructing the 3D surface model, cracks in smaller regions can be identified from a variation of the 3D depth measurement.

Mertz (2011) used a low cost “laser line striper” to evaluate the unevenness of the road with a step-operator to detect road damage. Based on the number of the data points in one line, significant road damage is found. However, noise data can trigger the larger number of the points in the line, which lead to incorrect damage to be detected.

3. DATA

3.1 Data acquisition system

As mentioned in section 1, Iv-Infra has a measurement car with 3 laser scanners, 10 cameras for 360° photos, 3 HR cameras in the bumper, a GPS and a IMU system (Fig. 2).

Figure 2. Measurement car

In this research, the data from one scanner, the Z+F PROFILER 9012A is used. This is a profile scanner using the the phase-shift method for measuring the range. An outgoing laser beam is intensity-modulated by a sine-wave signal. This signal is reflected back from an object and the received intensity pattern is compared with the original transmitted signal. A phase-shift in the modulated signal is caused by the travelling time of light forth and back to the measured object. The phase measurement can be transformed directly into a distance/range: $d = \frac{c}{2*f}$, with c the speed of light (with atmospheric corrections) in m/s, f the modulated frequency in Hz.

The profile scanner produces measurement points with x, y, z coordinates. Each measurement point (x,y,z) is geo-referenced by the QINSy (Quality Integrated Navigation System) software (Quality Positioning Services B.V, 2018) such that the IMU, the GNSS locations, the vehicle odometer, the intensity and range are taken into account. This is done in the Dutch coordinate system, RD-coordinates. The z component is given in Normaal Amsterdams Peil (NAP), the Dutch height reference. Each measurement point contains the following data fields: intensity, range, profile number and beam number. The intensity is the amount of reflected light, which has no clear unit. The range is the distance between the scanner and a hit point on the object surface and it is given in meters. When the laser beam hits multiple "targets" of different heights, for example when the laser beam partly hits the road surface and partly falling into a crack, the laser scanner will detect a combination of multiple reflections, one for each target. Unfortunately phase-based ranging devices can never discern all the single vectors but only measure the resultant vector; the geometrical sum of all vectors. So the resultant range is a mixture of the distances to the surface and into the crack (Mettenleiter, M. (Zoller + Fröhlich GmbH), 2019).

A profile number is given to each new line which the profiler measures. A new profile starts nadir and the laser beam turns anticlockwise, see Figure 3. The beam numbers are given to each consecutive point in each profile. In this project, the laser scanner is configured such that each profile (360°) contains 5100 points (beams), with a spindle speed of 200 rotations per second (profiles). When the car is driving, a spiral pattern is formed, illustrated in Figure 3. The distance between each profile depends on the car velocity and the spindle speed of the laser scanner. In this case, this results in a distance of 4 cm between the profiles while driving 30 km/h and 14 cm at 100 km/h. The point spacing along the profile is approximately 3 mm on the road in nadir direction and does not depend on driving velocity, but on range.

3.2 Research area

A road section of the R106 near Haarlem city, the Netherlands, is selected for a pilot survey. This is a touristic and quiet road where the driving speed is between 30 and 50 km/h. On this road, 36 road damages are found by a road inspector from a third party and are categorised as 2 ravelling, 7 craquelure, 2 potholes, 8 longitudinal and 11 transverse cracks and 6 boundary damage, (van den Assem, 2019). Figure 4 shows the damage of the road as classified by a road inspector. For this paper a subset of around 120 meters of road is used, which includes 6 million points, given in Figure 4. Road sections 1 and 2 are evaluated extensively in Section 5.5.

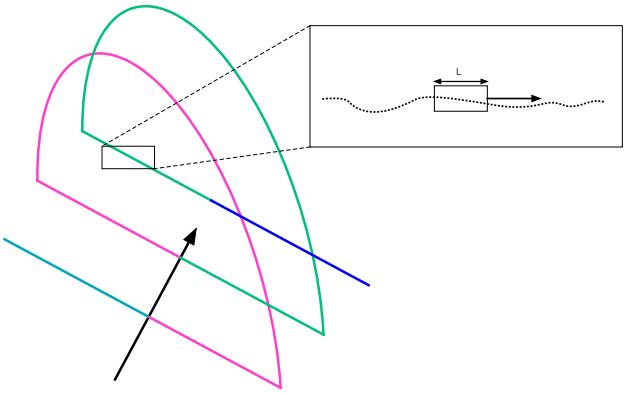


Figure 3. Scanning pattern of the profile. Each new colour represents a new profile number. The driving direction is marked with an arrow. The zoomed in section gives how the sliding window algorithm is used.

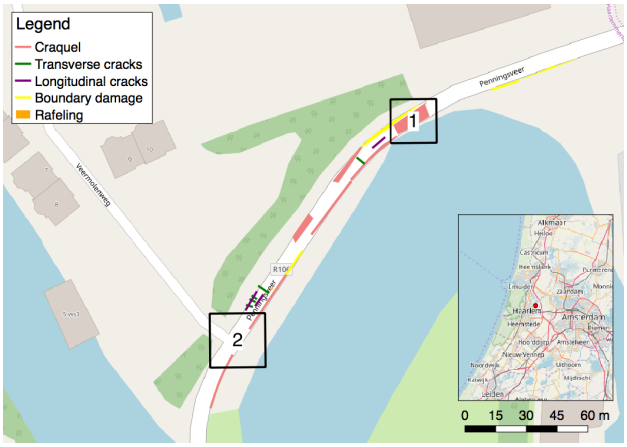


Figure 4. Part of the research area with locations of damages found by a road inspector. The black squares show the locations which are discussed in more detail.

3.3 Data selection

The laser beam width defines which sizes of damages can be measured. A large beam is more likely to hits multiple "targets" which results in a resultant vector. Therefore, it was decided to use only beam widths smaller than 5 mm for this research. To avoid that the beam widths are larger than 5 mm, the theoretic intersection of the laser beam with a horizontal was calculated based on trigonometric properties. For this laser scanner the beam divergence is 0.5 mrad and it has a beam diameter of 1.9 mm (at 0.1 m distance) (Zoller + Fröhlich GmbH, n.d.). In Figure 5 it can be seen that at 42 degrees the beam width is below 5 mm, so this is taken as the boundary angle. This results in around 600 beam numbers on each side of the nadir, when there are 5100 in one profile.

