Reports on Geodesy and Geoinformatics, 2026, Vol. 121, pp. 19–42

DOI: 10.2478/rgg-2026-0002
Received: 10 October 2025 / Accepted: 20 March 2026
Published online: 15 April 2026

ORIGINAL ARTICLE

Analysis of the impact of TLS point cloud feature sets on the detection of building displacements using machine learning algorithms

Ewa Świerczyńska [1*] and Damian Wojda [1]

[1]Faculty of Geodesy and Cartography, Warsaw University of Technology, Pl. Politechniki 1, 00-661 Warsaw, Poland

*ewa.swierczynska@pw.edu.pl

Abstract

The study addresses the problem of detecting displacements of building walls using terrestrial laser scanning (TLS) data and machine learning methods. Traditional displacement measurement techniques are often time-consuming. They are also limited in capturing the full geometry of monitored objects. In response, this research proposes a methodology based on the analysis of geometric and radiometric features extracted from point clouds. Controlled experiments were conducted with a geodetic rosette equipped with distance-measuring prisms, which were displaced in the XY plane by 4 mm, 9 mm, and 13 mm. Data were recorded with a Leica RTC360 scanner from three stations, yielding nine point clouds. Selected features describing differences between corresponding points in the reference and displaced series were used as input for neural network models. Both binary classification (displacement/non-displacement) and multi-class classification (0, 4, 9, 13 mm displacement) were performed. The results demonstrated high classification accuracy: 99.1% for binary models and 96.0% for multi-class models. Feature ranking revealed that geometric attributes, such as displacement vector length, curvature, and normal vectors, were the most relevant for model training, while color features had minor importance. The study confirmed that scanner position and incidence angle of the laser beam strongly affect classification quality. The developed procedure proved effective in detecting displacements that occur in directions parallel to the plane of building walls. The conclusions drawn from the research constitute a valuable contribution to the theory of monitoring building structures using TLS.

Key words: machine learning, feature, point cloud, TLS, classification, displacement

1 Introduction

Ensuring the structural integrity and safe operation of buildings, particularly strategic infrastructure and heritage sites, depends on the systematic monitoring of their technical condition. Modern engineering surveying increasingly relies on terrestrial laser scanning (TLS) to obtain dense 3D point clouds that represent the complete geometry of investigated surfaces. Compared to traditional discrete point measurements, this remote sensing technology provides a significantly more detailed perspective on structural behavior, facilitating the detection of subtle deformations that may signal structural degradation.

However, the efficient interpretation of these vast spatial datasets and the automation of change-detection workflows remain primary challenges in the field. Although machine learning algorithms are frequently employed for point cloud analysis, the optimal selection of feature sets to describe individual points remains a critical and open research problem. The inclusion of redundant or irrelevant geometric and radiometric attributes can compromise classification accuracy or lead to excessive computational overhead, limiting the practical utility of these models in structural diagnostics.

This article aims to assess how the selection of different TLS point cloud feature sets affects the performance of machine learning algorithms in detecting building displacements. The study evaluates the relevance of these features for both binary and multi-class classification scenarios, specifically targeting deformations occurring within the plane of the monitored structural elements.

The subsequent sections of the paper describe the developed methodology for feature extraction and model training, followed by an analysis of the classification results and the influence of measurement geometry on data reliability. Finally, the study concludes with a discussion on the methodological limitations and identifies potential directions for future research.

1.1 Determination of displacements and deformations of building elements

The study of displacement and deformation of building walls is a main element of structural diagnostics. It allows for the assessment of the stability and safety of structures.

Deformation is defined as a change in shape or dimensions under the influence of forces. In geodesy, it means a change in the relative position of points on a solid without affecting its continuity (Prószyński and Kwaśniak, 2015). Displacement, on the other hand, is defined as a change in the position of a point on an object in space relative to its initial position. In mechanics, displacement describes the vector difference between the position of a point in a deformed state and its position in its original state (Zienkiewicz et al., 2013). In geodetic terms, displacements refer to the difference in the coordinates of structural points, determined in successive measurement periods (Prószyński and Kwaśniak, 2015). Displacement measurements form the basis for determining deformation. Only the analysis of a set of control points allows for the description of the changes in the shape of the entire structure (Gili et al., 2000).

Methods for determining deformation and displacement have changed with the development of technology. Methods such as terrestrial laser scanning (Zaczek-Peplinska and Popielski, 2013) and UAV photogrammetry from the ground or from low altitude (Świerczyńska et al., 2024) have become increasingly important. These methods enable the acquisition of dense 3D point clouds and precise detection of changes in the geometry of entire surfaces (Becker et al., 2024). Traditional measurement methods, such as levelling, tachymetry, or GNSS, provide high accuracy but limit the analysis to selected points (Sztubecki et al., 2022; Karsznia et al., 2023). Due to the lack of complete mapping of object surfaces, geodetic monitoring must be supplemented with data obtained using crack meters, inclinometers, or strain gauges (Scaioni et al., 2018). Remote methods are gaining an increasing advantage over traditional methods. Moreover, with the development of technology, the role of data analysis automation is growing (Besl and McKay, 1992; Łępicka and Kornuta, 2017). Machine learning algorithms support point cloud classification, surface segmentation, and the identification of deformation trends (Karsznia et al., 2025). Displacement and deformation monitoring is gaining new value thanks to the use of artificial intelligence and machine learning methods. However, proper analysis of construction processes is only possible with proper validation and verification of models using real data (Bao et al., 2023, 2025; Zhuang et al., 2022).

