Pothole Classification using Point Cloud Data: a Comparison between Machine Learning and Deep Learning

Kristin Eggen[1], Hongchao Fan[1]

[1]Department of Civil and Environmental Engineering, Norwegian University of Science and Technology, Trondheim, Norway
(kriegg@ntnu.no, hongchao.fan@ntnu.no)

Keywords: Pothole, Classification, Point Cloud, Machine Learning, Deep Learning

Abstract

Automatic pothole detection is important for improving road maintenance and transportation safety. While image-based pothole detection often struggles under poor lighting and weather conditions, point cloud data provides a robust alternative by capturing detailed surface geometry. Machine learning has demonstrated strong performance in point cloud classification. While traditional machine learning is simpler and relies on handcrafted features, deep learning models are more powerful, as they learn complex, high-dimensional patterns directly from the input data. While most existing work relies on deep learning models, which are time-consuming to train and require extensive labelled datasets, potholes can be well described by geometric features, making pothole detection well-suited for feature engineering. This paper compares traditional machine learning and deep learning approaches for pothole classification using point cloud data, to evaluate whether the added complexity and data demands of deep learning models are justified, or if traditional machine learning techniques are sufficient for accurate classification. A dataset with labelled pothole instances is created to train both models. The machine learning approach uses manually engineered geometric features as input to an ensemble classifier, while the deep learning model is trained on sampled data. Experimental results show that the machine learning approach outperformed the deep learning model. These results suggest that for this particular task, where informative domain-specific features can be manually engineered, the machine learning approach offers a more practical and efficient solution for real-world deployment, where labelled data may be limited.

1. Introduction

Pothole detection is important for road maintenance to ensure safe and reliable transportation, as potholes can cause damage to vehicles and contribute to passenger discomfort. Manual inspections to discover potholes are labour-intensive, time-consuming, and costly. Automatic detection of potholes is therefore valuable, offering an accurate and faster solution for maintaining road infrastructure.

Mobile Laser Scanning (MLS), where a Light Detection and Ranging (LiDAR) sensor is mounted on a moving platform, allows for capturing accurate 3D representations of urban environments, making it well-suited for pothole detection. Point cloud data addresses the limitations of image-based detection, which might encounter challenges such as poor lighting conditions, shadows, and occlusion. In contrast, point cloud data provide detailed information of the scanned environment, offering a more robust and accurate method for identifying potholes, solely based on geometric features, such as depth and curvature variations.

Several researchers have leveraged point clouds to develop methods to accurately identify potholes. Ma et al. (2023) use MLS point cloud data to detect potholes by calculating the directed distance of points from a locally fitted pavement plane. Negatively skewed histograms of these distances are identified as indicators of true pothole depressions. Fan et al. (2020) present a robust pothole detection algorithm that combines disparity transformation and 3D road surface modelling using stereo vision.

Previous research has also focused on handling sparse or incomplete 3D data for pothole detection. Zhong et al. (2025b) introduce a boundary completion algorithm to reconstruct the geometry of occluded potholes. Faisal and Gargoum (2025) apply curvature estimation to low-density LiDAR scans, identifying high-curvature regions and characterising potholes by measuring their geometric attributes through boundary extraction and voxelization.

Machine learning is a branch of artificial intelligence that enables algorithms to learn from data and make informed predictions. It is a powerful tool which has shown great success in various domains. Deep learning, a subfield of machine learning, has revolutionised the field by enabling models to learn directly from the raw input data. In particular, Convolutional Neural Networks (CNNs) learn hierarchical feature representations through multiple layers to capture increasingly complex relations within the data, making them better suited for complex tasks. Existing research has therefore leveraged these methods to improve accuracy and efficiency for pothole detection (Cai et al., 2025; Zhong et al., 2025a; Dong et al., 2024; Talha et al., 2024; Nawale et al., 2023).

Despite the great performance of deep learning models, they require a large amount of labelled training data to learn informative features. Additionally, training deep learning models is time-consuming and computationally demanding. In contrast, while traditional machine learning[1] models depend on handcrafted features and domain knowledge to identify the most informative information from the data, they require less data with lower computational cost.

Existing studies have applied machine learning to identify potholes using smartphone sensor data, (Egaji et al., 2021; Wu

[1] Hereafter, the term machine learning refers to traditional algorithms, unless stated otherwise.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-447-2026 | © Author(s) 2026. CC BY 4.0 License.

et al., 2020) or image data (Dhanvanth Prasath et al., 2024; Jobayer Al Masud et al., 2021). In contrast, machine learning combined with 3D point cloud data is less explored.

However, since potholes can be well described through geometric features derived from point cloud data, this task is well-suited to handcrafted feature engineering. Although deep learning is more powerful than machine learning, the added complexity of the automatically learned features may not be justified for simpler tasks where domain-specific features can be manually engineered. It is therefore valuable to compare these two approaches in terms of performance, data requirements, and computational costs.

This paper will therefore apply traditional machine learning to classify potholes using point cloud data and evaluate how it compares to deep learning. First, point cloud data will be manually labelled and processed. Features will be manually engineered and used as input to a machine learning ensemble classifier. Then, a deep learning model will be trained using the same data. Both models will be evaluated on the same test set, consisting of challenging samples, to ensure a comprehensive evaluation of their performance.