4. METHODOLOGY

To identify damage of the road surface from MLS data, the proposed workflow includes (I) feature creation, (II) K-means clustering to create training data set, (III) Random Forest classification and (IV) Mathematical morphological operations to reduce small damage points and connect larger damage points.

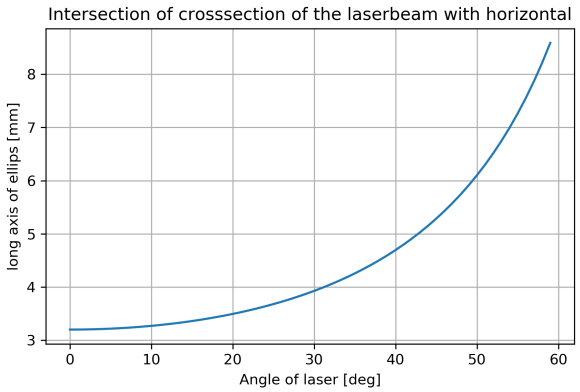


Figure 5. Effect of the angle of the beam on the beam width. A larger angle causes a larger beam width.

4.1 Step I: Feature creation

Various independent features are made with a sliding window algorithm. A sliding window algorithm means that a window with size L is moving along the points, in this case along the profile, see figure 3. In this research the results of the calculation of a window are given to the centre point of that window. Notably, feature values of the points strongly depend on the window size. An overview of the six different features and their calculations is given.

4.1.1 Deviation from the mean The first feature is to calculate the absolute elevation deviation of the centre point from the mean of a window of length L. This can also be written as:

$$\Delta Z = \mid Z_{\frac{L+1}{2}} - \frac{1}{L}\sum_{i=1}^{L} Z_i \mid, \tag{1}$$

where Z = the point data
and L = the number of the points within the window.

This can be interpreted as the surface roughness, which can be defined as the irregularities in the surface texture which are inherent in the production process and wear (Taylor Hobson Limited, 2011).

This feature can also be calculated for height values as well as with the intensity values.

4.1.2 Difference to the surrounding points Another method is to take the difference with the centre point of a window and the neighbouring points along the profile. This can written as:

$$(Z_{\frac{L+1}{2}} \cdot L) - \sum_{i=1}^{L} Z_i. \tag{2}$$

This feature can be used with the height values as well with the intensity values.

4.1.3 Standard deviation of range In this feature the range is calculated for each beam number when the road would be horizontal and flat. This is done by calculating the angle which each beam number should have, by taking the fraction of the beam number by the maximum beam number times 360°. The range is then calculated by the height of the scanner divided by the cosine of the above calculated angle. This calculated range is subtracted from the measured range. This is done because the range may vary with the angle. Over a window with length 20 points the standard deviation is taken over the range difference.

4.1.4 Standard deviation of number of points With CloudCompare (Girardeau-Montaut et al., 2017) the number of neighbours inside a sphere of radius R are calculated for each point. In this case a radius of 0.02 metre is taken. Here the standard deviation is also taken over a window of length 20 points.

4.1.5 Sum of different windows For the deviation from the mean and difference with surrounding points different window sizes can be used to calculate the feature. The results of different window lengths are added as a separate feature a new feature is created.

4.2 Step II: K-means clustering

In this step K-means clustering is used to create a training data set for the Random Forest classification. K-means clustering, (Hartigan & Wong, 1979), divides M points in N dimensions into K clusters so that each point belongs to the cluster with the closest centroid.

In this study, K-means clustering is used to classify a small selection of the data with known damage into two clusters (“no damage” and “damage”).

But before the clustering is done, each feature is scaled. This is done by first subtracting the mean value, and scale it by the standard deviation of the feature.

Both scaling and clustering are done with the python scikit-learn module (Pedregosa et al., 2011).

4.3 Step III: Random Forest classification

After clustering, the small training data set can be used for training the Random Forest algorithm. Random Forest Classification is a supervised classification method, based on classification trees (Liaw et al., 2002). A classification tree is a multistage approach which breaks up a complex decision into a union of several simpler decisions (Safavian & Landgrebe, 1991). Each node in a tree makes a binary decision, and multiple decisions in a tree lead to a class label. This is done by dividing the small training data set in 3 parts, and use 1 part for training the algorithm.

In this research, the RandomForestClassifier from the scikit-learn module is used (Pedregosa et al., 2011). After training, the whole data set is classified by using this random forest classifier.

4.4 Step IV: Mathematical morphological operations

With the Python scikit-image (van der Walt et al., 2014) morphology module, objects smaller than 3 points are removed as a first step. This is done by projecting the data as a matrix with the number of profiles as rows and the number of beams as columns. After the small objects are removed, morphological

closing is used. Mathematical morphological operations assign pixels in an image based on the values of neighbouring pixels. Mathematical morphological closing is a combination of dilation followed by an erosion operation (Smith, 1997). Dilation is an operation, which changes a "no damage" pixel into a "damage" pixel when a neighbouring pixel is classified as "damage". Erosion is the opposite operation of dilation. Erosion gives "damage" pixels a "no damage" value when the neighbouring pixels are classified as "no damage". Erosion decreases objects, while dilation increases objects, and can merge multiple objects into one (Smith, 1997). Mathematical morphological closing removes gaps in connected damage pixels. As neighbouring pixels a + shape of 1 centre point is used.

5. RESULTS

In this section, results from the proposed method are presented. Furthermore were the results validated based on the classification of a road inspector from a third party with no connection to this project.

5.1 Features

For this research 22 features are calculated as described above. For the deviation from the mean and difference to the surrounding points four different window sizes (5, 21, 41, 101) are used. Figure 6 illustrates the correlation between the features. It is clear that there is a large correlation between different window sizes of the same feature.

Examples of different features for road section 1, can be found in Figure 8.

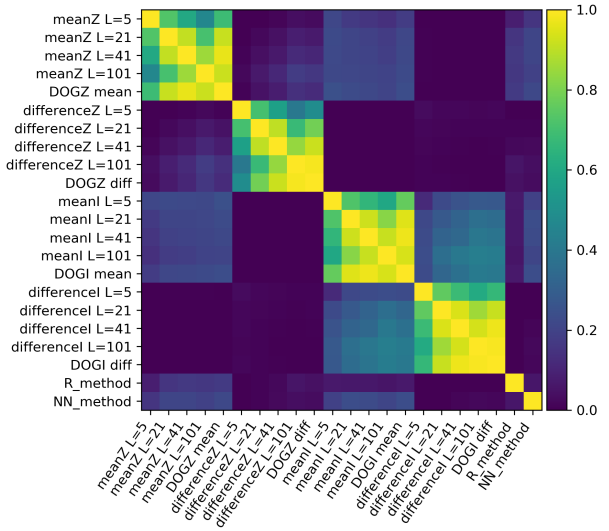


Figure 6. Correlation matrix of the features. L gives the window size which is used for calculating the feature.