1.2 Comparison of point clouds acquired periodically

In the context of deformation and displacement analysis based on data obtained using TLS, the mutual orientation of point clouds registered in different measurement cycles plays a key role. There are always certain deviations in position and orientation between data sets registered periodically.

The issue of comparing point clouds is discussed in the literature on the subject. Si et al. (2022) compare different registration algorithms in the context of laser measurements of large objects, discussing the strengths and weaknesses of registration methods and highlighting flat surfaces with limited geometric characteristics. Peng et al. (2023) propose the use of FPFH descriptors in combination with RANSAC selection for registering clouds from different sensors, which allows for a partial solution to the noise problem.

Intensive research is being conducted to develop algorithms that are resistant to point cloud imperfections and data gaps. Probabilistic methods (CPD) are currently used and are based on the local geometry of the investigated surface, such as normal information (Liu et al., 2021). A comprehensive review of deep learning-based point cloud registration methods was presented by Chen et al. (2024), with an analysis incorporating both fully overlapping and partially overlapping clouds. The authors investigate methods to enhance network performance and discuss their applications in various fields. The article (Yang et al., 2023) presents a new way of describing 3D features based on the LRF method and spatial voxel homogenization. The point cloud development procedure, tested on three datasets, shows better performance than existing methods and increases descriptiveness and resistance to interference.

Scan orientation involves combining multiple point clouds into a single coherent 3D model. This procedure can be performed using the ICP (Iterative Closest Point) algorithm, which iteratively matches points from two point clouds by minimizing the average distances between points of all pairs (Pomerleau et al., 2015; Rusinkiewicz and Levoy, 2001). One of the most important stages of the ICP algorithm is point association. For each point of the reference cloud, the nearest point of the other cloud is searched for. Association can be performed in various ways, depending on the similarity of the metric used. The simplest and most commonly used approach is the point-to-point metric, where the Euclidean distance is calculated for each pair of points. The point-to-point metric is based on the nearest neighbor method and is widely used for point association, with the effectiveness depending on the precise orientation of the clouds. In more advanced variants, the association process is not based solely on XYZ coordinates, but other point features are considered as well, which increases the effectiveness of the fitting. The association can use geometric point features (e.g., normal vector), radiometric point features (e.g., reflectance intensity, RGB), or both types simultaneously. Depending on the characteristics of point clouds, new methods are constantly being developed that use the ICP algorithm (Bae et al., 2025; Yuan et al., 2025; Zhao et al., 2024; Chen et al., 2024; Hexsel et al., 2022). In the case of high-density clouds, basic geometric features such as spatial coordinates or normal vectors are often sufficient. However, for less dense clouds, especially when there is no geometric surface continuity, it may be necessary to use more advanced descriptors, such as FPFH, 3DSC, SHOT, SIFT, KAZE, or FREAK (Rusu et al., 2009; Amamra and Boumaza, 2019; Tareen and Saleem, 2018; Yang et al., 2025). Experiments have shown that the hybrid method (NDT + PL-ICP) significantly reduces scan orientation errors, particularly those obtained using SLAM with LiDAR in large-scale measurements, compared to traditional approaches (Zhang et al., 2025).

Incorrect orientation can lead to false fittings and, consequently, to incorrect analysis results. If the function finds a point that is further away than the standard distance, this may indicate actual displacement or deformation of the object, but it may also indicate a registration error or the presence of outliers.

1.3 Theory of machine learning

Machine learning is a rapidly developing branch of artificial intelligence. Its goal is to enable computers to acquire knowledge and make decisions based on data, without the need to program rules each time. The input data provided to the model is referred to as independent variables, while the values that the model tries to predict are dependent variables or labels. Input variables are features that describe an object or phenomenon, and their set for a single example is called a feature vector. Based on such data, a model is constructed, i.e., a mathematical structure that, after an appropriate learning process, can predict output values. This process is carried out using a training set on which the model learns, and its

effectiveness is verified using a test set (Spears et al., 2018; Hardt and Recht, 2021).

The literature on the subject distinguishes several basic machine learning paradigms (Syed and Lokhande, 2024; Alnuaimi and Albaldawi, 2024; Rathod, 2024). In supervised learning, the model is trained on labeled data, i.e., data for which both the features and their corresponding labels are known. This type of data structure enables effective training of predictive models whose task is to classify new cases based on patterns recognized in the training set (Li and Pisztora, 2021). In an unsupervised approach, the algorithm operates on unlabeled data, searching for hidden patterns or groups. There are also intermediate methods, such as semi-supervised learning, which combine a small number of labeled examples with a large unlabeled dataset, thereby improving the overall quality of predictions (Prakash and Nithya, 2014). Deep learning, based on multi-layer neural networks, is also gaining in importance, enabling the automatic extraction of increasingly complex data representations and achieving high performance in classification and regression tasks (LeCun et al., 2015).

