buildings

MDPI

Article

Feature-Centric Approach for Learning-Based Prediction of Pavement Marking Retroreflectivity from Mobile LiDAR Data

Dmitry Manasreh [1], Munir D. Nazzal [1,*] and Ala R. Abbas [2]

[1] Center for Smart, Sustainable & Resilient Infrastructure (CSSRI), Department of Civil &Architectural Engineering & Construction Management, University of Cincinnati, Cincinnati, OH 45221, USA; manasrda@ucmail.uc.edu

[2] Department of Civil Engineering, The University of Akron, Akron, OH 44325, USA; abbas@uakron.edu

* Correspondence: munir.nazzal@uc.edu; Tel.: +1-513-556-3736

Abstract: Given the crucial importance of pavement marking retroreflectivity in ensuring visibility for road safety, this research investigates the correlation between pavement marking reflectivity and LiDAR data. Empirical data were collected from eight road sections using both a handheld retroreflectometer and a mobile LiDAR. The approach proposed focuses on extracting important features from pavement marking regions of the LiDAR point cloud. A comprehensive feature extraction and feature selection process was employed. In addition, a well-rounded selection of learning algorithms was evaluated. A rigorous hold-out evaluation was incorporated, ensuring that the reported performance metrics were robustly generalizable. The best performing model was able to achieve an R^2 of 0.824 on unseen data. The findings of this study illuminate the potential for leveraging relatively inexpensive mobile LiDAR sensors in combination with machine learning techniques in conducting efficient pavement marking assessments, not only to detect completely degraded markings, but to accurately estimate retroreflective properties.

Keywords: road marking retroreflectivity; LiDAR; intensity; machine learning; AI; road condition assessment

check for updates

Citation: Manasreh, D.; Nazzal, M.D.; Abbas, A.R. Feature-Centric Approach for Learning-Based Prediction of Pavement Marking Retroreflectivity from Mobile LiDAR Data. *Buildings* **2024**, *14*, 62. https://doi.org/10.3390/buildings14010062

Academic Editor: Salvatore Antonio Biancardo

Received: 25 November 2023
Revised: 15 December 2023
Accepted: 18 December 2023
Published: 25 December 2023

1. Introduction

Pavement markings are a fundamental way of providing information to road users, vehicle operators, and advanced driver assistance systems (ADAS). They give useful information to help vehicle operators place their vehicles inside the traffic lane. Lane marker retroreflectivity refers to the capacity of marks to reflect light back to the light source [1]. This attribute is especially important at night when pavement markings are mainly visible because headlight light is reflected into drivers' eyes by retroreflectors implanted in road stripes [2]. Pavement markings' nighttime visibility is typically measured in terms of the coefficient of retroreflected luminance (R_L).

The characteristics of lane markings, acceptable evaluation methods, and minimum retroreflectivity requirements measured in millicandelas per square meter per lux (mcd/m^2/lx) are described in the Manual on Uniform Traffic Control Devices (MUTCD) [3] which sets the national standard for traffic control devices. The MUTCD minimum retroreflectivity requirement is 50 mcd/m^2/lx for roads where speeds are between 35 and 50 mph, and 100 mcd/m^2/lx on any roads with speeds equal to or higher than 55 mph. However, one exception is two-lane roads with only centerline markings. For this type of road, the minimum R_L requirements are 100, and 250 mcd/m^2/lux for ≥35 mph, and ≥55 mph roads respectively. A detailed description of the acceptable evaluation methods is provided in [4]. ASTM D7585 "Standard Practice for Evaluating Retroreflective Pavement Markings Using Portable Hand-Operated Instruments" [5], and ASTM E1710-18 "Standard Test Method for Measurement of Retroreflective Pavement Marking Materials with CEN-Prescribed

Geometry Using a Portable Retroreflectometer" [6] define standard techniques, including sampling standards for measured retroreflectivity.

Traditional methods of measuring retroreflectivity, whether handheld or mobile, have significant overheads in terms of cost and labor. Additionally, these methods are solely focused on retroreflectivity, and the collected data cannot be applied to other purposes. Light Detection and Ranging, often known as LiDAR, is a promising alternative that has the potential to evaluate retroreflectivity while also being useful for a wide range of other applications [7]. It is a remote sensing technology that makes use of light in the form of a pulsed laser to detect distances. In a LiDAR point cloud, each X, Y, and Z point is accompanied with a characteristic known as intensity, which acts as a measurement of the strength of the return signal and is provided by most LiDAR systems [8]. The word intensity in a LiDAR output refers to the amplitude of the return signal, which may be the analog electrical signal produced from the photodetector or the digital waveform. One of the fundamental advantages of LiDAR intensity is that it is connected to surface reflectance and other surface properties. This is because LiDAR intensity represents the strength of the laser beam reflected from the object surface back to the beam source, which means it is inherently representative of the same properties as retroreflectivity.

2. Related Studies

There are a variety of confounding factors to which intensity is connected other than the object reflectance, including data gathering geometry, the scanning environment, and sensor characteristics [9]. There is no consensus in the literature on a method to perform intensity correction. One of the reasons it is hard to find a single correction method is that different LiDAR manufacturing companies are integrating various correction methods that are typically not disclosed to the customer. For instance, in [10], researchers evaluated the correction of LiDAR intensity for range and incidence angle using a Velodyne VLP-32 (Ultra Puck) LiDAR. The researchers selected a well-known model from the literature (Equation (1)) and implemented the correction on 49,584 points in a distance range (2.3 m to 16.1 m) and angle range (30.5 to 82.7). The study reported no correlation between the intensity and any of the evaluated factors. The researchers concluded that raw intensity from the tested LiDAR could be used directly to establish the correlation with R_L.

$$I_c = I \times \frac{D^2}{D_{ref}^2} \times \frac{1}{cos\alpha} \tag{1}$$

where:

I_c: corrected intensity.
I: intensity.
D: distance between sensor and target.
D_{ref}: user-defined reference range.
α: angle between the beam and the target.

In the same study [10], a correlation was established between the retroreflectivity measured by a handheld device and the VLP 32 LiDAR intensity. The LiDAR was placed at approximately 45 degrees from the horizon and the positioning was performed using a global navigation satellite system (GNSS) corrected with real-time kinematic positioning (RTK). The RTK base station was placed no further than 15 km from the location of the vehicle. The established correlation used a single term exponential model and 64 data points to establish the relationship. The model was validated by its ability to detect areas with high marking degradation on new points.

In ref. [11], the study concluded that LiDAR pavement marking retroreflectivity evaluation is a feasible method to inventory and manage pavement marking conditions in transportation agencies. The researchers presented a framework that included: (1) data collection using a surveying-grade LiDAR system (Reigl VMZ 2000), (2) pavement marking point extraction from the point cloud using an improved version of the Road Marking Extractor (RoME) method originally developed by Jung et al. in [12], (3) correlation of LiDAR intensity to retroreflectivity as measured by a mobile retroreflectometer, and (4) tracking pavement marking deterioration over an 18-month period. It is noted that the researchers performed radiometric intensity normalization in accordance with the findings in [13], which was established based on the same LiDAR sensor.

The researchers in [14] established a correlation between the LiDAR intensity measurements of four LiDAR sensors (three Velodyne HDL-32Es and one Velodyne VLP16) and the retroreflectivity measurements obtained using a mobile retroreflectometer (Road Vista Laserlux G7 (LLG7)). The data were collected over a 70-mile span of rural roads at a speed of 50 mph. The study reported a positive linear relationship with r-squared values up to 0.63 for the center skip line, and up to 0.87 for the right edge lane. In addition, it was indicated that the rear-mounted LiDAR performed as well as, or even better than, other installment locations on the vehicle. Previous studies that studied the intensity retroreflectivity relationship for pavement markings found that the best relationship was established using the intensity values in the top 10% in the active window region using a single term power model, which was able to achieve 0.836 r-squared [15].

Several studies have considered the correlation between LiDAR intensity and pavement marking retroreflectivity. Most of the studies have documented a correlation between retroreflectivity and LiDAR intensity but have not tested if this correlation holds true for new unseen data. This lack of validation raises concerns about the reliability of using this correlation for future predictions beyond the original datasets analyzed. In addition, the relationship is usually established with the mean or median of the intensity values. This assumption may be overlooking other features that could be obtained from the LiDAR data such as measures of intensity dispersion or variability within the marking area. Furthermore, some of the studies used expensive surveying-grade LiDAR systems, which means that the relationships established in these studies might not be transferable to more widely used, relatively inexpensive LiDAR sensors.