Moreover, this performance comparison can reveal to what extent each approach can generalise to new data and how robust they are across different road conditions. The insights from this performance evaluation can guide the trade-offs between accuracy, computational demand, and hardware requirements, particularly in real-world deployment, where labelled data may be limited.

The remainder of this paper is organised as follows: Section 2 outlines the methodology. The experimental results are presented in Section 3, followed by a discussion in Section 4. Finally, Section 5 concludes this paper.

2. Methodology

This section outlines the methodology, describing the point cloud preprocessing, followed by the machine learning pipeline with feature engineering and the development of the ensemble classifier. Then, the deep learning approach, including sampling, data augmentation, and model training, is presented. The workflow is illustrated in Figure 1.

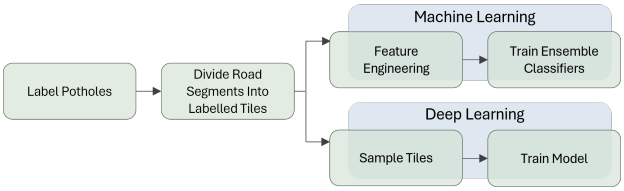

Figure 1. Overview of the workflow. Initially, potholes are manually labelled, and the point clouds are divided into tiles. For the machine learning approach, handcrafted features are extracted to train an ensemble classifier. For the deep learning approach, sampled tiles are used to train a neural network.

2.1 Data Preprocessing

The data are preprocessed by isolating road surfaces, including only points within a 2.5-meter buffer on each side of the road centreline, which are obtained from an additional shapefile. Several road segments are then extracted, and potholes are manually labelled.

To reduce the complexity of large point clouds and allow for better pattern recognition of potholes, each labelled road segment is divided into a regular grid structure, where each tile is treated as an independent sample. The grid is constructed by dividing the road segment along its centreline and perpendicular to the driving direction in 2-meter intervals, with a 0.5-meter overlap to ensure coverage.

Tiles are classified as ”pothole” or ”non-pothole”. Because a pothole can extend across multiple tiles, some neighbouring tiles may contain only a small number of pothole points near the boundary. To avoid labelling these boundary tiles as ”pothole” when no significant depression is present, only the tile or tiles where the pothole is dominant are labelled.

This is achieved by first clustering all points labelled as ”pothole” using DBSCAN. From a small range of predefined ϵ values, the ϵ that yields a number of clusters similar to the expected pothole count for the given road segment is selected. Then, for clusters intersecting more than one tile, cluster points falling inside each tile are counted. A tile is considered dominant for that cluster if it contains a sufficient portion of the cluster points. Otherwise, the tile is only touching the edge of the pothole and is labelled as a non-pothole tile. This reduces noisy pothole labels, improving label consistency, and enabling more reliable feature learning.

2.2 Machine Learning

Machine learning models depend on informative and diverse input features to learn to distinguish between classes. Therefore, a range of descriptive features is computed to describe the geometry of the potholes and the surrounding road surface. All features are computed using normalised coordinates.

The curvature feature highlights the contrast between smooth road surfaces and irregularities around potholes. The curvature is calculated at each point using a local neighbourhood with Gaussian weighting from the eigenvalue decomposition of the covariance matrix. The curvature feature is defined as

$$\kappa_i = \frac{\lambda_{1,i}}{\lambda_{1,i} + \lambda_{2,i} + \lambda_{3,i}}, \quad \text{Curvature} = \text{median}(C_{10}), \quad (1)$$

where κ_i denotes the local curvature and $\lambda_{1,i} \leq \lambda_{2,i} \leq \lambda_{3,i}$ are the eigenvalues of the covariance matrix at point i. C_{10} denotes the set of the 10 largest values of $\{\kappa_i\}$. To avoid errors caused by noise in the curvature distribution, the median is used to represent the most prominent surface change.

The global depth feature captures vertical deviations from the road plane, which can indicate deeper depressions. This feature is defined as

$$\text{Global Depth} = \sum_{i=1}^{N} \mathbf{1}(d_i < -\tau), \quad (2)$$

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-447-2026 | © Author(s) 2026. CC BY 4.0 License.

where d_i is the vertical distance from a globally fitted plane, estimated using RANSAC with a small threshold of 0.01 over 100 iterations, and τ is the depth threshold. In contrast, while this feature captures deviations from the overall plane, the local depth feature captures small-scale surface anomalies by counting points significantly lower than the average of their local neighbourhood. The local depth of the tile is defined as

$$\text{Local Depth} = \sum_{i=1}^{N} \mathbf{1}((z_i - \bar{z}_i) < -\tau), \tag{3}$$

where z_i is the height of point i and $\bar{z}_i$ is the average height of its local neighbourhood. Finally, the slope feature reflects how steep the road surface is in a small area. A plane is fitted to the neighbourhood of each point, and the local slope is defined as

$$\theta_i = \arccos(n_{z,i}), \quad \text{Slope} = \text{median}\big(\mathcal{S}_{10}\big), \tag{4}$$

where $n_{z,i}$ is the z-component of the unit normal vector of the fitted plane at point i, and θ_i is the slope angle. $\mathcal{S}_{10}$ denotes the set of the 10 largest values of $\{\theta_i\}$. The median of these values is used to represent the overall slope of the tile. Table 1 provides an overview of the computed features along with a short description.