5.2 K-means clustering

K-means clustering is done on road section 1. Results for this road section are given in Figure 9. From this figure it can be seen that large longitudinal cracks are classified as damage, while the transverse cracks are not detected.

5.3 Random Forest

The centre figure of Figure 9 gives the results of the Random Forest classification and morphological operations for road section 1, and Figure 10 gives the road damage classification for the whole research area. Figure 9 shows that less small objects are present in compare with the K-means clustering result. The transverse crack is as well not detected as damage.

5.4 Validation

The validation of the above described method is done with the help of damage shapefiles of a road inspector from a third party. The shapefiles are three files with point, line and polygon data. These data files are converted to raster data with the GDAL (GDAL/OGR contributors, 2018) tool gdal_rasterize. This tool rasterizes the shapefile (vector geometries) with a pixel size of 0.05×0.05 meter. Then the three raster files are combined to one large raster file.

This validation data is projected to the point data, such that each point gets a damage value. The areas of connected damage points are calculated, such that orthogonal and diagonal point neighbours are included. This is also done for the method data. Through rasterising the shapefiles, some pixels are no longer connected to each other, such that the number of damage areas are increased. This results in 153 connected road damages instead of 20 damages. The results of the method contains 3512 damage areas, most of them are smaller than 30 points. The distribution of the damage areas (below 30 points) for the validation data and the method are given in Figure 12. Here it can be seen that there is a large amount smaller damages detected for the described method, and less for the road inspector. When the larger areas (>30 points) are compared, there are 139 damage areas for the method and 62 validation damage areas.

When the intersection of union should be calculated, this would result in a low number. The intersection of union can be calculated by the area of overlap divided by the area of union. This can explained by the large and rough damage areas of the road inspector. The area of union is large, while the classified damage of the method are detailed and relative small. So for calculating the intersection of union another more detailed validation data is needed. This can be done by taking orthogonal photos of the road and take the road inspector's classification as guide.

When only method damage points are compared with validation method points, 79% of the method points are right classified as damage. However, the question is whether the false positives really are false positives.

5.5 Cases

In this section, two cases are discussed in detail, road sections 1 and 2 in Figure 4.

In road section 1 (Fig. 10, between profile numbers 19360 and 19600), the proposed method classified only parts as damage, while the road inspector (Fig. 11) the whole area classified as damage.

Road section 2 (Fig. 10, between profile numbers 17500 and 17800) contains road markings classified as damage by the proposed method. This can be explained by the higher elevation of road markings and the higher intensity compared to the surrounding points. Therefore it is important to pre-process the point data, for example filter out high intensities.

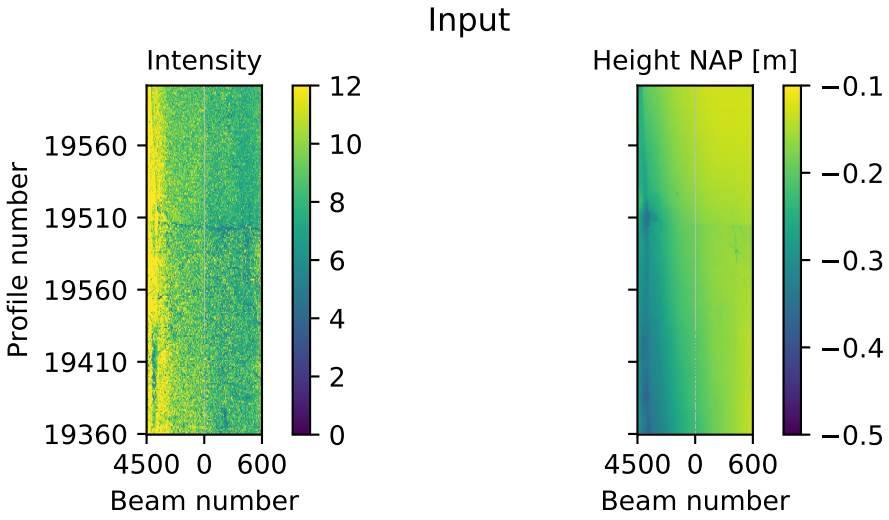


Figure 7. Input data, left the intensity, right the height in NAP [m]

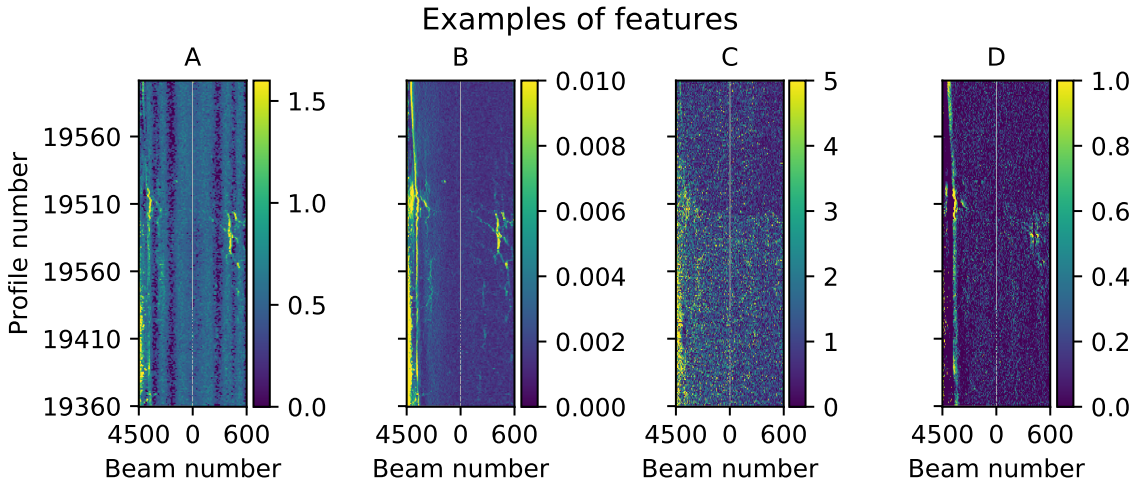


Figure 8. Examples of features. A: standard deviation of the numbers of neighbours, B: standard deviation of the range difference, C: deviation from the mean of Intensity with window size 41, D: Sum of different windows of the difference to the surrounding points