In this study, data were collected from several road sections with various low to medium pavement marking conditions. The reference standard retroreflectivity was measured using a standard handheld retroreflectometer. LiDAR data were collected using a Velodyne VLP32C sensor. Over 600 measured stations were used to establish the relationship. A rigorous feature extraction and selection methodology was employed. In addition, validation was performed using five-fold cross-validation as the performance metric to ensure robustness and generalizability. Several machine learning algorithms were evaluated, including linear models such as ridge, lasso, and elastic net; gradient boosting implementations like XGBoost and lightGBM; as well as Random Forest, ExtraTrees, and k-Nearest Neighbors. The methodology proposed in this study presents a promising framework for leveraging machine learning techniques in conducting efficient pavement marking assessments using relatively inexpensive LiDAR sensors.

3. Methodology

3.1. Data Acquisition Tools

The LiDAR data collection system consisted of three main components, namely a LiDAR, a GNSS unit, and a computer for data acquisition and processing. The LiDAR used in this study was a Velodyne VLP-32C Ultra Puck. The LiDAR used a 903 nm wavelength and had 360° horizontal and 40° vertical fields of view (FOV), minimum vertical angular resolution of 0.33° distributed nonlinearly across the FOV, horizontal angular resolution ranging from 0.1° to 0.4°, 200 m range, and a maximum of 1.2 million points per second generated in the dual return mode. The LiDAR supports a frame rate of 5 to 20 Hz. The intensity measurement obtained by the LiDAR is calibrated by the manufacturer using a method that is not disclosed. Previous studies showed that the VLP32-C can be used to estimate retroreflectivity using no more than the manufacturer's intensity calibration [10]. The data acquisition and synchronization were conducted using a robot operating system (ROS) framework. The LiDAR data collection system is displayed in Figure 1.

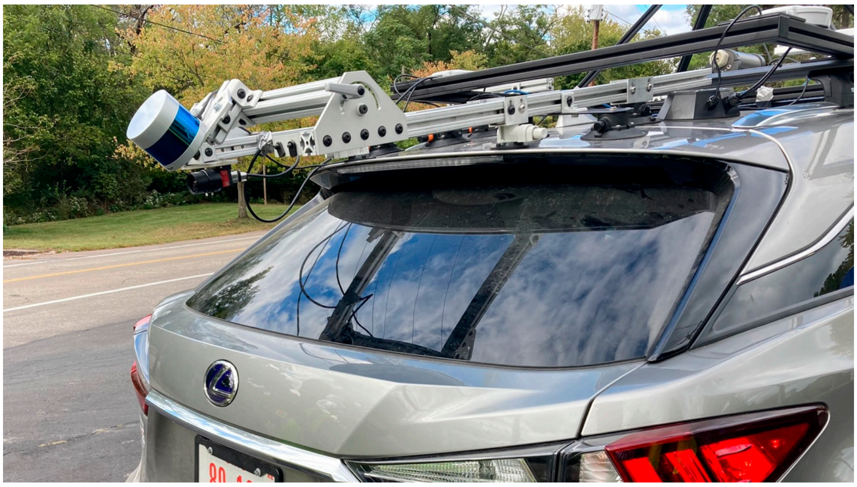


Figure 1. LiDAR data collection system.

The reference standard retroreflectivity of the pavement marking was measured using a handheld retroreflectometer. The retroreflectometer used was a Delta LTL-X Mark II which is compliant with most standards adopted by national transportation agencies for measuring pavement marking retroreflectivity [16]. The instrument has a reproducibility of ±5% and a repeatability of ±2%. The process of collecting measurements using the handheld retroreflectometer is depicted in Figure 2.

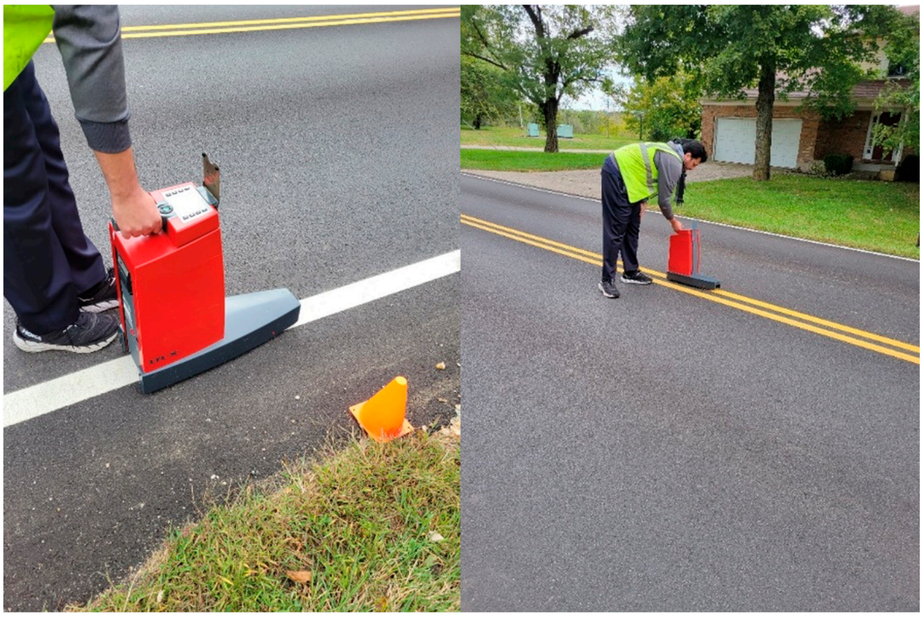


Figure 2. Manual measurement using a LTL-X handheld retroreflectometer.

3.2. Test Sections and Experiment Description

A total of eight road sections were selected around the Cincinnati, Ohio area. The map of sections locations is displayed in Figure 3. The sections were selected to represent variable marking conditions. Figure 4 shows different pavement marking conditions. All the images are from the tested sections. Control points were marked beforehand on the ground at approximately 20 ft intervals. The retroreflectivity was measured with the handheld device at each control point. For some sections, retroreflectivity measurements were obtained only at the edge lines while in other sections single or double lines of the centerline marking were included.

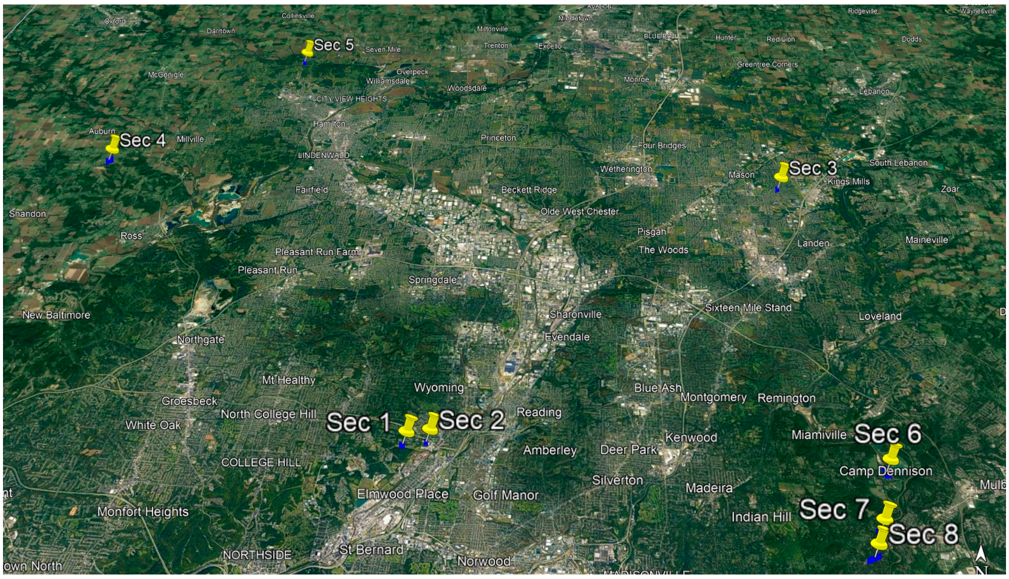


Figure 3. Map of tested road section locations.

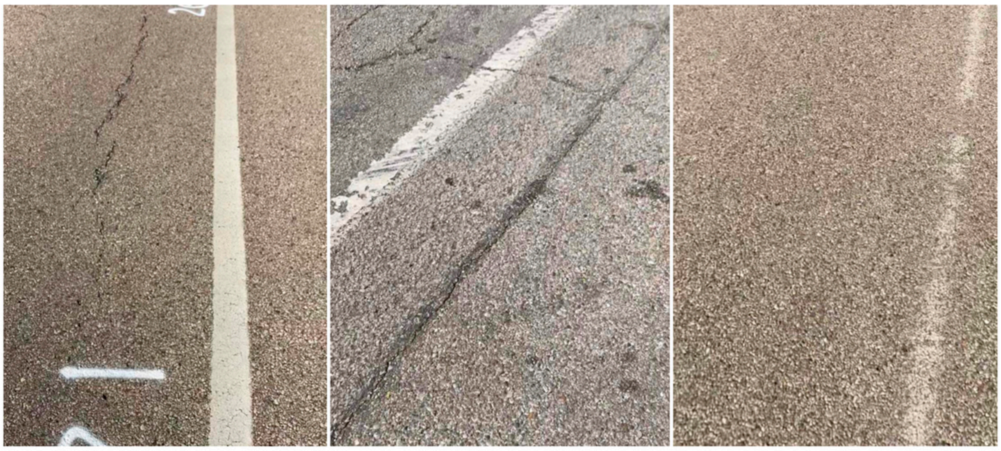


Figure 4. Variable pavement marking conditions.