Table 1. Overview of the feature set.

Feature	Description
Curvature	Median of the largest local curvature values.
Global Depth	Number of points notably below a globally fitted surface plane.
Local Depth	Sum of z-deviations in local neighbourhoods.
Slope	Median of the largest slopes computed in local neighbourhoods.

The heatmap in Figure 2 shows the Spearman correlation between the features and the target in the dataset. Both depth features show the highest correlation with the target, while the curvature and slope features show slightly lower correlation.

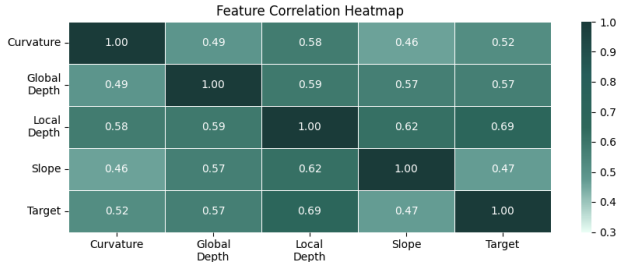

Figure 2. Heatmap showing the correlation between the features and the target.

Several machine learning models are trained on the dataset to find the best-performing ones for this task. Random Forest (RF) and Extreme Gradient Boosting (XGBoost) are combined into a weighted ensemble model to leverage the strengths of both classifiers. RF constructs multiple independent decision trees on different subsets of the data. The predictions from each tree are combined, reducing the risk of overfitting. In contrast, XGBoost sequentially adds decision trees to correct previous errors, minimising bias over iterations and capturing complex patterns in the data. By combining these classifiers, the ensemble model benefits from the robustness of RF while also leveraging the high accuracy capabilities of XGBoost, resulting in improved overall performance.

5-fold stratified cross-validation is used to obtain a robust estimate of model performance on the limited dataset. It reduces bias from a single train/validation split by averaging performance across folds. Moreover, to address the issue of class imbalance in the dataset, sample weights are applied to both models, assigning higher weights to the minority class (potholes). In addition, both models are manually fine-tuned on the validation set. Table 2 lists the hyperparameters used for both models.

Table 2. Hyperparameters.

Classifiers	Hyperparameters
Random Forest	Depth: 5 Estimators: 12 Random Seed: 42
XGBoost	Depth: 4 Estimators: 50 Learning Rate: 0.1 Random Seed: 42

2.3 Deep Learning

Deep learning network architectures demand a fixed input size, requiring sampling of each tile. Each tile is sampled to 2,048 points. On average, each tile contains 3,157 points, making downsampling the dominant operation. Downsampling is generally preferred over upsampling because it preserves the integrity of real measurements and avoids introducing synthetic or potentially noisy data.

Tiles containing more than 2,048 points are downsampled using Farthest Point Sampling (FPS), while tiles containing fewer than 2,048 points are upsampled by duplicating random points and adding a small amount of noise. Each tile is also normalised after sampling to fit within a unit sphere centred at the origin of the coordinate system.

The dataset is imbalanced, containing more non-pothole tiles than pothole tiles. Therefore, the minority class is oversampled by augmenting only pothole tiles during training. Augmentation is performed using a Principal Component Analysis (PCA)-based alignment strategy inspired by PCAlign (Zhang et al., 2024). Each pothole tile is first augmented using random point dropout, random scaling, and random shifting. PCA-aligned copies are then generated by flipping the sign of the dominant principal axis while constraining the smallest principal axis to point in the positive Z-direction. For every pothole tile, the original tile and the two PCA-aligned versions are kept.

Similar to the machine learning approach, cross-validation is also applied when training the deep learning model. New augmentations are generated within each fold for the pothole tiles in the training split, ensuring that the model sees varied versions of the minority class during training.

Several frequently employed deep learning models for point cloud classification are trained on the dataset. Each model is

trained for 50 epochs with a batch size of 32 using the Adam optimiser and a learning rate of 0.001. Binary Cross Entropy with Logits Loss is used for this binary classification task. Moreover, the optimal decision boundary for classification is determined by selecting the threshold that maximises the F1-score. Among the evaluated models, PointNet++ (Qi et al., 2017) was found to be the best-performing model.

The final prediction for each sample is obtained by averaging the probabilities across all folds, which reduces variance and improves the robustness of the final predictions. To determine a single global decision threshold, the out-of-fold validation predictions from all folds are aggregated, and the optimal threshold is selected by maximising the F1-score on the combined validation outputs.

3. Experiment

This section describes the point cloud data used in this research, followed by the results of the machine learning and deep learning approaches.

3.1 Experimental Dataset

The dataset is created from MLS point cloud data of the road environment in Trondheim, Norway. The data are acquired across urban, suburban, and rural areas in Trondheim municipality, capturing the road surface and surrounding infrastructure. Figure 3 shows one of these MLS point clouds with a pothole marked with a red circle.