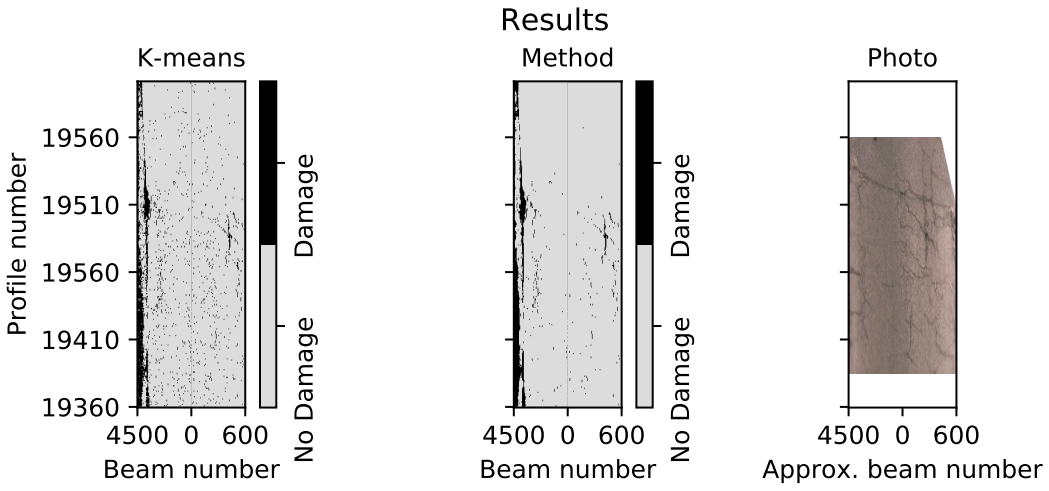


Figure 9. Results, left: the K mean classification, centre: the method classification, right: a photo of the damage

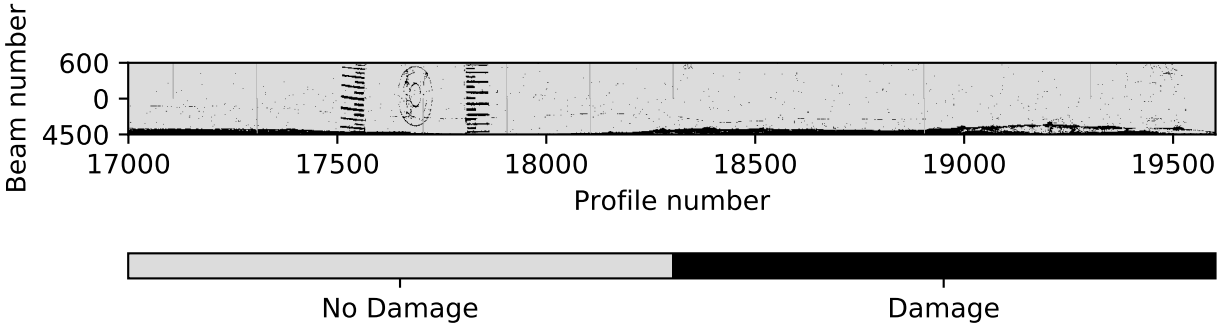


Figure 10. Road damage classification of the road with the method

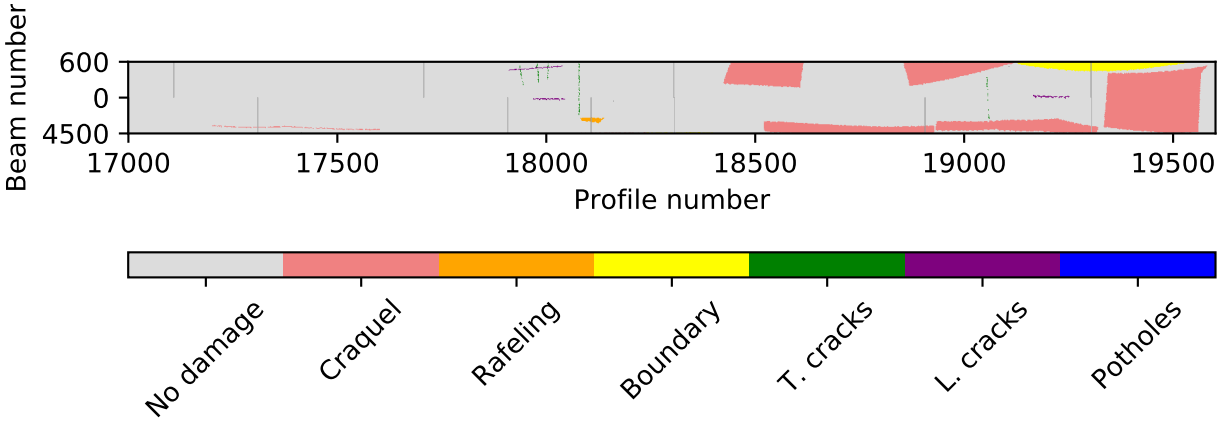


Figure 11. Road damage classification of the road of the road inspector

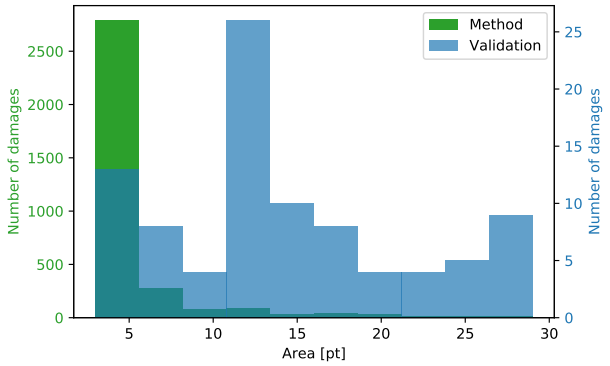


Figure 12. Damage area distribution of validation and method damages.

Also transverse cracks are difficult to detect with this method. The change that they are not detected is large, because the spacing between profiles is relative large, especially when the driving speed is high.

6. CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion

This paper presented a possible technique for detecting road damages with a Z+F PROFILER 9012A laser scanner mounted on a mobile mapping system. This is done by making features with a moving window method. Then K-means clustering is applied to create training data for a Random Forest algorithm. After that classification mathematical morphological operations are used to remove small objects and connect points which are close to each other. Validation is done with the help of a road inspector's classification. Although this validation data is too rough to calculate the intersection of union on areas, when points are compared to each other 79% of the points are correct classified as damage. However, more research is needed for analysing the false positives.

Also road markings are classified as damages, probably due to the high reflectivity and due the fact that road markings are elevated with respect to the road which results in a deviation when compared to the surrounding road surface.

Due to the spacing between profile lines, the probability is large that transverse cracks are not detected.

6.2 Recommendations

To create a higher level of accuracy for this method pre-processing of the point data is needed in order to remove road markings

for example. To remove the K-means classification for extracting training data for the Random Forest algorithm self made training data can be used. Also a new and more detailed road classification can be used for validation. This can be done by making a detailed road damage classification by help of orthogonal road photos. Further, a look at the false positives is needed and a confusion matrix can be used to distinguish which damages types can and can not be recognised well. A next step in this research could be to identify different types of damages.

References

Bhoraskar, R., Vankadhara, N., Raman, B., Kulkarni, P., 2012. Wolverine: Traffic and road condition estimation using smartphone sensors. *Fourth International Conference on Communication Systems and Networks (COMSNETS 2012)*.

Bouwend Nederland and emulsie asfaltbeton, n.d. Red mijn weg. Retrieved from `www.redmijnweg.nl`. Date Accessed: April 01, 2019.

Cheng, Heng-Da, Miyojim, Mario, 1998. Automatic pavement distress detection system. *Information Sciences*, 108, 219–240.

CSIRO, n.d. Road crack detection. Retrieved from `www.csiro.au`. Date Accessed: April 01, 2019.

Eriksson, Jakob, Girod, Lewis, Hull, Bret, Newton, Ryan, Madden, Samuel, Balakrishnan, Hari, 2008. The pothole patrol: using a mobile sensor network for road surface monitoring. *MobiSys'08 - International Conference on Mobile Systems, Applications, and Services*, 29–39.

GDAL/OGR contributors, 2018. GDAL/OGR geospatial data abstraction software library. Open Source Geospatial Foundation. `gdal.org` (December 2018).

Girardeau-Montaut, Daniel et al., 2017. Cloudcompare (version 2.9.1). Retrieved from `www.cloudcompare.org`.

Guan, Haiyan, Li, Jonathan, Yu, Yongtao, Wang, Cheng, Chapman, Michael, Yang, Bisheng, 2014. Using mobile laser scanning data for automated extraction of road markings. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87, 93 - 107.

Hartigan, John A, Wong, Manchek A, 1979. Algorithm AS 136: A K-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28, 100–108.

Kneepkens, A. G., Heesbeen, J., 2017. Rotating Surface Abration Test (RSAT): Rafeling wordt voorspelbaar en te voorkomen. *Civile Techniek*, 3.

Laurent, John, Savard, Yves, Lefebvre, Daniel, 2014. 3D laser road profiling for the automated survey of road surface conditions and geometry. *IRF*, 30–34.

Liaw, Andy, Wiener, Matthew et al., 2002. Classification and regression by Random Forest. *R news*, 2, 18–22.

Mertz, Christoph, 2011. Continuous road damage detection using regular service vehicles. *Proceedings of the ITS world congress*, 5–8.

Mettenleiter, M. (Zoller + Fröhlich GmbH), 2019. private communication.

Pedregosa, F. et al., 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

Quality Positioning Services B.V, 2018. Qinsy (version 8.18.3).

Safavian, S Rasoul, Landgrebe, David, 1991. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, 21, 660–674.

Smith, Steven w., 1997. *The scientist and engineer's guide to digital signal processing*. California Technical Publishing.

Taylor Hobson Limited, 2011. *Exploring Surface Texture - A fundamental guide to the measurement of surface finish*. Taylor Hobson Limited.

Tedeschi, A, Benedetto, F., 2017. A real-time automatic pavement crack and pothole recognition system for mobile Android-based devices. *Advanced Engineering Informatics*, 32, 11–25.

van den Assem, Jason, 2019. private communication.

van der Walt, Stéfan et al., 2014. scikit-image: image processing in Python. *PeerJ*, 2.

Vittorio, Astarita, Rosolino, Vaiana, Teresa, Iuele, Vittoria, Caruso Maria, Vincenzo, P Giofrè et al., 2014. Automated sensing system for monitoring of road surface quality by mobile devices. *Procedia-Social and Behavioral Sciences*, 111, 242–251.

Waylink, n.d. Waylinks digital highway data vehicle. Retrieved from `http://waylink.com/`. Date Accessed: April 01, 2019.

Yu, Si-Jie, Sukumar, Sreenivas R, Koschan, Andreas F, Page, David L, Abidi, Mongi A, 2007. 3D reconstruction of road surfaces using an integrated multi-sensory approach. *Optics and lasers in engineering*, 45, 808–818.

Zoller + Fröhlich GmbH, n.d. Laser measurement technology. Retrieved from `www.zf-laser.com`. Date Accessed: April 01, 2019.

B

GRS-GSE Research Day poster

The following poster was presented during the GRS-GSE Research Day, 15 April 2019, in Delft, the Netherlands.

MOBILE LASER SCAN DATA FOR ROAD SURFACE DAMAGE DETECTION

B. B. van der Horst[1], R. C. Lindenbergh[1] , S. W. J. Puister[2]

[1] Department of Geoscience & Remote Sensing, Delft University of Technology, Delft, Netherlands
[2] Iv-Infra, Haarlem, Netherlands

Introduction

- **Road surface damage** detection is important to determine **road safety** and road **maintenance planning**.
- It affects **driving conditions** such as **driving comfort, safety**, increases **fuel consumption**, **traffic circulation** and **noise emission**.

- **Traditional method** of of road inspectors is driving and stop on the road. This is **dangerous**, **time-consuming** and **costly**.
- **Image based** detection is **susceptible to human subjectivity**.

Goal

An automatic method for road damage detection is developed using mobile laser scan data of a Z+F PROFILER 9012A.