3.3. Data Preparation

The LiDAR data were first adjusted to a local Cartesian coordinate system in which the x-axis was parallel to the moving direction, the y-axis was transverse to it, and the road surface was horizontal to the best extent possible. With the physical LiDAR configuration angles $\alpha,\ \beta,\ \gamma$ representing the rotation in degrees around the x, y, and z axes respectively, the rotation matrices are defined as follows:

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\alpha & -sin\alpha \\ 0 & sin\alpha & cos\alpha \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} cos\beta & 0 & sin\beta \\ 0 & 1 & 0 \\ -sin\beta & 0 & cos\beta \end{bmatrix}$$

$$R_z(\gamma) = \begin{bmatrix} cos\gamma & -sin\gamma & 0 \\ sin\gamma & cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The origin is taken as the physical center of the LiDAR; thus, no translation is necessary. The rotation is performed around the origin (0, 0, 0). The combined rotation matrix R is defined as:

$$R = R_z(\gamma)R_y(\beta)R_x(\alpha)$$

Given a point cloud P represented as:

$$P = \{p_1, p_2, \ldots, p_n\}$$

Each point p_i in the point cloud is represented as a vector (in addition to a corresponding intensity i_i):

$$p_i = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix}$$

The rotation matrix R is applied to each point in the point cloud P resulting in the rotated point cloud P':

$$p'_i = Rp_i$$

$$P' = \{p'_1, p'_2, \ldots, p'_n\}$$

All subsequent references to the point cloud P herein pertain to the rotated point cloud P', unless explicitly stated otherwise. An example of the top, cross-sectional, and intensity cross-section views of a single scan are shown in Figure 5.

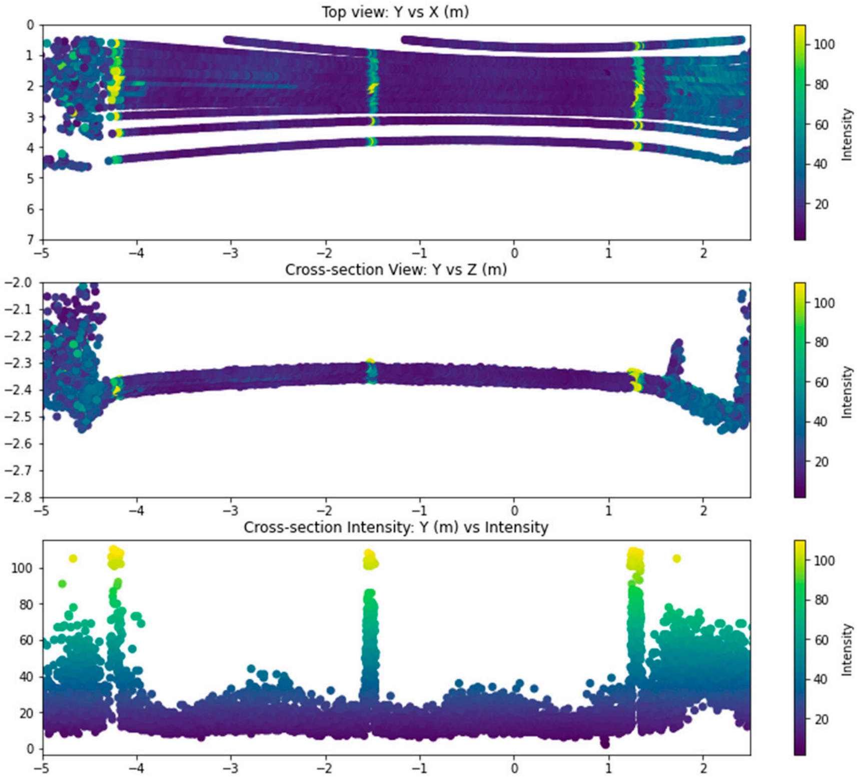


Figure 5. Top, cross-section, and intensity cross-section views of a single LiDAR scan.

To extract the lane marking data points from the LiDAR scans, a process was developed to simplify manual segmentation. Rasterized graphs of the top view and intensity cross-section were used to segment the lane marking. The image limits were independent from the range of the point cloud used to plot it, so that each pixel in the rasterized image was exactly matched to the dimensions in the original point cloud. Next, the rasterized images were manually annotated to segment the points belonging to the pavement marking at the retroreflectometer’s measuring window. The annotated bounding boxes were then translated back to actual dimensions. Finally, points belonging to the annotated areas were extracted from the full point cloud. A rasterized intensity cross-section graph is shown in Figure 6.

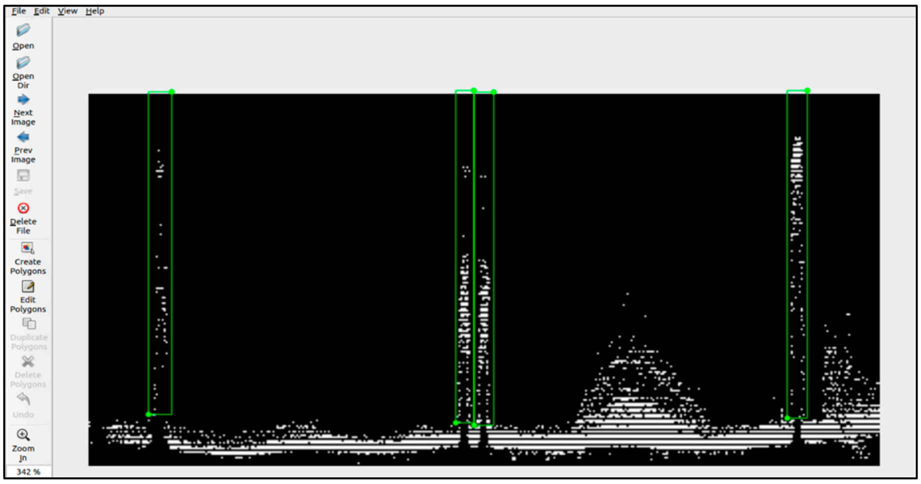


Figure 6. Annotation of rasterized cross-sectional intensity view.

After annotation and segmentation of the marking points, the output of this process at each station is an array of shape (n, 4, where n is the number of point cloud points at the measured station and 4 represents the x, y, z, and intensity readings. Figure 7 summarizes the proposed data preparation process.

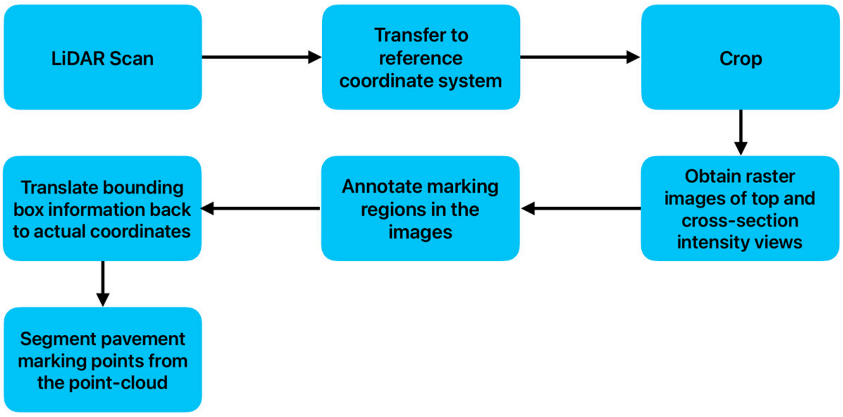

LiDAR Scan
Transfer to reference coordinate system
Crop
Obtain raster images of top and cross-section intensity views
Annotate marking regions in the images
Translate bounding box information back to actual coordinates
Segment pavement marking points from the point-cloud

Figure 7. Data preparation flowchart.

3.4. Feature Extraction

After selecting points for each pavement marking location using the procedure discussed previously, the dataset was structured in such a way that each measured retroreflectivity value corresponded to a collection of point cloud points. The number of point cloud points varied between locations, making it challenging to fit a regression model without consistent input. To address this issue, features were extracted from the point cloud data. This feature extraction process transformed the dataset into a standardized format, where each pavement marking location was represented by a fixed set of features, regardless of the original number of points in the point cloud. These extracted features captured the relevant information from the point cloud data, providing a consistent shape of input for the regression model. The shape of the dataset is described in Figure 8, where, at location (N): S_N is a subset of point cloud points of shape $\left(n_{\,N} \times 4\right)$, $(R_L)_N$ is the measured retroreflectivity value, and f_N is the vector of extracted features of constant shape $(N_{features})$.