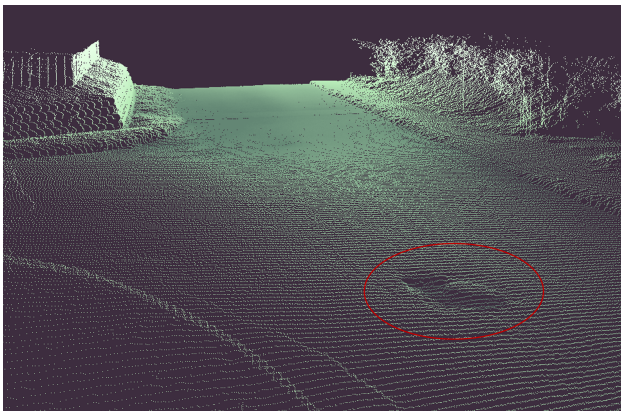

Figure 3. Example of point cloud with a pothole marked with a red circle.

Different road segments across the municipality are randomly selected, and the identified potholes are manually labelled using the open-source software CloudCompare. These road segments can be divided into three categories: asphalt roads, residential gravel roads (unpaved roads in residential neighbourhoods) and dirt roads (rural roads / unpaved country roads).

Table 3 lists the number of labelled potholes in the dataset. A total of 121 potholes across 38 road segments are used for training and validation, yielding 131 tiles with a pothole and 599 tiles without. This data is obtained from 15 asphalt roads, 12 gravel roads, and one dirt road.

A larger and more challenging test set is also developed to evaluate the models. This test set consists of 122 potholes across 37 road segments, resulting in 124 pothole tiles and 432 non-pothole tiles. This set spans five asphalt roads, 16 residential gravel roads, and 15 dirt roads.

Table 3. Overview of the labelled dataset.

	Train/Validation Set	Test Set
# Potholes	121	122
# Pothole Tiles	131	124
# No Pothole Tiles	599	432

Figure 4 shows the distribution of the road segments in the dataset across Trondheim. The green circles mark the road segments used for training and validation, while the red circles mark the road segments in the test set. Unlike most road segments used for training and validation, which are from urban and suburban areas, the test set presents a greater challenge due to the increased representation of uneven dirt roads.

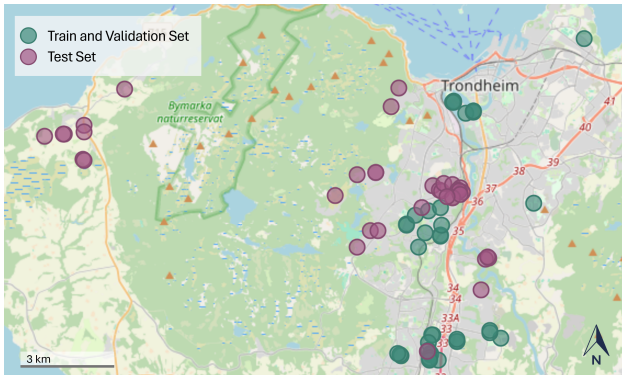

Figure 4. Distribution of road segments used for training and validation (green), and road segments used for testing (red).

3.2 Experimental Results

The independent test set is used to evaluate the performance of both the machine learning and deep learning model. The balanced accuracy, which better reflects performance on imbalanced data, precision, recall, and PR-AUC are used as evaluation metrics. While the machine learning classifiers are trained using an Intel Core Ultra 7 165U CPU @ 2.10 GHz, the deep learning model is trained using a Tesla V100-PCIE-32GB GPU.

Table 4 presents the balanced accuracy, precision, recall, and PR-AUC achieved by each model using cross-validation, and Table 5 shows the performance of both models for each road category, including asphalt, gravel, and dirt roads.

Table 4. Machine learning and deep learning results on the test set.

Model	Balanced Accuracy (%)	Precision (%)	Recall (%)	PR-AUC (%)
Machine Learning	89.8	68.3	91.9	86.3
Deep Learning	86.2	73.5	80.6	86.9

4. Discussion

This section evaluates the results, beginning with a discussion of the handcrafted features, followed by an analysis of

Table 5. Machine learning and deep learning results on the test set for each road category.

	Asphalt Road (60 Tiles)			Gravel Road (238 Tiles)			Dirt Road (258 Tiles)		
	A (%)	P (%)	R (%)	A (%)	P (%)	R (%)	A (%)	P (%)	R (%)
Machine Learning	90.5	66.7	88.9	89.3	81.8	87.0	91.8	56.8	100
Deep Learning	91.5	72.7	89.0	87.3	81.3	75.4	88.5	65.6	87.0

the sampling strategies and the performance of various deep learning models. Finally, this section provides a comprehensive comparison between the machine learning and deep learning approaches.

4.1 Machine Learning

As illustrated in the heatmap in Figure 2, several features exhibit moderate correlation. The strongest intercorrelation is observed between Slope and Local Depth, suggesting potential redundancy. However, experimental results indicate that discarding the slope feature leads to a decline in all evaluation metrics.