After building a dataset and training various classification models, the key step is to evaluate their accuracy. Due to the diverse types of input data and the numerous features that can impact the effectiveness of the model, there is no universal algorithm that is optimal in all conditions; the best choice depends on the specific case. The evaluation is performed on a held-out test set that was not used for model training.

The main evaluation tool is the confusion matrix, which divides the classification results into four categories: True Positives, True Negatives, False Positives, and False Negatives (Vakili et al., 2020). Based on this, various statistical metrics are calculated: overall accuracy, kappa coefficient, precision, recall, and F1-score. These metrics allow us to compare how well the models perform and to account for both FP and FN errors, which is important when classes are unevenly represented (Crall, 2023).

1.4 Feature selection in machine learning

Feature selection is the process of reducing the dimensionality of an input dataset by selecting a subset of features that have a key impact on the capacity of the model to correctly distinguish between classes. The predictive accuracy of a classification model depends on the number and quality of the input features used. The use of redundant, unnecessary, or insignificant features can lead to a decrease in the effectiveness or overfitting of the model to the training data. Each feature represents an additional dimension in the operational space of the model. Inclusion of redundant, unnecessary, or insignificant features leads to a high dimensionality that can complicate the search for optimal decision boundaries. Various feature selection algorithms can effectively assess the impact of features on the model by ranking their importance.

In applications such as MATLAB, within the Classification Learner tool, algorithms such as Minimum Redundancy Maximum Relevance (MRMR), Chi-square, ReliefF, ANOVA, and Kruskal-Wallis are available. These algorithms allow you to generate a feature importance ranking, so that features with a high score are considered key to the model. Research shows that combining ReliefF and MRMR methods can improve feature selection efficiency. First, ReliefF selects candidates, then MRMR reduces redundancy, resulting in a concise and effective feature set (Zhang et al., 2008). A comparison of MRMR, ReliefF, and Chi-square in the context of MATLAB shows that good feature selection algorithms allow for similar or better model accuracy with fewer features (Knight et al., 2023).

MRMR is an algorithm that finds the optimal set of features that can effectively represent a variable response. The algorithm minimizes redundancy, i.e., excess information, and maximizes relevance. This means that the selected features provide the most unique information needed to predict the class, eliminating repetitive information (Peng et al., 2005).

The main assumption of the ReliefF algorithm is to assess the significance of variables by comparing attribute values between observations belonging to the same class and different classes (Robnik-Šikonja and Kononenko, 2003). This mechanism uses the nearest neighbors of a given sample and determines the extent to which a given feature contributes to the correct differentiation of objects. ReliefF is used in models that analyze distances between observations, such as k-NN or SVM.

1.5 Motivation for research

Displacements and deformations of building walls are an important indicator of the deteriorating condition of structures. The motivation for research stems from the growing need for effective monitoring of the technical condition of buildings. Traditional measurement methods often prove insufficient due to limitations in terms of time consumption and the need for discretization of objects. In response to these challenges, the importance of machine learning methods is growing, enabling the automation of data analysis and increasing the efficiency and repeatability of displacement assessment processes. The introduction of such tools allows for faster identification of potential structural problems. Predicting the development of changes supports the decision-making process regarding the maintenance and repair of buildings.

On the other hand, machine learning methods are increasingly being used to analyze large data sets. For example, the use of Random Forest (RF), Support Vector Machine (SVM), AdaBoost (ADA), and Stacking Ensemble (STACK) algorithms made it possible to eliminate gaps, anomalies, and discontinuities in the raw dataset. This dataset consisted of hourly readings from six automatic weather stations around Lake Titicaca (Sirpa-Poma et al., 2025). In recent years, there have been many publications emphasizing the role of deep learning with various neural network architectures (including CNN, LSTM, and hybrid networks. These are used in the processing and interpretation of InSAR data, which is a radar technique allowing for high-precision monitoring of ground surface deformation (Fontes et al., 2025). Deep learning is also used to detect and classify geometric changes in 3D point clouds, for example, to detect changes that may have more than two types (multiclass) in an urban environment between two measurements, using the DC3DCD method – DeepCluster 3D Change Detection (De Gélis et al., 2023) or using the Encoder Fusion SiamKPConv (De Gélis et al., 2024). Comprehensive methodologies are being developed for detecting and analyzing structural defects in various objects using data obtained from Terrestrial Laser Scanning (TLS) and machine learning algorithms. TLS provides dense 3D point clouds that reflect the geometry of the observed structures (e.g., walls, building elements, structures) in high resolution, while ML models are used to identify, classify, and analyze defects and deformations in this data (Dey et al., 2024; Zhou et al., 2025).