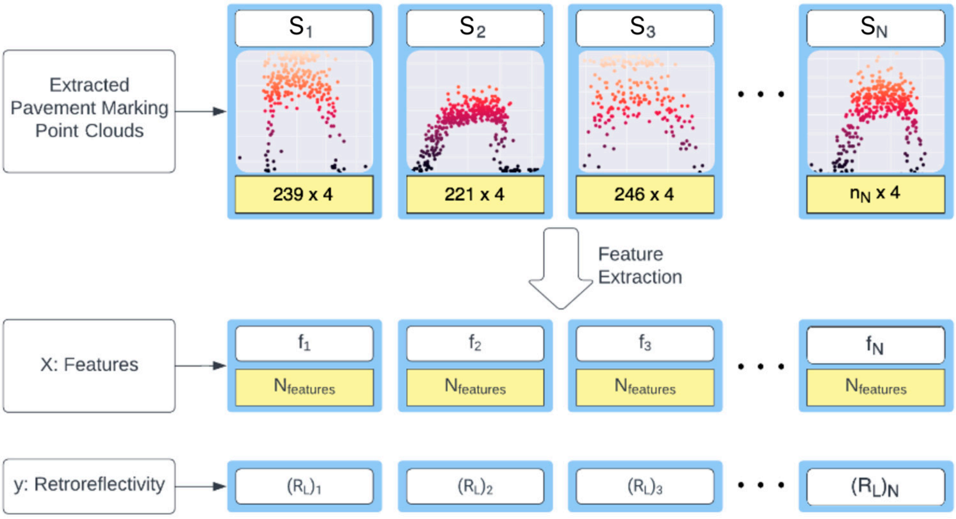


Figure 8. Feature extraction and dataset shape description.

For the task of extracting and evaluating a wide range of features, the Python package (tsfresh) proved to be a valuable tool. tsfresh is designed to automatically compute numerous features from time series data [17]. In this study, the data could be treated as time series data, such that the lateral axis (y axis) corresponded to the time axis. By using tsfresh, we could leverage its functionality to extract a multitude of features for each channel of the input data. These features included basic statistics (such as minimum, maximum, mean, various percentiles, and skewness), complexity features (such as count above and below mean), linear trend features, and entropy features, among many others. In general, the feature names were encoded such that the first letter represented the variable from which the feature was extracted, followed by double underscores and the feature family, followed by any parameters used to define the feature. For example, $i\_\_quantile\_\_q\_0.7$ was applied to i which was the intensity, the feature family was the $quantile$, and the particular quantile this feature represented was the 0.7 $quantile$. Following is a summary of the most important extracted features:

Basic features:

These include the mean, median, mode, standard deviation, root mean square, absolute energy, maximum, minimum, range, inter-quartile range, quantiles from 0.1 to 0.9, skewness, kurtosis, sum of recurring values, and many other basic and statistical features describing the central tendency, the variation, and the distribution shape of the data. The full list of features extracted is described in [18].

Change quantiles features:

This family of features involves fixing a window with top and bottom limits based on quantiles ql (the lower quantile) and qh (the higher quantile). Only values inside this window are considered and consecutive changes of the values are determined. The differences are taken either raw or as absolute values. Finally, the differences are aggregated using an aggregate function f_agg such as mean, median, variance, etc.

***C*3 statistics:**

This feature is a measure of the nonlinearity of the data values across the direction of y (lateral distance). The feature is implemented in the tsfresh library and was proposed in [19]. It is calculated as [20]:

$$\frac{1}{n-2lag}\sum_{j=0}^{n-2lag} i_{j+2lag}^{2} \cdot i_{j+lag} \cdot i_{j} \tag{2}$$

where:

n: is the number of data points.
i: is the intensity value.
lag: is the lag.

Aggregated linear trend features:

This family of features involves calculating the linear least-squares regression of values aggregated over defined chunks. First, the series was divided into chunks of defined length $chunk\_len$. Next, an aggregation function f_agg was applied on each chunk to summarize its values. The aggregation function could be max, min, mean, or median. Finally, a linear regression was applied on the aggregated values. In this regression, the aggregated values were the dependent variable, and the chunk number was the independent variable. The attribute returned ($attr$) could be "p-value", "r-value", "intercept", "slope", or "stderr".

3.5. Feature Selection

The feature extraction process from the previous section resulted in more than 3000 features extracted. Dealing with such high dimensional data required rigorous observation to exclude useless data. The feature selection process implemented in this study consisted of multiple steps. The feature selection process is discussed in detail in the following subsection.

3.5.1. Filter Methods

The first step was simply to analyze the variance of the feature and observe that it did not stay unchanged across the different samples. The second step of the feature selection process utilized a set of three filter methods. The methods implemented included Pearson's correlation coefficient, Spearman's rank correlation coefficient, and the maximal information coefficient (MIC). These approaches were chosen for their ability to evaluate features based on their relationship with the output variable, independent of any specific model's influence. Each method, with its unique properties, provided valuable insights into the presence of a relationship for each feature with the measured retroreflectivity. Thresholds were set for each of the filter methods used, and the final set of features was selected based on passing the thresholds of the filter methods. This step drastically reduced the number of features to be considered in the following steps. In general, the number of features was reduced from more than 3000 features to sub 100 related features.

Variance thresholding:

The variance $Var(X)$ of a feature X is calculated as:

$$Var(X) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})^2 \tag{3}$$

where:
N: is the number of samples.
x_i: is the value of the feature for the ith sample.
$\overline{x}$: is the mean value of the feature across all samples.

The variance threshold method then checks the variance of each feature against a specified threshold such that, if $Var(X) < Threshold$, then remove feature X.

Pearson's correlation coefficient (*r*):

Pearson's correlation coefficient is a metric measuring the linear association between two continuous variables. Its value ranges between -1 and 1, with values near the extremes indicating strong negative or positive linear relationships. When selecting features, a significant Pearson's correlation with the target variable can suggest the importance of the feature. However, it is imperative to note that features with high inter-correlations can introduce redundancy, suggesting the potential removal of overlapping features. Nonetheless, in this study, this will be addressed using a different method in later steps of feature selection. The Pearson's correlation coefficient (r) is calculated using the formula:

$$r = \frac{\sum_{i=1}^{n} (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \overline{x})^2 \ \sum_{i=1}^{n} (y_i - \overline{y})^2}} \tag{4}$$

where:

n: is the number of data points.

x_i and y_i: are the individual data points of the feature x and the variable y.

$\overline{x}$ and $\overline{y}$: the means of all the data points of x and y respectively.

Spearman's rank correlation coefficient (ρ):

Spearman's rank correlation coefficient is a non-parametric measure used to assess the strength and direction of the monotonic relationship between two variables. The value of Spearman's rank coefficient can range from -1 to 1, with values close to the extremes indicating a strong negative or positive monotonic relationship. The ranking process makes Spearman's method more robust to non-linear relationships and outliers compared to methods that rely on actual data values, such as Pearson's correlation.

Spearman's rank correlation coefficient (ρ) is given by:

$$\rho = 1 - \frac{6\sum d^2}{n(n^2 - 1)} \tag{5}$$

where:

d: is the difference between the ranks of the two variables for each data point. (The ranks refer to the ordering of data points when sorted in ascending or descending order.)

n: is the number of data points.

Maximal information coefficient (MIC):

The maximal information coefficient (MIC) is a modern statistical measure used to identify and quantify associations between two variables. The basis of MIC lies in the concept of mutual information. Mutual information quantifies the amount of information one can acquire about one variable by observing another. For MIC, the objective becomes finding an optimal grid partitioning for the paired variables that pushes this mutual information to its peak. Once the mutual information is computed, it undergoes a normalization process which aims to confine the MIC values within a range of [0, 1]. A MIC value close to 0 would suggest a negligible association between the variables in question. In contrast, a value veering towards 1 would be indicative of a pronounced association.

3.5.2. Exhaustive Search

After the preliminary round of feature selection, a refined subset of features remained. The features that demonstrated their significance in the initial selection phase were then subjected to a rigorous evaluation to observe their individual and combined performance in a model. First, each feature was employed independently to train the model, after which the model's performance was gauged. This meticulous individual analysis aimed to pinpoint the feature that, on its own, offered the most substantial contribution to the model's performance. After the singular evaluations, Exhaustive Search was employed to explore all possible pairings of the shortlisted features. In practical terms, for a dataset with n remaining features post the initial selection, Exhaustive Search would evaluate $\binom{n}{2}$ potential feature pairs. Each of these pairs was used to train the model, and its performance was recorded.