Car

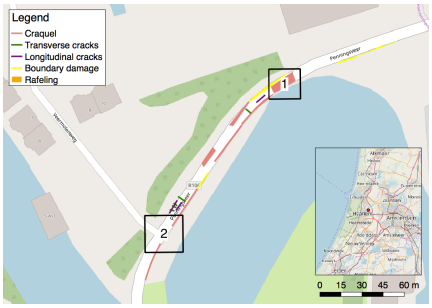
Iv-Infra has a measurement car with

- 3 laser scanners,
- 10 cameras for 360° photos,
- 3 HR cameras in the bumper,
- a GPS and
- a IMU system

Test road

- Subset of regional road, R106
- Near Haarlem, the Netherlands

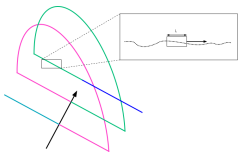
- **20 damages** classified by independent road inspector.



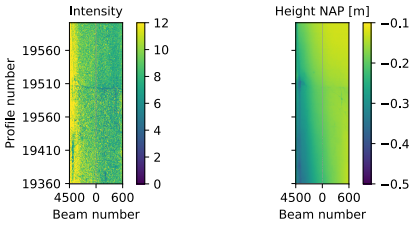
Examples of damages

Data

- x, y, z locations
- intensity,
- range,
- profile number,
- beam number.



- each profile (360°) contains 5100 points (beams), with a spindle speed of 200 rotations per second (profiles).



Methodology

I) feature creation with sliding windows
   A. Deviation from the mean:

$$\Delta Z = \left| Z_{\frac{L+1}{2}} - \frac{1}{L}\sum_{i=1}^{L} Z_i \right|$$

   B. Difference to the surrounding points:

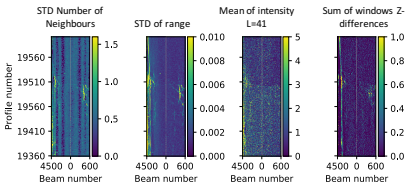
$$\Delta Z = \left(Z_{\frac{L+1}{2}} \cdot L \right) - \sum_{i=1}^{L} Z_i$$

   C. Standard deviation of the range
   D. Standard deviation of the # of neighbours
   E. Sum of different window sizes

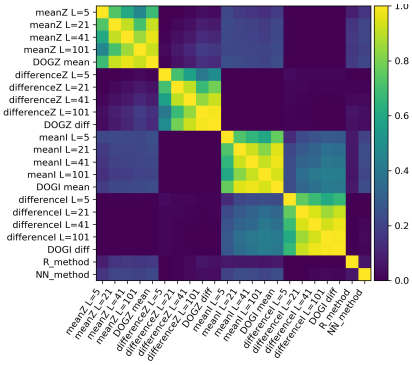
II) K-means clustering to create training data set,
III) Random Forest classification,
IV) Mathematical morphological operations to reduce small damage points and connect larger damage points

Results

Features:

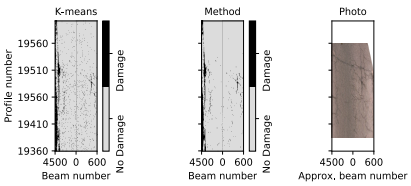


Correlation between features

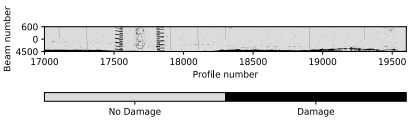


Classification:

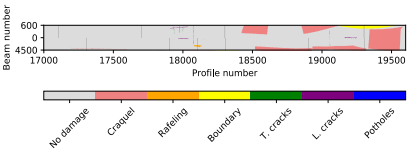
Section 1:



Method classification:



Validation set by a human road inspector:



Conclusion

- **Promising** results
- **Specific** maintenance
- Insensitive to human **subjectivity**
- No road block
- **Safety**
- **Automatic**

C

ISPRS Geospatial week 2019 poster presentation

The following poster was presented during the ISPRS Geospatial week 2019, 12 June 2019, in Enschede, the Netherlands.

MOBILE LASER SCAN DATA FOR ROAD SURFACE DAMAGE DETECTION

B. B. van der Horst[1], R. C. Lindenbergh[1] , S. W. J. Puister[2]

[1] Department of Geoscience & Remote Sensing, Delft University of Technology, Delft, the Netherlands
[2] Iv-Infra, Haarlem, Netherlands

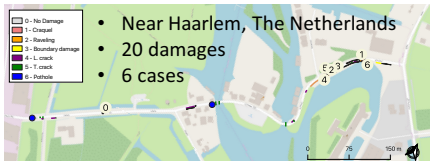
Introduction

- **Road surface damage** detection is important to determine **road safety** and road **maintenance planning**.
- It affects **driving conditions** such as **driving comfort, safety**, increases **fuel consumption**, **traffic circulation** and **noise emission**.
- Traditionally, road inspectors stop on the road to inspect possible damage. This is **dangerous**, **time-consuming**, **costly** and **susceptible to human subjectivity**.
- **Image based** detection is

Goal

Automatic detection of road damage from mobile laser scan data.

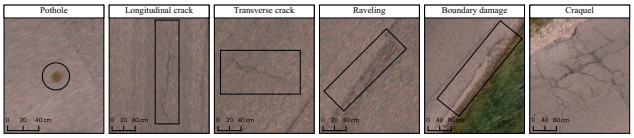
Test road



- Near Haarlem, The Netherlands
- 20 damages
- 6 cases

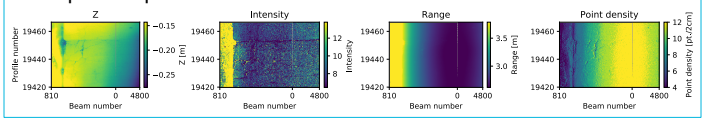
IV's mobile mapping system

Damage types



LiDAR Data

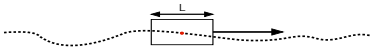
Example: craquel



Method

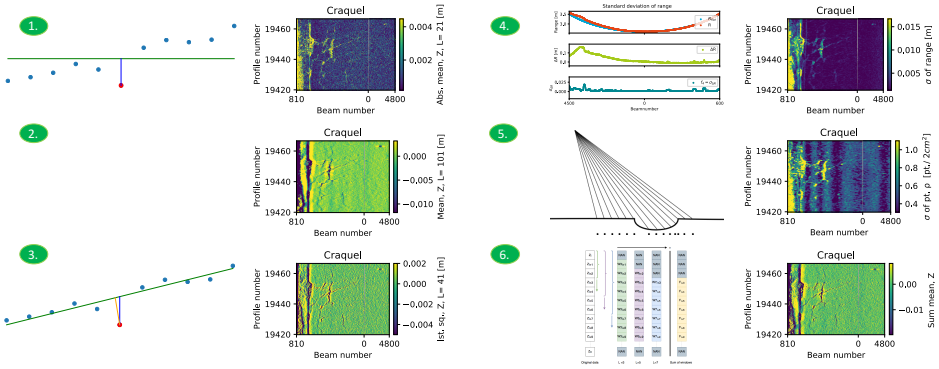
I) Feature Extraction using sliding window