The subject of this thesis was the analysis of selected features of point clouds as input data for a machine learning model based on neural networks. The main objective was to optimize the process of detecting displacements of building elements. The measurement data were collected during an experiment involving the recording of simulated displacements of prisms mounted on the wall of a building. As a result of the research, the most efficient set of point cloud features was identified, based on which it is possible to detect displacements using a proprietary procedure that enables binary or multi-class classification. In the authors' opinion, compared to previous studies, this work introduces a new approach to displacement detection using machine learning algorithms. The innovation of the research lies in focusing not so much on the algorithms themselves and their hyperparameters, but on the relevance of features for binary and multi-class classification. Another key contribution

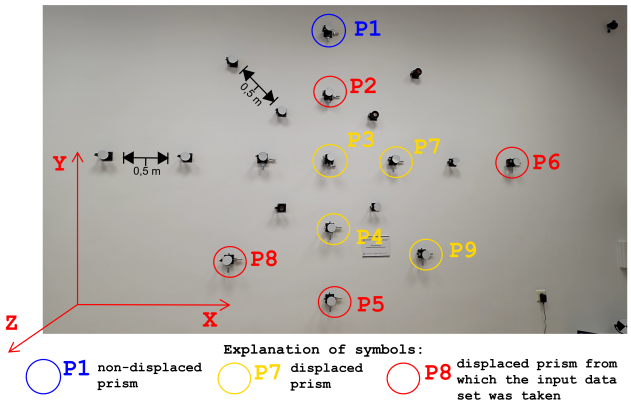


Figure 1. Arrangement of prisms on the rosette – photo taken during measurements

is the in-depth analysis of a specific type of displacement, namely displacements in the plane of a wall element, detected based on TLS data. It should also be noted that the experiment was conducted in laboratory conditions using a professional rosette, which enabled the simulation of controlled displacements.

The article has the following structure. Section 2 describes the methodology of the research and the theoretical aspects of the machine learning model analysis, outlines the specifications of the equipment used in the study, and characterizes the data collected. Section 3 presents the study results and is divided into two subsections: the first focuses on binary classification results and the second on multi-class classification outcomes. Section 4 compares the results obtained with existing studies using similar technologies. Finally, Section 5 presents a summary and concluding remarks.

2 Materials and methods

The article analyzes in detail the impact of input data sets on the capability of the developed algorithm to detect displacements occurring in the plane of building walls.

The proposed methodology for displacement detection relied on identifying differences between point clouds acquired in successive measurement epochs. The classification task was defined in two variants. The first, a binary problem, involved distinguishing between displaced and undisturbed fragments of the cloud. In the second, a multi-class problem, points were assigned to one of four classes based on the magnitude of displacement (no displacement, displacement equal to 4 mm, displacement equal to 9 mm, displacement equal to 13 mm). The following subsections describe the experimental part of the research and the method used to analyze the measurement results.

2.1 Description of the research object

The measuring device used in the experiment was a geodetic rosette designed for surface deformation testing, which is located in the Displacement and Deformation Monitoring Laboratory at the Warsaw University of Technology. Its design allows for the testing of controlled displacements thanks to the use of two types of distance-measuring prisms:

- a prism fixed in the housing, serving as a reference point,
- a prism mounted on a micrometric table, allowing for precise adjustment of its position.

The rosette design permits the controlled displacement of the distance measuring prisms in the XY plane (within the rosette plane) and, for the central prism, also along the Z axis (perpendicular to the rosette plane). The prism arrangement is shown in Figure 1.

Figure 1 shows a rosette with marked prisms that played a key role in the experiment. Prism P1 is generally located on a micrometric table, but during the experiment, it remained unmoved for nearly all measurement series, with its position changed by only 4 mm during the simulation of one measurement epoch from station 2. Located at the very top of the rosette, in theory, it was recorded at the lowest scanning resolution. Its stability was used to test the sensitivity of machine learning algorithms to the uneven distribution of points on the cloud.

The remaining prisms, marked in Figure 1 and numbered from P2 to P9, were moved by 4 mm and 13 mm. Prisms P2, P4, P5, P6, P7, P8, and P9 were displaced in directions parallel to the wall surface, i.e., along the X and Y axes of the local object system. Prism P3, on the other hand, was displaced in a direction perpendicular to the wall surface, i.e., along the Z axis of the local object system. However, it should be noted that this study focused exclusively on displacements in the XY plane, ignoring changes in the Z-axis direction. Accordingly, during the experiment, prism P3 was moved but excluded from the analysis. The study, therefore, examined changes in the object occurring in its plane rather than those perpendicular to the surface. Displacements perpendicular to the wall surface will be addressed in future research, as they require a separate approach to the problem and the use of a different set of input features.

It should also be noted that prisms moved in directions parallel to the wall surface can be divided into two groups. Prisms P2, P5, P6, and P8 provided input data samples to train machine learning models, while prisms P4, P7, and P9 were excluded from a training set and served as a basis for assessing the effectiveness of the trained models.

2.2 Measuring instruments

The measurements were performed using a Leica RTC360 (TLS) terrestrial laser scanner. This instrument operates using laser pulse transit time measurement technology with waveform digitizing (WFD). The key features of the scanner include:

- 360° × 300° field of view,
- range from 0.5 m to 130 m,
- maximum recording speed of 2 million points/s,
- distance measurement accuracy of 1.0 mm + 10 ppm,
- 3D accuracy of the point is 1.9 mm @ 10 m.