3.5.3. Wrapper Feature Selection: Sequential Backward Selection

In general, wrapper methods are a type of feature selection technique where different subsets of features are used to train a model. The performance of the models trained on the different subsets is then used to gauge the feature importance and select the features that result in the best performance metrics. In contrast to filter methods where the feature usefulness is determined by intrinsic properties of the data, wrapper method depend on empirical performance of the model trained on the particular subset of features. In sequential backward selection, the evaluation is initiated by including the full set of features then sequentially removing the least important features. The decision to remove a feature is based on the impact of its removal on the model performance. The process is repeated until a desired number of features is reached, or until the removal of further features significantly degrades the model performance.

3.6. Evaluated Models

In the comprehensive evaluation of potential models that correlate LiDAR data to retroreflectivity, a diverse set of regression models was explored. Evaluating a diverse suite of algorithms ensured that the analysis was both comprehensive and robust, catering to the myriad possible structures and relationships within the data. Following is a discussion of the evaluated models.

Linear Regression is one of the foundational algorithms in regression analysis. It assumes a linear relationship between the independent and dependent variables, aiming to find the best-fitting line that minimizes the sum of squared errors. While it offers simplicity and interpretability, its assumption of linearity can be a limitation, especially when the true relationship between variables is more complex.

Lasso Regression, which stands for Least Absolute Shrinkage and Selection Operator, introduces L1 regularization to the traditional linear regression, adding a penalty equivalent to the absolute value of the magnitude of coefficients. This often results in some feature coefficients being exactly zero, effectively leading to feature selection. Lasso can be particularly useful when it is suspected that many features are redundant or irrelevant.

Ridge Regression is another variant of linear regression that employs L2 regularization. Instead of eliminating certain features as Lasso does, Ridge tends to shrink the coefficients of less important features closer to zero, balancing out the contribution of all features. It is especially beneficial in situations where features are highly correlated, known as multicollinearity.

ElasticNet Regression offers a middle ground between Lasso and Ridge by combining both L1 and L2 regularizations. This hybrid approach ensures that ElasticNet inherits the strengths of both methods, making it versatile in handling various data structures.

Random Forest is an ensemble technique that builds multiple decision trees and aggregates their results. By leveraging the power of multiple trees, it mitigates the risk of overfitting, to which a single decision tree might be prone. Random Forest starts by creating multiple sets of data from the original dataset using a technique called bootstrap sampling. This means that for a dataset of size N, several samples each of size N are randomly selected with replacement, such that a sample can be picked more than once. Let us say that B bootstrapped samples are created. For each of these bootstrap samples, a decision tree is constructed. However, instead of considering all features at each split, only a random subset of the features is considered. The final prediction of the regression is then the average of the predictions from all the trees.

Extra Trees, or Extremely Randomized Trees, is an ensemble method similar to Random Forest. The primary distinction lies in how the trees are constructed: while splits in Random Forest are based on the most discriminative thresholds, Extra Trees introduces more randomness in selecting features and thresholds, often leading to increased diversity among individual trees.

Gradient Boosting is another ensemble technique, but instead of building trees in parallel like Random Forest, it constructs them sequentially. Each tree in the sequence tries to correct the errors made by the previous ones. The algorithm starts with a simple model, often a constant value representing the mean of the target variable. The primary objective then becomes the minimization of residuals or the discrepancies between the predicted and actual values. As the algorithm progresses, each tree is specifically trained on these residuals from the preceding model. This strategy ensures that every new tree in the sequence addresses and rectifies the shortcomings of the previous ones.

XGBoost stands for eXtreme Gradient Boosting. It is an optimized and more efficient implementation of the gradient boosting algorithm, known for its speed and performance. XGBoost introduces regularization to the boosting process, reducing the risk of overfitting and often leading to better generalization on unseen data.

LightGBM or Light Gradient Boosting Machine is another gradient boosting framework designed for speed and efficiency. It differs from other boosting algorithms in its approach to tree growth, opting for a leaf-wise strategy rather than the traditional depth-wise approach.

kNN (k-Nearest Neighbors) for Regression is a non-parametric algorithm that predicts output based on the average of the k nearest training examples in the feature space. It is especially adept at capturing localized patterns within the data.

3.7. Performance Metrics

Two fundamental performance metrics in the context of regression modeling are the coefficient of determination, R^2, and the mean squared error (MSE). The coefficient of determination, denoted as R^2, quantifies the proportion of the variance in the dependent variable that is predictable from the independent variables. It is mathematically defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2} \tag{6}$$

Here, y_i is the actual value, $\hat{y}_i$ is the predicted value, and $\overline{y}$ is the mean of the observed data.

In this study, the coefficient of determination R^2 was predominantly employed as the performance metric. The primary rationale behind this choice was to maintain consistency with the conventions established in related literature.

Cross-Validation

One of the foremost challenges in machine learning is overfitting, which occurs when a model, while aiming to achieve the best possible performance on training data, becomes overly complex, capturing not just underlying patterns but also the noise or random fluctuations present. The result is a model that performs exceptionally on the training set but falters when exposed to unseen data. Cross-validation emerges as a safeguard against this. Furthermore, cross-validation provides an estimate of a model's generalization performance. If a model performs well across different cross-validation folds, it is more likely to perform well on unseen data.

In k-fold cross validation, the dataset is randomly partitioned into k equal-sized subsamples or folds. Of these k folds, a single fold is retained as the validation set, and the remaining $k-1$ folds are used as the training set. The model is then trained on the $k-1$ training folds and validated on the held-out fold. This process is repeated k times. At the end of this process, k different models are obtained and k performance estimates are calculated. The number of folds k can be selected based on multiple factors such as the size of the dataset available. In this study, the number of folds k was selected to be 5. The overall performance was then calculated as the mean of these k performance metrics. The process is illustrated in Figure 9. Let us say that R^2 is the performance metric used; the overall R^2 is calculated as:

$$R^2 = \frac{1}{k}\sum_{i=1}^{k} R_i^2 \tag{7}$$

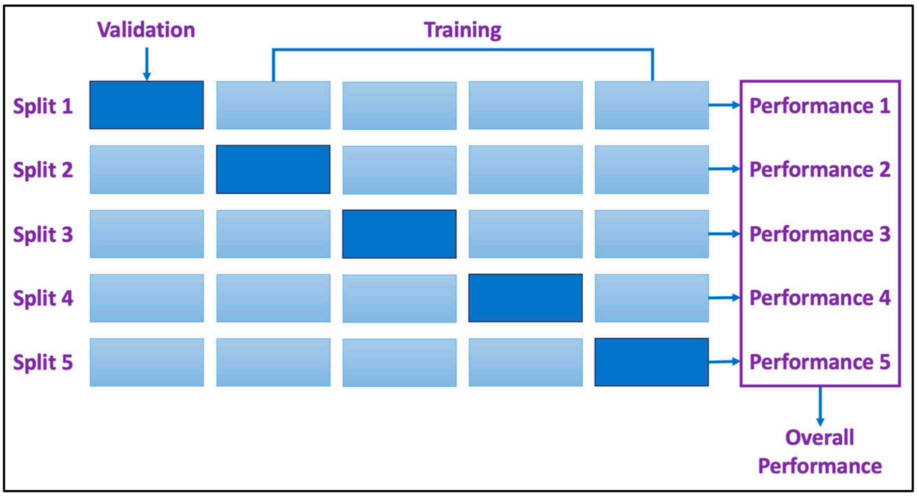


Figure 9. *k*-Fold cross-validation.

4. Results

First, it is important to observe the distribution of the measured retroreflectivity represented in the dataset collected. The results show that the measured retroreflectivity of the dataset was well distributed around values from 20 to 400 (mcd/m^2/lx). While there were some data with values higher than 400, these were less frequent. Nonetheless, since the recommended thresholds for satisfactory retroreflectivity according to most transportation agencies lay much lower, at around 50, 100, or 250 mcd/m^2/lx, increasing the number of points above 400 was not expected to have a significant impact on the usefulness of the model for degraded evaluation. The distribution of retroreflectivity values in the dataset is displayed in Figure 10.