Similarly, the depth-related features are also moderately correlated. However, experimental research demonstrates that both are important for accurate predictions. In particular, removing Global Depth results in a decline in performance, with balanced accuracy reduced to 86.1%, precision decreasing to 62.8%, a recall of 87.1%, and a PR-AUC of 80.5%. These results show that correlated features can still be valuable for accurate predictions.

Figure 5 presents the SHAP Beeswarm plot for XGBoost, illustrating the contribution of individual features to the model's predictions. The curvature feature shows the strongest influence on predictions, with larger values increasing the likelihood of a pothole. Both Local Depth and Global Depth exhibit a strong contribution, though Local Depth shows a more pronounced separation where high values align more consistently with positive SHAP values. Slope shows a weaker impact compared to that of other features.

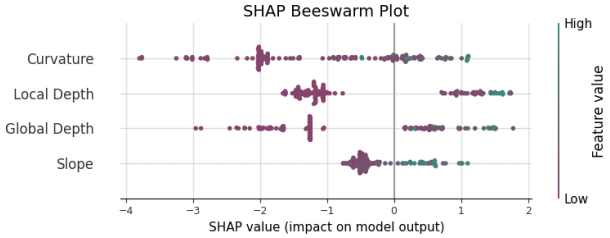

Figure 5. SHAP values for XGBoost.

4.2 Deep Learning

Deep learning architectures require a uniform input size, meaning resampling of tiles to a fixed number of points. Different sampling strategies are compared to assess how sampling impacts model performance. Although the simplest downsampling approach, by randomly removing points is computationally efficient, it likely discards structurally important points, leading to a loss of geometric information and poorer performance.

Unlike regular FPS, which selects points based on spatial distance, a curvature-weighted FPS strategy is evaluated. To include geometric importance when selecting points, distances are weighted using curvature values derived from the eigenvalues of local covariance matrices, where a power parameter controls the influence of the curvature.

Although high-curvature regions such as potholes are more likely to be preserved, the reduced overall performance compared to traditional FPS can be attributed to the loss of uniform spatial coverage. This curvature-focused sampling strategy tends to cluster points in high-curvature areas while leaving gaps in smooth, flatter regions, impacting the model's ability to learn the overall structures.

Instead of randomly upsample points, a strategy involving interpolating points in local neighbourhoods is evaluated. For each additional point needed, a random seed point is selected along with one of its k-Nearest Neighbours (k-NN). A new point is then interpolated at a random location along the line connecting these two points.

While this method is computationally more expensive than random upsampling, it preserves local geometry, which is likely to provide a more accurate sampling. However, in this case, where most tiles are downsampled rather than upsampled, a more advanced upsampling proves not to show a clear performance advantage.

Table 6 summarises the performance of various deep learning models with data augmentation and cross-validation applied. Among them, 3D-GCN (Lin et al., 2020) and CurvNet (Xiang et al., 2021) demonstrate the strongest performance. While DGCNN (Wang et al., 2019), GDANet (Xu et al., 2021), and PointConT (Liu et al., 2024) provide a moderate trade-off between precision and recall, they underperform compared to the other models. Although 3D-GCN achieves a higher precision than both PointNet++ and the machine learning classifier, recall is significantly lower. Compared to the machine learning model, neither of these models achieves the same balanced accuracy, indicating that their increased complexity might not translate to better performance for the given task.

Table 6. Performance of deep learning models on the test set.

Model	Balanced Accuracy (%)	Precision (%)	Recall (%)	PR-AUC (%)
3D-GCN	84.3	81.2	73.4	82.6
CurvNet	83.9	76.7	74.2	83.6
DGCNN	79.1	74.8	64.5	80.0
GDANet	81.2	78.5	67.7	77.1
PointConT	80.0	72.8	66.9	79.0

4.3 Comparison Between Performance of Machine Learning and Deep Learning Models

During cross-validation, the deep learning model shows low variability across folds, with a standard deviation of 0.024 for

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-447-2026 | © Author(s) 2026. CC BY 4.0 License.

balanced accuracy and 0.027 for PR-AUC, indicating stable performance across different train/validation splits. The machine learning model shows higher variability across folds, with a standard deviation of 0.047 for balanced accuracy and 0.059 for PR-AUC, suggesting it is more sensitive to how the data is split.

Handcrafted features can only provide a fixed and limited representation of each tile. These features may not capture all geometric variation present in the raw point clouds, making the model more sensitive to the specific distribution of data in each fold. In contrast, the deep learning model learns rich, high-dimensional features directly from the raw data and benefits from data augmentation, which improves generalisation and makes the model less dependent on the exact composition of each fold.

The overall results on the test set presented in Table 4 show that the machine learning model achieves a higher balanced accuracy and recall compared to the deep learning model. However, the deep learning model achieves higher precision, indicating that while machine learning is better at correctly identifying most potholes, deep learning is better at avoiding false positives overall. Despite, both models are better at identifying potholes than avoiding false positives, meaning they are more likely to assign a positive label when uncertain. Moreover, both models achieve comparable PR-AUC scores.