2.3 Experiment procedure

The measurement data were obtained by recording the object from three independent stations. The first position was located directly opposite the rosette, while the second and third were located to the left and right of the rosette. This strategy minimized systematic errors and allowed for the assessment of the stability of the results. A point cloud representing the laboratory room where the rosette is located is shown in Figure 2. The rosette is marked in black, while the locations of the TLS during the experiment are marked in red (center), blue (left), and green (right).

The average distance between the stations and the rosette was approximately 4.0–5.0 m. The geometry of the measuring structure is shown inFigure 3 in a horizontal cross-section (a) and a vertical cross-section (b). The figure also shows approximate distances between the stations and the object, as well as approximate angles of incidence of the laser beam on the object surface. The angle of incidence is decomposed into horizontal and vertical components.

Three measurement series were performed at each station, corresponding to simulations of successive measurement epochs:

- series 0 (reference) – no prism displacement,
- series 1 – simulated prism displacement of 4 mm in the X-axis

Table 1. The size of the analyzed point clouds obtained from three stations in each series

Displacement between series	Prism displacement [mm]	Size of the point cloud being analyzed		
		TLS station 2 - left	TLS station 1 - central	TLS station 3 - right
0–1	4	2387968	3836012	3781836
1–2	9	2374710	3842591	3861657
0–2	13	2365082	3849213	3780574

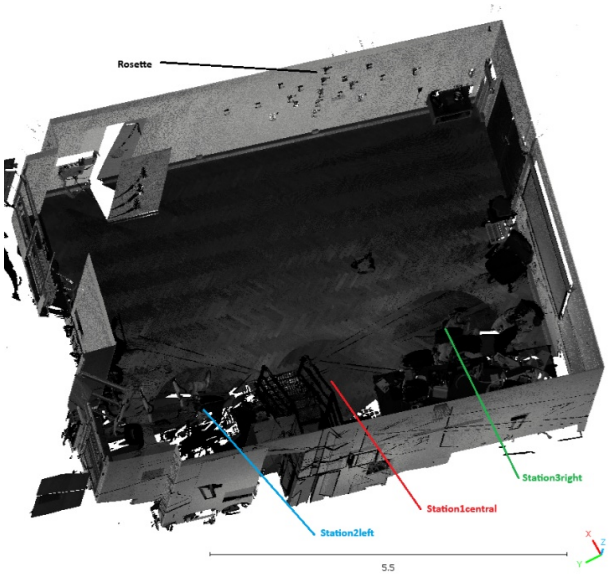


Figure 2. A point cloud representing the laboratory room where the rosette is located – marked in black. The figure also shows the locations of the TLS stations during the experiment, marked in red for the central station, blue for the left station, and green for the right station.

direction and 4 mm in the Y-axis direction,

- series 2 – simulated displacement of prisms by 13 mm in the X-axis direction and by 13 mm in the Y-axis direction.

Table 1 presents the sizes of point clouds obtained from three stations in individual measurement series.

A total of 9 point clouds were obtained and used as input data for further analysis. The average resolutions of point clouds representing the rosette in its extreme fragments were as follows:

- From station 1 (central), point clouds were obtained with a resolution of approximately 1.3–1.7 mm in the horizontally extreme fragments of the rosette. The resolution was 1.3 mm in the left fragment (beam incidence angle 80 degrees in the horizontal section) and 1.7 mm in the right fragment of the rosette near prism P6 (incidence angle 60 degrees). In the vertically extreme fragments of the rosette, the resolution was 1.4 to 1.6 mm, reaching 1.4 mm in the lower part near prism P5 (incidence angle 85 degrees in the vertical section) and 1.6 mm in the upper part of the rosette near prism P1 (incidence angle 65 degrees).
- From station 2 (left), point clouds were obtained with a resolution of approximately 1.4–2.4 mm in the horizontally extreme fragments of the rosette. The resolution was 1.4 mm in the left fragment (beam incidence angle 105 degrees in the horizontal section) and 2.4 mm in the right fragment near prism P6 (incidence angle 45 degrees). In the vertically extreme fragments of the rosette, the resolution ranged from 1.8 to 1.9 mm, reaching 1.8 mm in the lower part of the rosette near prism P5 (incidence angle 85 degrees) and 1.9 mm in the upper part of the rosette near prism P1 (incidence angle 65 degrees).
- From station 3 (right), point clouds were obtained with a resolution of approximately 1.2–1.7 mm in the horizontally extreme fragments of the rosette. The resolution was 1.7 mm in the left fragment (beam incidence angle 50 degrees in the horizontal section) and 1.2 mm in the right fragment near prism P6 (incidence angle 85 degrees). In the vertically extreme fragments of the rosette, the resolution ranged from 1.3 to 1.6 mm, reaching 1.3 mm in the lower part of the rosette near prism P5 (incidence angle 85 degrees) and 1.6 mm in the upper part of the rosette near prism P1 (incidence angle 65 degrees).

Preprocessing involved preliminary data preparation using the Fine Registration tool in CloudCompare, which automatically matches point clouds using the ICP algorithm. Iterative orientation of one cloud relative to another is achieved by minimizing the distance between corresponding points. Importantly, the reference cloud remains unchanged, while the matched cloud is transformed. Scan matching was performed on full-point clouds of the entire laboratory room, not just the section of the wall where the rosette is mounted.