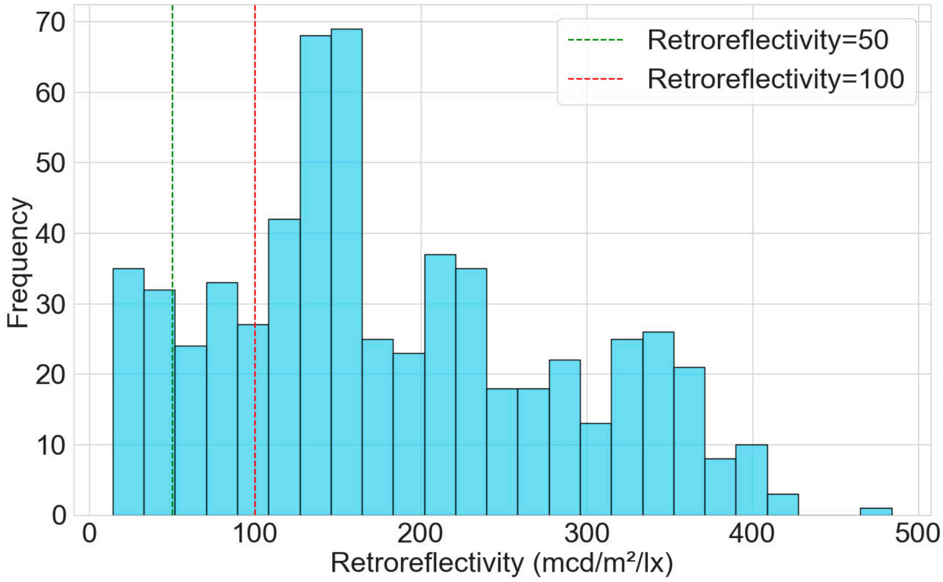


Figure 10. Distribution of measured retroreflectivity values in the dataset.

As discussed in Sections 3.3 and 3.4 the point cloud points belonging to the evaluated pavement markings were used to generate a vast set of features which were initially filtered on variance as described in (Variance thresholding). The total number of features generated exceeded 3500 features. After initial filtering, by removing features with many unreasonable or undefined values or stagnant features through simple variance thresholding, 1367 features remained. Next, exploring the data began with calculating statistical association metrics Pearson's r, Spearman's ρ, and maximal information coefficient (MIC).

The results of these parameters are displayed in Figure 11. The criteria selected to filter features for further exploration were set such that any feature that achieved an absolute value of 0.5 on any of the coefficients was selected for further evaluation. As observed in Figure 11, only a small subset of 82 features passed the threshold levels.

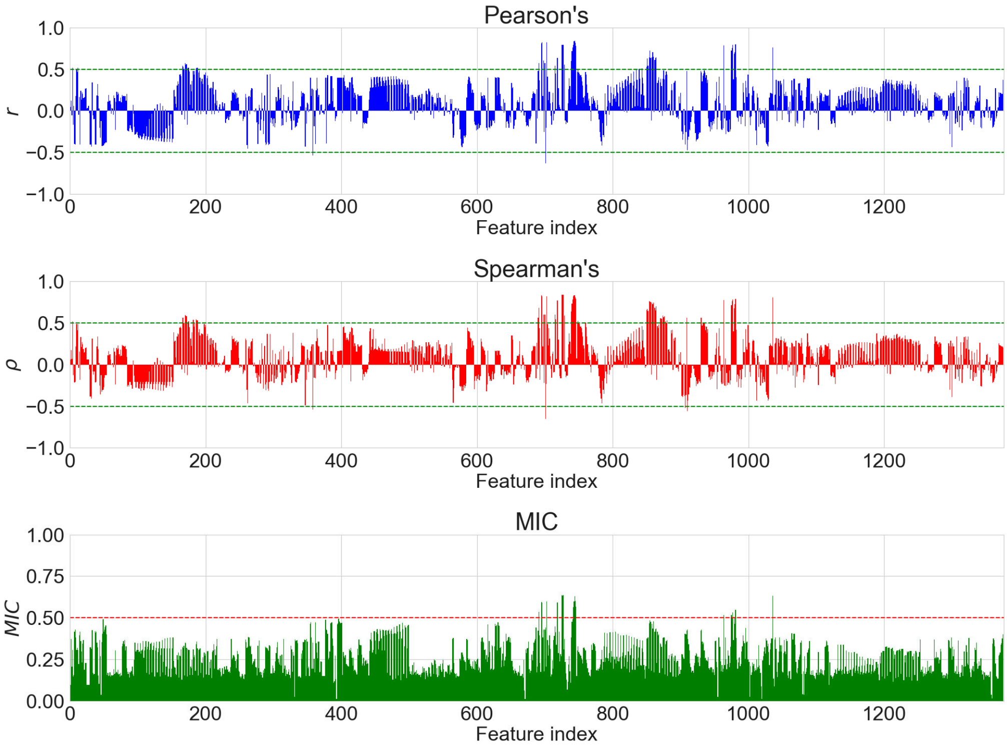


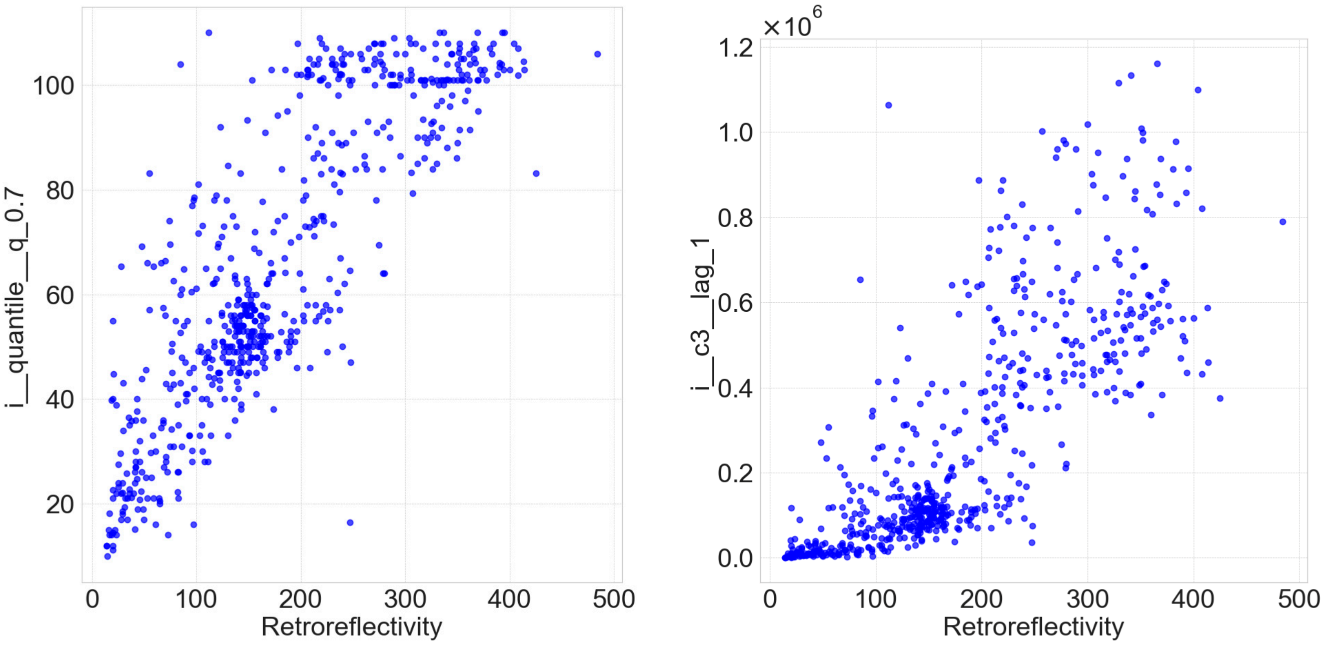
Figure 11. Statistical association metrics results.

The features with the highest score for each of the coefficients are displayed in Table 1. For Pearson's r the top feature with the highest score was $i\_\_quantile\_\_q\_0.7$ with a value of 0.845. This feature was the 70th percentile of the intensity values of the set of points in the marking. This result suggests a high linear correlation of the feature with measured retroreflectivity values. Figure 12a displays this relationship. On the other hand, the feature with the highest Spearman's ρ score and the highest MIC value was $i\_\_c3\_\_lag\_1$ with scores of 0.837 and 0.635, respectively. This feature clearly showed a monotonic nonlinear correlation with retroreflectivity, as displayed in Figure 12b. The feature was a measure of the nonlinearity of the intensity values across the direction of the y-axis (lateral distance).

All the different models were proceeded by standardization. The evaluation metrics reported were the result of a five-fold cross-validation for each model. In particular, the mean R^2 of the five-folds was used for selection. The evaluation process started with the full set of features after filter-based selection (a total of 82 features). Exhaustive Search was used to evaluate all single-feature models, as well as all combinations of two features. Sequential backward selection was used to find the best combinations of 5, 10, 20, 30, and 40 features. The results of the model evaluation on the full set of features are shown in Table 2. The best performing model on the full set of features was the ExtraTrees model, which was able to achieve a R^2 value of 0.823. The model had an average inference time of 18 ms on the test set, which was on the slow side compared to the methods evaluated. Nonetheless, its speed was more than sufficient for potential real-time processing. The LightGBM model achieved 0.809 R^2 with a scoring time that was almost 10 times faster. However, it was noted that the estimation of all 82 features required a lot of processing power and time. In addition, in some cases lower feature dimensionality could increase the performance of some models.