Table 5 shows the performance across the three different road categories in the test set. When evaluating the models' performance across the different road categories, the machine learning model performs best on gravel roads, while the deep learning model outperforms the machine learning model across all metrics on asphalt roads.

An unexpected decline in precision is observed for both models on asphalt roads compared to gravel roads. The smooth, uniform asphalt surface is expected to enhance pothole classification, whereas the irregular gravel roads pose a greater challenge. Analysis of the five asphalt roads in the test set revealed that one uneven asphalt segment produced several false positives. Besides misclassifying a manhole cover as a pothole, both models performed well on the other asphalt samples. Increasing the number of asphalt samples in the test set would likely reduce the influence of the irregular segment and improve performance.

The test set contains significantly more dirt road samples compared to the training and validation sets. Including these more challenging road samples allows for a more comprehensive evaluation. While the machine learning model achieves a perfect recall score for dirt roads, precision is sacrificed due to a substantial number of false positives. For this type of road, the deep learning model outperforms the machine learning model on precision. The lower precision, but higher recall, observed for both models is expected for dirt roads due to more surface variations leading to more false positive predictions.

4.4 Saliency Analysis of Machine Learning and Deep Learning Models

Saliency scores are computed for both models to further compare the two approaches. For the machine learning pipeline, saliency values are derived for all handcrafted geometric features, while the saliency scores for the deep learning model are obtained using the integrated gradients technique (Sundararajan et al., 2017). This analysis provides a better understanding of which regions of the point cloud are considered most important for classification. All tiles in the test set are used for this comparison.

To evaluate the relevance of each feature, the most salient points (defined using the 95th percentile) are matched with the ground truth points of each pothole. When averaging over all pothole tiles in the test set, the Global Depth feature achieves the highest alignment, with an overlap of 66.5%. This is followed by the Curvature, Local Depth, and Slope, which show progressively lower levels of overlap.

Figure 6 illustrates the most salient points for two pothole-tiles. Red points correspond to high saliency, while blue points correspond to low saliency. Global Depth has a high concentration of salient points in the pothole centre, whereas Local Depth shows a slightly more scattered distribution. Curvature and Slope focus on the pothole boundaries, capturing edge information that complements the depth-based features. In contrast, the integrated gradients technique provides a different saliency distribution, where the highlighted regions are less concentrated.

The saliency scores from the deep learning model show the lowest overlap with the ground truth, with an average of 38.8% across the test set. Unlike the geometry-based features, the salient points highlighted by the deep learning model that lie outside the pothole region are often located along the tile edges. This may indicate that the model learn that discontinuities at tile edges or missing neighbouring context correlate with pothole points.

While the machine learning model relies on geometric features, which are easy to interpret, the deep learning model can capture richer representations and more complex patterns in the data. The overlap between the handcrafted features and the integrated gradients saliency maps is computed to better understand the deep learning model's predictions. The overlap is measured using the 95th percentile of the saliency values.

Global Depth shows the highest average overlap (33.4%), followed by Local Depth, while Curvature and Slope exhibit the lowest overlap. The larger overlap with the depth-related features suggests this is the main indicator, while boundary and edge features are less important. Moreover, the differences in overlap also indicate that the deep learning model extends beyond the handcrafted features by incorporating additional contextual information that can not be captured by simpler, geometry-based features.

Moreover, to determine whether the most salient values reflect relevant or noisy information, they are assessed not only through their overlap with the ground truth but also by examining how often each feature highlights points in non-pothole tiles. For these computations, a global floor threshold is defined as the lowest saliency value among the 95th percentile salient points across all pothole tiles within a given road segment. This threshold is then applied to non-pothole tiles to identify any highly salient points.

The feature importance in Figure 5 revealed that Slope is the least important feature, and it also has the lowest overlap with ground truth points. At the same time, 5.4% of points in non-pothole tiles are above the global floor threshold, which is the highest portion of salient points in non-pothole tiles among

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-447-2026 | © Author(s) 2026. CC BY 4.0 License.

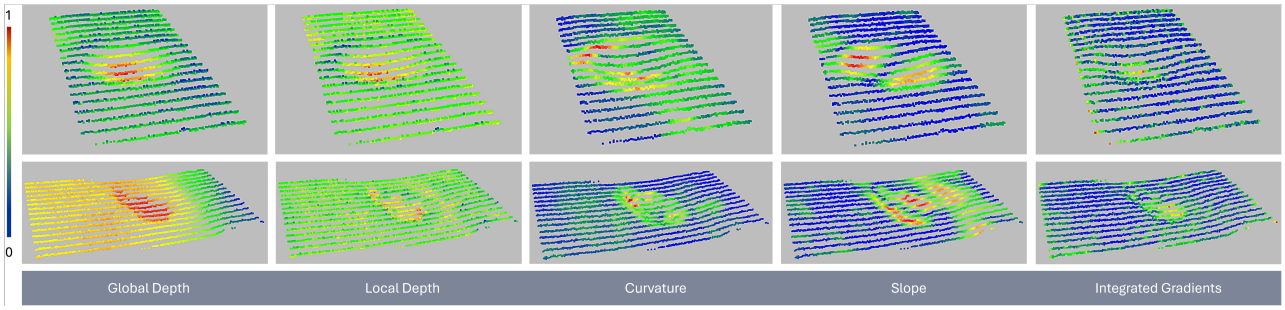

Figure 6. Saliency scores for two pothole tiles using handcrafted features (Global Depth, Local Depth, Curvature, Slope) and the deep learning model with Integrated Gradients. Red points indicate the most salient values, while blue points correspond to low saliency.

all of the handcrafted features. This indicates that this feature is more likely to introduce additional noise compared to the others. In contrast, Global Depth is the most robust feature, with only 2.0% of points in non-pothole tiles exceeding the threshold, outperforming all other features.