2.4 Data analysis methodology

One of the most important stages in the point cloud classification process is the determination of a set of features that describe the geometry, structure, and radiometric properties of objects. In machine learning, the selection of the right features forms the foundation for effective model training.

In this research, the feature calculation stage for training classification models was based on the analysis of differences between corresponding points from two measurement series. These differences were determined between the reference series and the compared series.

To analyze the impact of cloud point characteristics on classification results, the machine learning algorithm was trained on four sets of input data. The following versions of feature sets for different training fields were prepared, containing data obtained from three stations during the simulation of individual measurement series:

- Data set 1: a complete set of input features for the training field, extracted from a single P2 prism, based on data obtained from the first station between series 0 and 1.
- Data set 2: a complete set of input features for the training field extracted for prisms P2, P5, P6, and P8, based on data obtained from all stations between series 0 and 1.
- Data set 3: full set of input features without density for the training field, extracted for prisms P2, P5, P6, and P8, based on data obtained from all stations between series 0 and 1.
- Data set 4: full set of input features for the training field, extracted for prisms P2, P5, P6, and P8, based on data obtained from all stations and series.

Table 2 presents the detailed characteristics of the input datasets used for binary classification. In the case of binary classification, class 1 was assigned to points that were displaced in the subsequent measurement series relative to the reference series, while class 0 was assigned to points that were not displaced in the subsequent measurement series relative to the reference series.

Table 3 presents the detailed characteristics of the input datasets used for binary classification. For multi-class classification, class 0 denotes points with no displacement in the subsequent measurement series relative to the reference series. Class 1 denotes points that were displaced by 4 mm in the subsequent measurement series

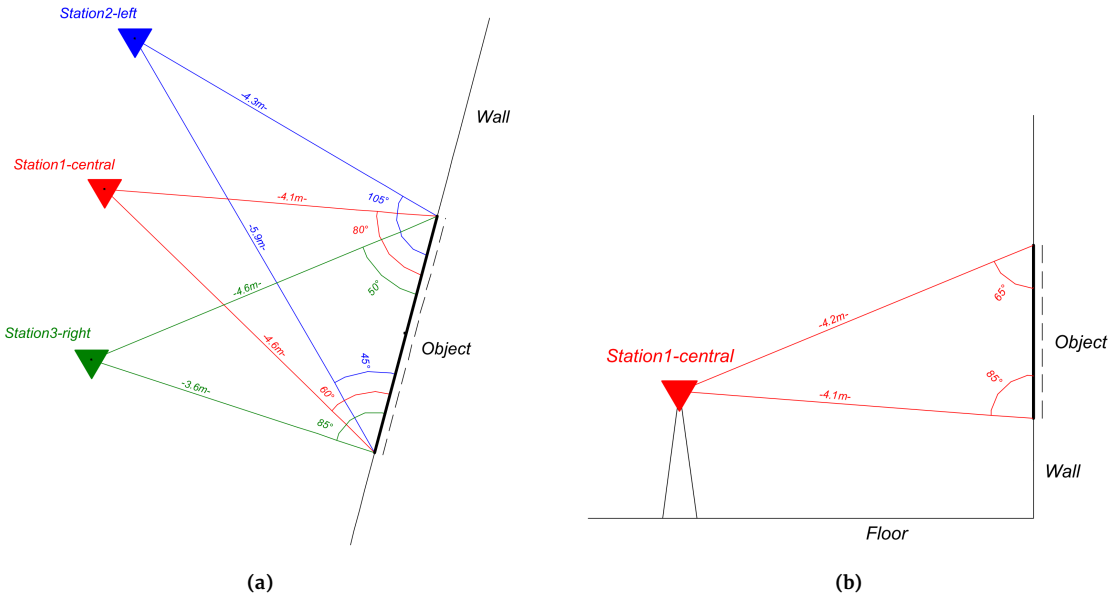


Figure 3. Schematic diagram of the measurement structure, location of TLS stations and the object during the experiment: (a) horizontal cross-section – recording of the object (rosette) from station 1 (center), station 2 (left), station 3 (right); (b) vertical cross-section – recording of the object (rosette) from station 1 (center)

Table 2. Parameters of four equal sets (Data set 1, Data set 2, Data set 3, and Data set 4) of input data used to train the machine learning algorithm to perform binary classification

Parameters of data set	Data set 1	Data set 2	Data set 3	Data set 4
Size of data set [points]	400	4358	4358	28209
Size of class 0 - No displacement [points]	291	3053	3053	14210
Size of class 1 - Displacement [points]	109	1305	1305	13999
Number of features	13	13	12	13

Table 3. Parameters of Data set 4 of input data used to train the machine learning algorithm to perform multi-class classification

Parameters of data set	Data set 4
Size of data set [points]	28209
Size of class 0 - No displacement [points]	14210
Size of class 1 - Displacement equal 4 mm [points]	1305
Size of class 2 - Displacement equal 9 mm [points]	5253
Size of class 3 - Displacement equal 13 mm [points]	7441
Number of features	13

relative to the reference series. Class 2 was assigned to points that shifted by 9 mm in the subsequent measurement series relative to the reference series. Finally, class 3 was assigned to points that moved by 13 mm in the subsequent measurement series relative to the reference series.