Table 1. Top feature of each metric.

Metric	Top Feature Name	Value
Pearson's (r)	*i__quantile__q*_0.7	0.845
Spearman's (ρ)	*i__c3__lag*_1	0.837
MIC	*i__c3__lag*_1	0.635



(**a**) (**b**)

Figure 12. Features with highest association vs. measured retroreflectivity (mcd/m^2/lx) (**a**) based on Pearson's (r) (**b**) based on Spearman's (ρ) and MIC.

Table 2. Model performance (all filtered features).

Model	Score Time (s)	Test MSE	Test R^2
ExtraTrees	0.018	1865.366	0.823
LightGBM	0.002	1919.982	0.809
RandomForest	0.017	1931.275	0.808
GradientBoosting	0.001	1951.247	0.807
XGBoost	0.002	1961.802	0.805
KNeighbors	0.035	2131.877	0.793
Lasso	0.002	2199.763	0.784
Ridge	0.001	2226.763	0.779
Elasticnet	0.001	2319.176	0.773
Linear	0.001	2376.140	0.764

Table 3 displays the results of the top single-feature performance for each model. The best single-feature model performance was obtained using the *i__quantile__q*_0.7 and a Ridge model, with Linear and Lasso models following very closely. It was noted that the same feature also had the highest Pearson's r. The best single-feature model had an average scoring time less than 1 ms.

Table 3. Top single-feature performance of each model.

Model	Score Time (s)	Test MSE	Test R^2	Feature
Ridge	0.000	3029.194	0.703	*i__quantile__q_0.7*
Linear	0.000	3029.060	0.703	*i__quantile__q_0.7*
Lasso	0.000	3030.795	0.703	*i__quantile__q_0.7*
LightGBM	0.001	3079.462	0.698	*i__c3__lag_1*
GradientBoosting	0.001	3282.211	0.678	*i__c3__lag_1*
KNeighbors	0.014	3296.097	0.676	*i__c3__lag_1*
RandomForest	0.019	3721.588	0.635	*i__quantile__q_0.7*
Elasticnet	0.000	3901.383	0.620	*i__quantile__q_0.7*
XGBoost	0.001	3953.639	0.613	*i__c3__lag_3*
ExtraTrees	0.013	3960.911	0.609	*i__quantile__q_0.7*

The results of Exhaustive Search and sequential backward feature selection are displayed in Figure 13. The R^2 results are displayed versus the number of features selected. The highest R^2 value of 0.824 was achieved by the model using 40 features. For the models using a single or two features, a linear model was the best performing model, while for all others, an ExtraTrees model was able to achieve the best results. The figure shows a clear increase in the performance when increasing the number of features utilized. This difference is most significant in the beginning when moving from one to five features. Adding more than 40 features did not improve the model's performance. In a real-life scenario, selecting the number of features to be utilized depends on the particular application. For example, if the goal is to develop the best prediction model with no limitations on time or computational power, more features can be used. On the other hand, if the goal is to develop a real-time analysis pipeline for retroreflectivity prediction from LiDAR data, utilizing the five-feature model would give a sufficient prediction performance with much lower computational cost. Figure 14 displays the predicted results using the model trained with one (a), five (b), 40 (c), and 82 (d) features versus the actual retroreflectivity values.

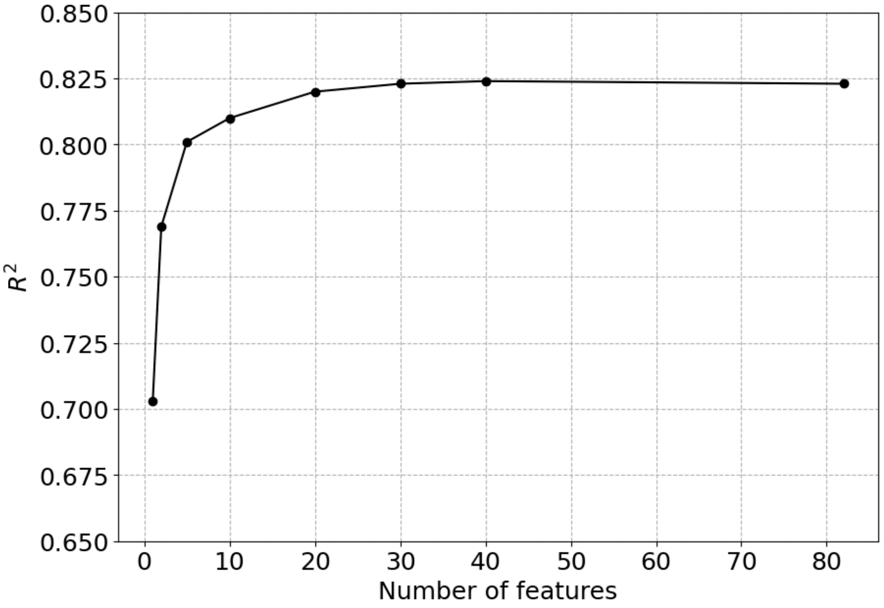
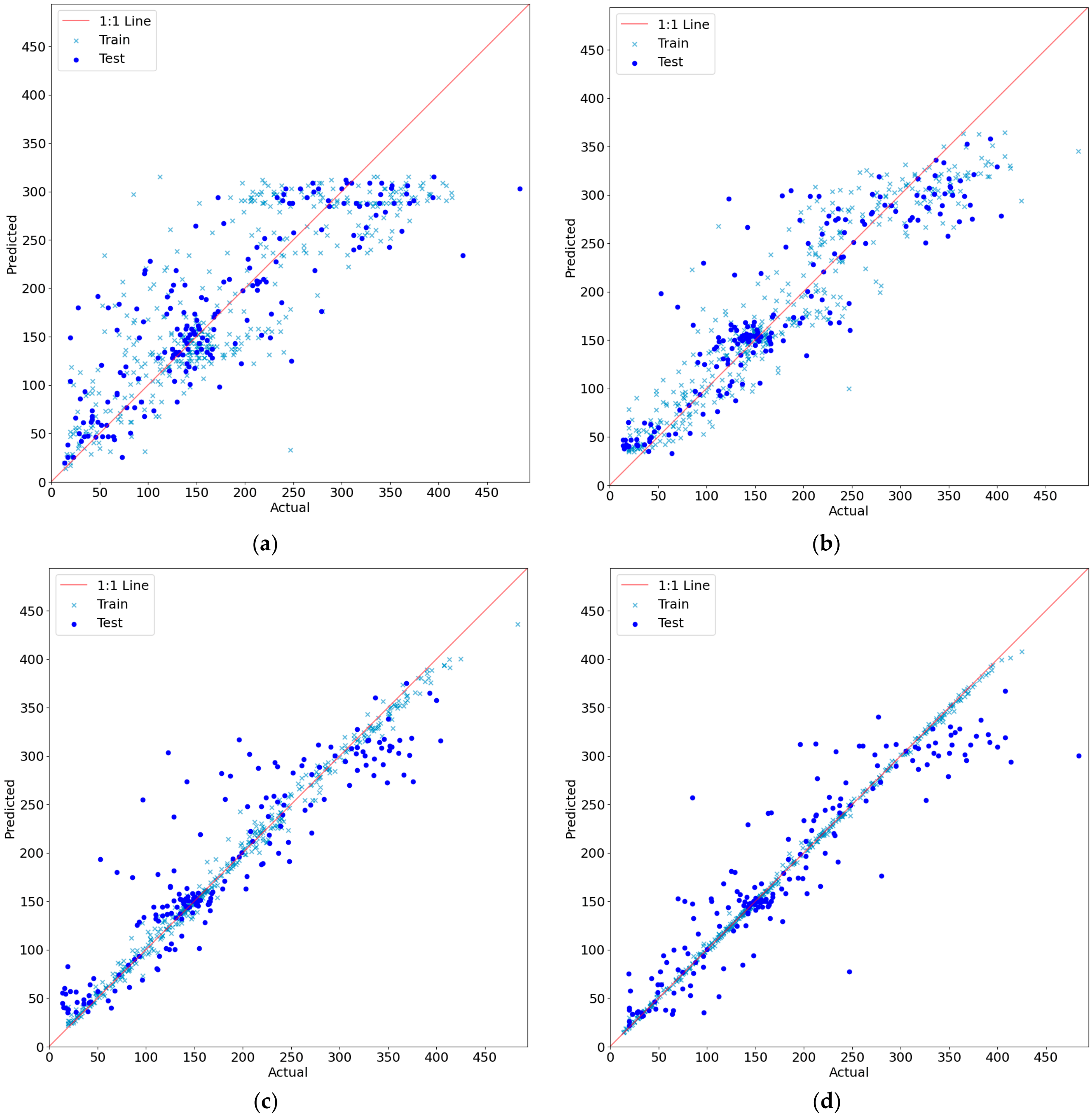


Figure 13. Feature selection results (R^2 vs. number of features).