Figure 7 compares the ground truth pothole regions with the most salient points identified by the handcrafted features and the integrated gradients method. The dirt road segment in this example has lower vegetation along the right road edge, which introduces additional surface irregularities. For pothole tiles, red points represent the top 95th percentile of salient values, whereas for non-pothole tiles, they indicate points exceeding the global floor threshold.

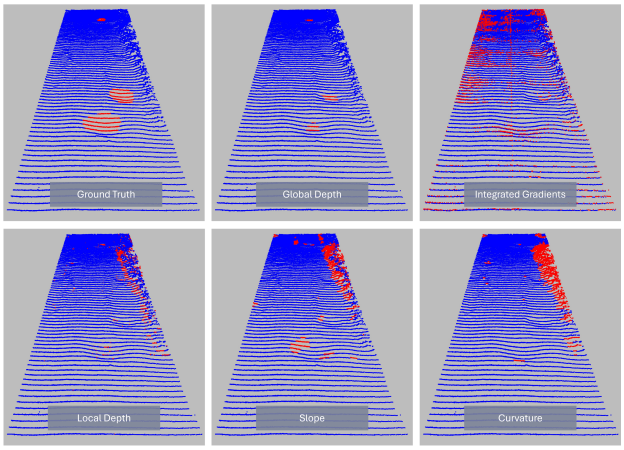

Figure 7. Comparison of salient points with ground truth pothole regions for a dirt road segment. The top row shows the ground truth, Global Depth, and Integrated Gradients saliency, while the bottom row displays Local Depth, Slope, and Curvature. For pothole tiles, red points represent the top 95th percentile of salient values, while for non-pothole tiles, red points correspond to saliency above the global floor threshold.

The ground truth frame shows three potholes marked in red. Among the handcrafted features, Global Depth aligns best with the ground truth, highlighting pothole depressions without adding noise. The other handcrafted features focus more on the noisy vegetation and barely highlight the potholes. In contrast, the integrated gradients method largely ignores the vegetation noise and focuses on potholes. Yet it strongly focuses on local surface variations on the left side of the road, which are mostly avoided by the handcrafted features.

The findings in Figure 7 illustrate the fundamental difference between handcrafted features and deep learning. The limitation of the traditional feature engineering becomes apparent when dealing with more complex and noisy inputs (dirt roads), as the features are constrained by explicit geometric definitions. This makes them sensitive to local shape changes, resulting in poorer performance when other geometric variations are present within the tile.

Notably, it is also observed in other segments that the downward tilt of the road surface near the edge, particularly pronounced on dirt roads, can resemble the depth depression of a pothole, leading the machine learning model to falsely classify these tiles as potholes. The fixed feature definitions of machine learning, such as Slope, can be triggered to cause false positives.

In contrast, the deep learning model can leverage complex, high-dimensional contextual patterns to avoid such noise. However, it also assigns higher saliency to general road irregularities, with 7.3% of points in non-pothole tiles across the test set having a saliency score above the global floor threshold, which is the largest among all features.

5. Conclusion

This paper explores the use of traditional machine learning for pothole classification from MLS point cloud data and compares the performance against deep learning approaches. The machine learning pipeline uses manually engineered features with an ensemble classifier, while a neural network is trained on sampled point cloud tiles for the deep learning approach. Both approaches were evaluated on a challenging test set.

The experimental results indicate that traditional machine learning is a suitable approach for this task. Although the deep learning model benefits from data augmentation, it is outperformed by the machine learning approach. These findings suggest that for the pothole classification task, where informative domain-specific features can be manually engineered, a feature-based method can achieve better performance with lower computational cost and fewer required resources.

By evaluating the performance of the different approaches, this research offers insight into balancing accuracy with computational cost and system requirements, which is valuable for real-world applications where labelled training data and computational resources may be limited.

Acknowledgements

This research received funding from the MobilitetsLab Stor-Trondheim (MoST) project:
https://mobilitetslabstortrondheim.no/en/.

References

Cai, Y., Deng, M., Xu, X., Wang, W., Xu, X., 2025. Road Pothole Recognition and Size Measurement Based on the Fusion of Camera and LiDAR. *IEEE Access*, 13, 46210-46227. https://doi.org/10.1109/ACCESS.2025.3549835.

Dhanvanth Prasath, M., Avinash, M., Prabu, M., 2024. Comparative analysis of machine learning models for smart pothole detection: Enhancing road safety and maintenance efficiency. *2024 Global Conference on Communications and Information Technologies (GCCIT)*, 1–6. https://doi.org/10.1109/GCCIT63234.2024.10862602.