2.5 Point cloud classification algorithm

The collected data provide a basis for verifying the effectiveness of machine learning algorithms, particularly neural networks, in tasks related to the detection and classification of surface changes. This analysis focuses on detecting displacements in the plane of building walls, while the detection of displacements perpendicular to the plane of the walls is considered a potential direction for future research. This would require the use of more complex deep learning architectures and an expanded input database. This approach is unconventional, as standard displacement measurement methods rely on detecting horizontal, vertical, or three-dimensional displacements.

In order to detect displacements of building walls, a point cloud classification algorithm was developed using machine learning methods based on neural networks. The developed algorithm integrates the stages of feature extraction, model training, and their application to spatial data classification, forming a complete displacement detection procedure. A block diagram of the procedure is shown in Figure 4.

The procedure consists of seven main stages. The source data consisted of point clouds recorded in successive measurement periods, with each point described by geometric coordinates (XYZ), color (RGB), and reflection intensity (I). In the first stage, the data underwent preliminary processing, including noise reduction and alignment with the reference cloud.

At the stage of extracting geometric and radiometric features for each point, a set of features describing its local properties was calculated. In addition, point surface curvature was computed to distinguish between fixed and displaced areas.

Based on data from individual series, training sets were created, including samples of points with a known class. Two classification scenarios were considered:

- binary classification (displaced/non-displaced points),
- multi-class classification (four levels of displacement).

The training sets were used to train classification models in MATLAB. The training process included:

- training models based on prepared features,
- cross-validation to assess prediction quality,
- feature selection to identify the parameters with the greatest diagnostic significance.

The final models were exported for further analysis. The trained models were used to classify all points in the clouds of subsequent series. The result of the classification was the assignment of a class

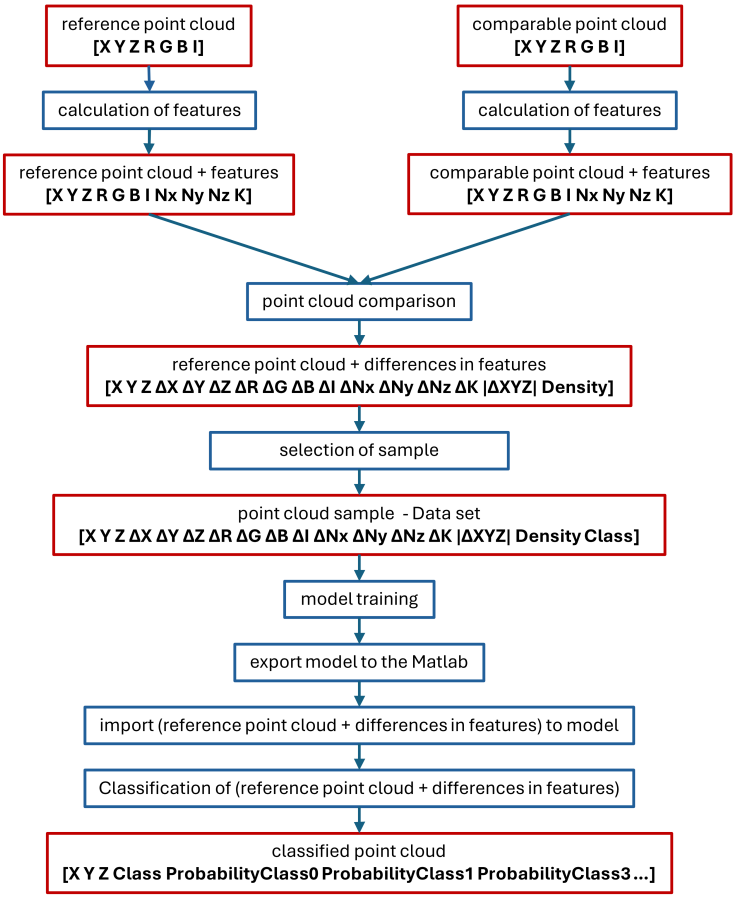


Figure 4. An algorithm for detecting displacements occurring in the plane of building walls, with a presentation of changes in the feature matrix in subsequent stages

label to each point and the estimation of the probability of belonging to a given class.

The classification results were imported into the CloudCompare environment, which enabled their visualization by coloring according to classes or probabilities. Further analysis consisted of examining changes over time and assessing the spatial distribution of displacements.

2.6 Mathematical references of the algorithm

The developed methodology uses machine learning algorithms based on artificial neural networks. The basic building blocks of such networks are neurons, which enable the mapping of nonlinear relationships between input and output variables (Haykin, 2009; Heaton, 2017). The key parameters that define the architecture of the network are:

- The number of hidden layers – the network can contain no hidden layers (we are then talking about a single perceptron), have one layer, or multiple layers.
- The number of neurons in the layers, which directly affects the ability of the network.
- The method of connections between neurons, which can be fully connected (i.e., every neuron to every neuron) or partially connected.