- Efficient sliding window approach:



1. Absolute deviation to the mean
2. Deviation to the mean
3. Ort. deviation to the linear least square
4. Std. of the deviation of expected range
5. Std. of the point density
6. Sum of different window lengths

- Different window lengths to detect different damage sizes



II) Trainings Data Refinement

- Traditional classification is not detailed enough
- Kmeans Clustering, two clusters

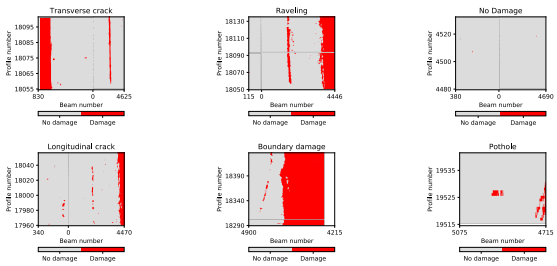
III) Random Forest Classification

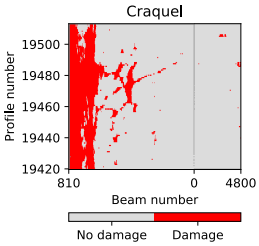
- 100 trees, 3 layers per tree
- $\sqrt{N}$ features per tree for independent trees

IV) Morphological Filtering

1. Original data 2. Remove small objects 3. Dilation 4. Erosion

Results





Conclusion

- New very detailed validation data is necessary
- **Promising** results
- **Targeted** maintenance planning
- Insensitive to human **subjectivity**
- No road block necessary
- **Fully Automated**
- Computationally **efficient**

Traffic sign: https://www.pngfly.com/png-zx9hdw/

Bibliography

G. Alessandroni, L.C. Klopfenstein, S. Delpriori, M. Dromedari, G. Luchetti, B. Paolini, A. Seraghiti, E. Lattanzi, V. Freschi, A. Carini, and A. Bogliolo. Smartroadsense: Collaborative road surface condition monitoring. *Proc. of UBICOMM-2014. IARIA*, pages 210–215, 2014.

ARCADIS. Private Communication, 2019.

BAM Infra. Bigger data in asset management - de groeiende mogelijkheden van automatische schadeherkenning, 2018. Retrieved June 9, 2019, from https://bit.ly/2ItZJBD.

M. Belgiu and L. Drăguț. Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing*, 114:24 – 31, 2016. doi: https://doi.org/10.1016/j.isprsjprs.2016.01.011. URL http://www.sciencedirect.com/science/article/pii/S0924271616000265.

R. Bhoraskar, N. Vankadhara, B. Raman, and P. Kulkarni. Wolverine: Traffic and road condition estimation using smartphone sensors. *2012 Fourth International Conference on Communication Systems and Networks (COMSNETS 2012)*, Jan 2012. doi: 10.1109/COMSNETS.2012.6151382. URL http://ravi.bhoraskar.com/papers/wolverine.pdf.

Bouwend Nederland. Craquelé, 2016. Retrieved June 9, 2019, from http://www.redmijnweg.nl/schadebeelden/.

J. Brownlee. What is a confusion matrix in machine learning, 2016. Retrieved June 24, 2019, from https://machinelearningmastery.com/confusion-matrix-machine-learning/.

H.D. Cheng and M. Miyojim. Automatic pavement distress detection system. *Information Sciences*, 108 (1):219 – 240, 1998. ISSN 0020-0255. doi: https://doi.org/10.1016/S0020-0255(97)10062-7. URL http://www.sciencedirect.com/science/article/pii/S0020025597100627.

CSIRO. Road crack detection, ca. 2000. Retrieved June 9, 2019, from https://research.csiro.au/qi/road-crack-detection/.

J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan. The pothole patrol: Using a mobile sensor network for road surface monitoring. *MobiSys'08 - International Conference on Mobile Systems, Applications, and Services*, pages 29–39, 2008. Retrieved June 9, 2019, from http://nms.csail.mit.edu/papers/p2-mobisys-2008.pdf.

E.S. Gadelmawla, M.M. Koura, T.M.A. Maksoud, I.M. Elewa, and H.H. Soliman. Roughness parameters. *Journal of Materials Processing Technology*, 123(1):133 – 145, 2002. ISSN 0924-0136. doi: https://doi.org/10.1016/S0924-0136(02)00060-2. URL http://www.sciencedirect.com/science/article/pii/S0924013602000602.

M. Gavilán, D. Balcones, O. Marcos, D.F. Llorca, M.A. Sotelo, I. Parra, M. Ocaña, P. Aliseda, P. Yarza, and A. Amírola. Adaptive road crack detection system by pavement classification. *Sensors*, 11(10): 9628–9657, 2011.

D. Girardeau-Montaut et al. Cloudcompare (version 2.9.1), 2017. Retrieved from www.cloudcompare.org.

H. Guan, J. Li, Y. Yu, C. Wang, M. Chapman, and B. Yang. Using mobile laser scanning data for automated extraction of road markings. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87:93 – 107, 2014. ISSN 0924-2716. doi: https://doi.org/10.1016/j.isprsjprs.2013.11.005. URL http://www.sciencedirect.com/science/article/pii/S0924271613002657.

J.A. Hartigan and M.A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.

A.G. Kneepkens and J. Heesbeen. Rotating surface abration test (rsat): Rafeling wordt voorspelbaar en te voorkomen. *Civile Techniek*, 3, 2017. Retrieved June 9, 2019, from http://www.redmijnweg.nl/sites/default/files/files/2017%2003%20-%20CT%20-%20RSAT.pdf.

W. Koehrsen. Random forest simple explanation, 2017. Retrieved July 4, 2019, from https://medium.com/@williamkoehrsen/random-forest-simple-explanation-377895a60d2d.

V. Y. Kulkarni and P. K. Sinha. Pruning of random forest classifiers: A survey and future directions. *2012 International Conference on Data Science Engineering (ICDSE)*, pages 64–68, 2012. doi: 10.1109/ICDSE.2012.6282329.