Dong, J., Wang, N., Fang, H., Lu, H., Ma, D., Hu, H., 2024. Automatic augmentation and segmentation system for three-dimensional point cloud of pavement potholes by fusion convolution and transformer. *Advanced Engineering Informatics*, 60, 102378. https://doi.org/10.1016/j.aei.2024.102378.

Egaji, O. A., Evans, G., Griffiths, M. G., Islas, G., 2021. Real-time machine learning-based approach for pothole detection. *Expert Systems with Applications*, 184, 115562. https://doi.org/10.1016/j.eswa.2021.115562.

Faisal, A., Gargoum, S., 2025. Cost-effective LiDAR for pothole detection and quantification using a low-point-density approach. *Automation in Construction*, 172, 106006. https://doi.org/10.1016/j.autcon.2025.106006.

Fan, R., Ozgunalp, U., Hosking, B., Liu, M., Pitas, I., 2020. Pothole Detection Based on Disparity Transformation and Road Surface Modeling. *IEEE Transactions on Image Processing*, 29, 897-908. https://doi.org/10.1109/TIP.2019.2933750.

Jobayer Al Masud, A., Sharin, S. T., Shawon, K. F. T., Zaman, Z., 2021. Pothole detection using machine learning algorithms. *2021 15th International Conference on Signal Processing and Communication Systems (ICSPCS)*, 1–5. https://doi.org/10.1109/ICSPCS53099.2021.9660216.

Lin, Z.-H., Huang, S.-Y., Wang, Y.-C. F., 2020. Convolution in the cloud: Learning deformable kernels in 3d graph convolution networks for point cloud analysis. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1797–1806. https://doi.org/10.1109/CVPR42600.2020.00187.

Liu, Y., Tian, B., Lv, Y., Li, L., Wang, F.-Y., 2024. Point Cloud Classification Using Content-Based Transformer via Clustering in Feature Space. *IEEE/CAA Journal of Automatica Sinica*, 11(1), 231-239. https://doi.org/10.1109/JAS.2023.123432.

Ma, X., Yue, D., Li, S., Cai, D., Zhang, Y., 2023. Road potholes detection from MLS point clouds. *Measurement Science and Technology*, 34(9), 095017. https://doi.org/10.1088/1361-6501/acdb8d.

Nawale, S., Khut, D., Dave, D., Sawhney, G., Aggrawal, P., Devadkar, K., 2023. Potholeguard: A pothole detection approach by point cloud semantic segmentation. *2023 International Conference on Modeling E-Information Research, Artificial Learning and Digital Applications (ICMERALDA)*, 138–143. https://doi.org/10.1109/ICMERALDA60125.2023.10458201.

Qi, C. R., Yi, L., Su, H., Guibas, L. J., 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (eds), *Advances in Neural Information Processing Systems*, 30, Curran Associates, Inc. https://doi.org/10.48550/arXiv.1706.02413.

Sundararajan, M., Taly, A., Yan, Q., 2017. Axiomatic attribution for deep networks. *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, JMLR.org, 3319–3328. https://doi.org/10.48550/arXiv.1703.01365.

Talha, S. A., Manasreh, D., Nazzal, M. D., 2024. The Use of Lidar and Artificial Intelligence Algorithms for Detection and Size Estimation of Potholes. *Buildings*, 14(4). https://doi.org/10.3390/buildings14041078.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., Solomon, J. M., 2019. Dynamic Graph CNN for Learning on Point Clouds. *ACM Trans. Graph.*, 38(5). https://doi.org/10.1145/3326362.

Wu, C., Wang, Z., Hu, S., Lepine, J., Na, X., Ainalis, D., Stettler, M., 2020. An Automated Machine-Learning Approach for Road Pothole Detection Using Smartphone Sensor Data. *Sensors*, 20(19). https://doi.org/10.3390/s20195564.

Xiang, T., Zhang, C., Song, Y., Yu, J., Cai, W., 2021. Walk in the cloud: Learning curves for point clouds shape analysis. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 915–924. https://doi.org/10.48550/arXiv.2105.01288.

Xu, M., Zhang, J., Zhou, Z., Xu, M., Qi, X., Qiao, Y., 2021. Learning Geometry-Disentangled Representation for Complementary Understanding of 3D Object Point Cloud. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(4), 3056-3064. https://doi.org/10.1609/aaai.v35i4.16414.

Zhang, C., Li, A., Zhang, D., Lv, C., 2024. PCAlign: a general data augmentation framework for point clouds. *Scientific Reports*, 14. https://doi.org/10.1038/s41598-024-72264-8.

Zhong, J., Kong, D., Wei, Y. et al., 2025a. YOLOv8 and point cloud fusion for enhanced road pothole detection and quantification. *Scientific Reports*, 15, 11260. https://doi.org/10.1038/s41598-025-94993-0.

Zhong, J., Kong, D., Wei, Y., Yang, Y., Su, Z., 2025b. A novel method for pothole detection based on incomplete point clouds. *Measurement*, 252, 117344. https://doi.org/10.1016/j.measurement.2025.117344.