(**a**)

(**b**)

(**c**)

(**d**)

Figure 14. Predicted vs. actual retroreflectivity (mcd/m^2/lx) from the best models trained with (**a**) a single feature (R^2 = 0.703); (**b**) five features (R^2 = 0.801); (**c**) 40 features (R^2 = 0.824); (**d**) full set (82 features) (R^2 = 0.823).

5. Discussion and Conclusions

Retroreflectivity stands as a pivotal property of road markings, playing a crucial role in maintaining visibility for drivers, especially during nighttime. This study was aimed at developing a framework for predicting pavement marking retroreflectivity using a mobile LiDAR sensor. Data were collected from several road sections with various low to medium pavement marking conditions. Data were collected using a standard handheld retroreflectometer and a relatively low-cost rotating-sensor mobile LiDAR. Over 600 measured stations from eight road sections were used to establish the relationship. The methodology proposed in this study included comprehensive feature extraction. The features described the distribution of LiDAR data and their intensity within the marking regions. Next, a multi-level rigorous feature selection methodology was utilized, and validation was performed using five-fold cross-validation as the performance metric to ensure robustness and generalizability. Several machine learning algorithms were evaluated, including linear models such as Ridge, Lasso, and ElasticNet; gradient boosting implementations like XGBoost and lightGBM; as well as Random Forest, ExtraTrees, and k-Nearest Neighbors.

The dataset obtained in this study had 615 measured retroreflectivity stations and was well distributed in the range from 20 to 400 mcd/m^2/lx. The dataset was focused on this range because typically the specifications use 50, 100, or 250 mcd/m^2/lx as the minimum criteria for different road categories. The feature with the highest linear correlation (Pearson's (r) = 0.845) to retroreflectivity was found to be the 70th quantile of the intensity values, while the $c3$ statistic feature had the best monotonic relationship (Spearman's (ρ) = 0.837). The best model performance was obtained by an ExtraTrees model trained using 40 features. The model achieved an average R^2 of 0.824. It was noted that this performance was of hold-out data never seen by the model. Thus, it should not be compared with the R^2 values reported in the literature describing the correlation of intensity–retroreflectivity values. An acceptable performance was obtained using the ExtraTrees model trained with five features, which achieved 0.801 R^2 while requiring much less computational cost (fewer features to extract and a smaller model).

The results of this study show the potential of accurate and generalizable prediction of road marking retroreflectivity using machine learning techniques and a relatively low-cost LiDAR. In future research, our focus will pivot towards evaluating the effect of different factors on the methodology and outcome models developed in this study. The factors include the effect of road environment such as wet road conditions, sensor factors such as comparing different LiDAR sensors, the effect of intensity normalization and calibration on the correlation of LiDAR data with retroreflectivity, and data collection factors such as a vehicle's lateral distance from the evaluated marking, vehicle speed, and data collection direction.

Author Contributions: Conceptualization, D.M. and M.D.N.; methodology, D.M. and M.D.N.; software, D.M.; validation, D.M. and M.D.N.; formal analysis, D.M. and M.D.N.; investigation, D.M. and M.D.N.; resources, M.D.N. and A.R.A.; data curation, D.M.; writing—original draft preparation, D.M.; writing—review and editing, M.D.N. and A.R.A.; visualization, D.M.; supervision, M.D.N.; project administration, M.D.N.; funding acquisition, M.D.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Karwa, V.; Donnell, E.T. Predicting pavement marking retroreflectivity using artificial neural networks: Exploratory analysis. *J. Transp. Eng.* **2011**, *137*, 91–103. [CrossRef]
2. Austin, R.L.; Schultz, R.J. *Guide to Retroreflection Safety Principles and Retroreflective Measurements*; Gamma Scientific: San Diego, CA, USA, 2020.
3. The Manual on Uniform Traffic Control Devices (MUTCD) 11th Edition. US Department of Transportation, Federal Highway Administration. 2023. Available online: https://mutcd.fhwa.dot.gov/kno_11th_Edition.htm (accessed on 14 December 2023).
4. Carlson, P.J.; Schertz, G.; Satterfield, C.; Falk, K.W.; Taylor, T. *Methods for Maintaining Pavement Marking Retroreflectivity*; (No. FHWA-SA-14-017); United States Federal Highway Administration. Office of Safety: Washington, DC, USA, 2014. Available online: https://rosap.ntl.bts.gov/view/dot/49516 (accessed on 15 June 2023).
5. *ASTM D7585/D7585M-10(2015)*; Standard Practice for Evaluating Retroreflective. ASTM International: West Conshohocken, PA, USA, 2005. Available online: https://www.astm.org/d7585_d7585m-10r22.html (accessed on 15 June 2023).
6. *ASTM E1710-18*; Pavement Markings Using Portable Hand-Operated Instruments. ASTM International: West Conshohocken, PA, USA, 2018. Available online: https://www.astm.org/e1710-18.html (accessed on 15 June 2023).
7. Manasreh, D.; Nazzal, M.D.; Talha, S.A.; Khanapuri, E.; Sharma, R.; Kim, D. Application of Autonomous Vehicles for Automated Roadside Safety Assessment. *Transp. Res. Rec.* **2022**, *2676*, 255–266. [CrossRef]

8. Che, E.; Olsen, M.J.; Parrish, C.E.; Jung, J. Pavement marking retroreflectivity estimation and evaluation using mobile LiDAR data. *Photogramm. Eng. Remote Sens.* **2019**, *85*, 573–583. [CrossRef]
9. Kashani, A.G.; Olsen, M.J.; Parrish, C.E.; Wilson, N. A review of LiDAR radiometric processing: From ad hoc intensity correction to rigorous radiometric calibration. *Sensors* **2015**, *15*, 28099–28128. [CrossRef] [PubMed]
10. Soilán, M.; González-Aguilera, D.; del-Campo-Sánchez, A.; Hernández-López, D.; Del Pozo, S. Road marking degradation analysis using 3D point cloud data acquired with a low-cost Mobile Mapping System. *Autom. Constr.* **2022**, *141*, 104446. [CrossRef]
11. Ai, C.; Hennessy, E. *A Pavement Marking Inventory and Retroreflectivity Condition Assessment Method Using Mobile LiDAR*; No. 22-030; Massachusetts Department of Transportation, Office of Transportation Planning: Boston, MA, USA, 2022. Available online: https://rosap.ntl.bts.gov/view/dot/63369 (accessed on 15 June 2023).
12. Jung, J.; Che, E.; Olsen, M.J.; Parrish, C. Efficient and robust lane marking extraction from mobile lidar point clouds. *ISPRS J. Photogramm. Remote Sens.* **2019**, *147*, 1–18. [CrossRef]
13. Ai, C.; Tsai, Y.J. An automated Sign Retroreflectivity Condition Evaluation Methodology Using Mobile LIDAR and Computer Vision. *Transp. Res. Part C Emerg. Technol.* **2016**, *63*, 96–113. [CrossRef]
14. Mahlberg, J.A.; Cheng, Y.T.; Bullock, D.M.; Habib, A. Leveraging LiDAR intensity to evaluate roadway pavement markings. *Future Transp.* **2021**, *1*, 720–736. [CrossRef]
15. Olsen, M.J.; Parrish, C.; Che, E.; Jung, J.; Greenwood, J. *Lidar for Maintenance of Pavement Reflective Markings and Retroreflective Signs*; No. FHWA-OR-RD-19-01; Oregon Department of Transportation: Salem, OR, USA, 2018. Available online: https://rosap.ntl.bts.gov/view/dot/36526 (accessed on 15 June 2023).
16. LTL-X Mark II Retroreflectometer Datasheet. Available online: https://roadsensors.com/wp-content/uploads/assets/docs/roadsensors/Leaflets/English/LTL-XL_LTL-X%20Mark%20-II%20_retroreflectometer_features.pdf (accessed on 15 June 2023).
17. Christ, M.; Braun, N.; Neuffer, J.; Kempa-Liehr, A.W. Time series feature extraction on basis of scalable hypothesis tests (tsfresh–a python package). *Neurocomputing* **2018**, *307*, 72–77. [CrossRef]
18. Tsfresh: Overview of Extracted Features. Available online: https://tsfresh.readthedocs.io/en/latest/text/list_of_features.html (accessed on 15 June 2023).
19. Schreiber, T.; Schmitz, A. Discrimination power of measures for nonlinearity in a time series. *Phys. Rev. E* **1997**, *55*, 5443. [CrossRef]
20. Tsfresh Feature Calculators (c3 Statistic). Available online: https://tsfresh.readthedocs.io/en/v0.11.1/api/tsfresh.feature_extraction.html#tsfresh.feature_extraction.feature_calculators.c3 (accessed on 15 June 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.