J. Laurent, Y. Savard, and D. Lefebvre. 3d laser road profiling for the automated survey of road surface conditions and geometry. *IRF Examiner*, 2:30–34, 2014. URL https://www.irf.global/ebooks/IRF-Examiner-14Vol2.pdf#page=30.

Lexico powered by Oxford. Definition of overfitting in english by lexico dictionaries, n.d. Retrieved June 11, 2019, from https://www.lexico.com/en/definition/overfitting.

A. Liaw and M. Wiener. Classification and regression by random forest. *R news*, 2(3):18–22, 2002.

C. Mertz. Continuous road damage detection using regular service vehicles. *18th ITS World Congress*, 2011.

M. Mettenleiter. Zoller + Fröhlich GmbH, private communication, 2019.

J.S. Miller and W.Y. Bellinger. *Distress Identification Manual for the Long-Term Pavement Performance Program*, June 2003. Retrieved June 9, 2019, from https://rosap.ntl.bts.gov/view/dot/28612.

N.K. Mubaris. K-means clustering in python, 2017. Retrieved June 9, 2019, from https://mubaris.com/posts/kmeans-clustering/.

OpenStreetMap contributors. Planet dump retrieved from https://planet.osm.org . https://www.openstreetmap.org, 2017.

Pavement Tools Consortium. Bleeding, n.d. Retrieved June 9, 2019, from https://www.pavementinteractive.org/reference-desk/pavement-management/pavement-distresses/bleeding/.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.

M. Perttunen, O. Mazhelis, F. Cong, M. Kauppila, T. Leppänen, J. Kantola, J. Collin, S. Pirttikangas, J. Haverinen, T. Ristaniemi, and J. Riekki. Distributed road surface condition monitoring using mobile phones. *International Conference on Ubiquitous Intelligence and Computing*, 6905:64–78, 2011.

F. Pirotti and F. Tonion. Classification of aerial laser scanning point clouds using machine learning: A comparison between random forest and tensorflow. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W13:1105–1111, 2019. doi: 10.5194/isprs-archives-XLII-2-W13-1105-2019. URL https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2-W13/1105/2019/.

Quality Positioning Services B.V. Qinsy (version 8.18.3), 2018.

S.R. Safavian and D. Landgrebe. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, 21(3):660–674, 1991.

F. Seraj, B.J. van der Zwaag, A. Dilo, T. Luarasi, and P. Havinga. Roads: A road pavement monitoring system for anomaly detection using smart phones. *Proceedings of the 1st International Workshop on Machine Learning for Urban Sensor Data, SenseML 2014*, pages 128–146, 9 2016.

S.W. Smith. *The scientist and engineer's guide to digital signal processing*. California Technical Publishing, 1997. Retrieved June 9, 2019, from www.DSPguide.com.

S.S. Soudarissanane. *The geometry of terrestrial laser scanning; identification of errors, modeling and mitigation of scanning geometry*. PhD thesis, Delft University of Technology, 2016.

Taylor Hobson Limited. *Exploring surface texture: a fundamental guide to the measurement of surface finish*. Taylor Hobson Limited, 7th edition, 2003. URL https://www.taylor-hobson.com/-/media/ametektaylorhobson/files/learningzone/exploringsurfacetexture2014.pdf?la=en.

A. Tedeschi and F. Benedetto. A real-time automatic pavement crack and pothole recognition system for mobile android-based devices. *Advanced Engineering Informatics*, 32:11 – 25, 2017. ISSN 1474-0346. doi: https://doi.org/10.1016/j.aei.2016.12.004. URL http://www.sciencedirect.com/science/article/pii/S1474034616301197.

C.C.J.M. Tiberius. *Primer on Mathematical Geodesy, CTB3310 / CTB3425*. Faculty of Civil Engineering and Geosciences, 2015.

H. van der Marel. Lecture notes in gps for civil engineering and geosciences - cie4522-15, May 2017.

S van der Walt, J.L. Schönberger, J. Nunez-Iglesias, F. Boulogne, Warner J.D., N. Yager, E. Gouillart, and T. Yu. scikit-image: Image processing in Python. *PeerJ*, 2:e453, 2014. ISSN 2167-8359. doi: 10.7717/peerj.453. URL https://doi.org/10.7717/peerj.453.

S. Van Huffel and J. Vandewalle. *The total least squares problem: computational aspects and analysis*. Siam, 1991.

A. Vittorio, V. Rosolino, I. Teresa, C.M. Vittoria, P.G. Vincenzo, and D.M. Francesco. Automated sensing system for monitoring of road surface quality by mobile devices. *Procedia - Social and Behavioral Sciences*, 111:242 – 251, 2014. ISSN 1877-0428. doi: https://doi.org/10.1016/j.sbspro.2014.01.057. URL http://www.sciencedirect.com/science/article/pii/S1877042814000585.

G. Vosselman and H.G. Maas. *Airborne and Terrestrial Laser Scanning*. CRC Press, 2010. ISBN 978-1904445-87-6.

M. Wang, Birken R., and S. Shahini Shamsabadi. Implementation of a multi-modal mobile sensor system for surface and subsurface assessment of roadways. *Proc.SPIE*, 9436, 2015. doi: 10.1117/12.2084852. URL https://doi.org/10.1117/12.2084852.

Waylink. Waylink's digital highway data vehicle, 2015. Retrieved June 9, 2019, from http://www.waylink.com/dhdv.htm.

L. Yan and W. Xia. A modified three-dimensional gray-level co-occurrence matrix for image classification with digital surface model. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W13:133–138, 2019. doi: 10.5194/isprs-archives-XLII-2-W13-133-2019. URL https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2-W13/133/2019/.

S.Y. Yu, S. R. Sukumar, Andreas F. Koschan, D.L. Page, and M.A. Abidi. 3d reconstruction of road surfaces using an integrated multi-sensory approach. *Optics and lasers in engineering*, 45(7):808–818, 2007. doi: https://doi.org/10.1016/j.optlaseng.2006.12.007. URL http://www.sciencedirect.com/science/article/pii/S0143816606002077.

Zoller + Fröhlich GmbH. *Z+F PROFILER 9012*, 2018. Retrieved June 9, 2019, from https://www.zf-laser.com/fileadmin/editor/Datenblaetter/Z_F_PROFILER_9012_Datasheet_E_final_compr.pdf.