The internal logic of a single neuron operation can be broken down into three steps. The first step is calculating the weighted sum of the inputs. Each input value is assigned a weight by which it is multiplied and then summed. An offset value (bias) is added to this sum, which allows the activation function to be shifted and increases the flexibility of the model:

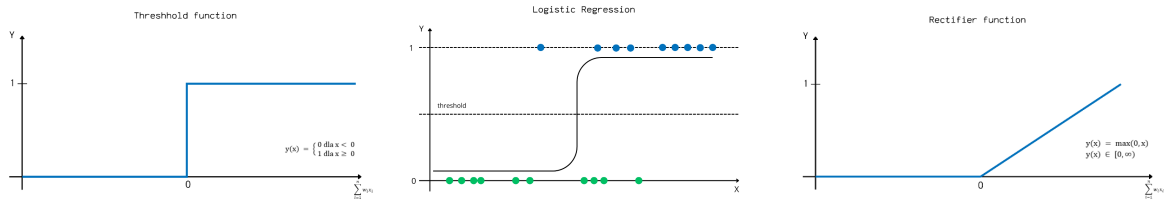
$$\sum_{i=1}^{n} w_i x_i + b, \tag{1}$$

where x_i – the value of the i-th input, w_i – the weight assigned to the i-th input, b – bias, the offset value.

The second stage of a single neuron is to pass the output through an activation function. Next, the output value of the neuron is passed on to subsequent neurons in the next layers of the network.

The activation function is a mathematical function that introduces nonlinearity into the model. It determines whether a neuron activates or fails to do so, that is, if the signal is transmitted. There are many examples of functions in the literature (Luo et al., 2023; Liu et al., 2019; Cennamo et al., 2021). Among the most popular are the threshold function (Figure 5a), sigmoid function (Figure 5b), and rectifier function (ReLU) (Figure 5c).

The research described in this article used a feedforward neural network. In such a network, the first layer of the neural network is connected to the network input, and each subsequent layer is connected to the previous layer. Each layer multiplies the input data by a weight matrix and then adds a bias vector, and then it is followed by an activation function. After the last layer, the network output and its posterior probabilities are generated. Standard neural network hyperparameter values were used, namely: 1 fully connected layer in the neural network, excluding the last fully connected layer for classification, a fully connected layer size of 10 neurons, ReLU activation function, no regularization, no standardization within the model, and the stopping criterion is 1000 iterations. However,



(a) (b) (c)

Figure 5. (a) Threshold function, (b) Sigmoid function, (c) ReLU rectifier function (Wojda, 2025)

in order to reliably estimate the effectiveness of the model and its generalization, 5-fold cross-validation was used during the analyses.

A set of geometric and radiometric features, determined on the basis of matched point clouds from different measurement periods, was used to describe and detect displacements. The methodology for calculating the attributes of the input data set was based on determining the differences in characteristics corresponding to points from two measurements: the reference and the compared. The following set of characteristics with a mathematical basis was proposed:

- changes in coordinates: ΔX, ΔY, ΔZ,
- changes in color: ΔR, ΔG, ΔB,
- change in intensity: ΔI,
- changes in normal vectors: ΔNx, ΔNy, ΔNz,
- changes in local curvature:

$$\Delta K = K_{\text{ref}} - K_{\text{target}}, \tag{2}$$

where $K = \frac{\lambda_1}{\lambda_1+\lambda_2+\lambda_3}$, $\lambda_1 \geq \lambda_2 \geq \lambda_3$ are eigenvalues of covariance matrices, K_{ref} is the curvature value of a point in the reference cloud, K_{target} – curvature value of a point from a fitted cloud,

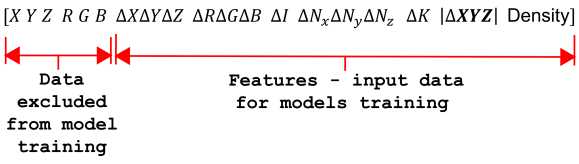
- displacement vector length: |ΔXYZ|,
- point density:

$$\text{Density}_i = \left| \left\{ P_j \in (\text{ref} \cup \text{target}) : \; |P_i - P_j| \leq r \right\} \right|, \tag{3}$$

where P_i – reference point for which we calculate density, P_j – potentially neighboring point, ref ∪target – point cloud created by combining point clouds recorded during two measurement epochs, r – sample radius.

The script includes information about the local density of points, with two point clouds combined into a single set. Then, for each point from the reference epoch, the number of neighboring points in its vicinity was determined. For this purpose, a function was used in which the radius r was defined as equal to the size of the deformation under study, i.e., the change in area, specifically 4 mm, 9 mm, and 13 mm. This parameter was selected individually for each pair of epochs. The radius must be defined in advance as one of the main parameters of the proprietary compute_features function (used to generate additional point features). It must be admitted that the proposed definition of proximity is a limitation of the developed methodology, since it assumes knowledge of the approximate magnitude of displacements.

The result of the developed script is a matrix containing a set of features, which is shown in Figure 6a. An example fragment of such a file is shown in Figure 6b. It should be noted that the presented set of features was developed on the basis of a series of tests, numerical analysis of feature rankings, and graphical analysis of the relationships between feature pairs.



(a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	3.6815	-3.4098	1.7651	164	165	159	4.3582e-04	-4.5228e-04	2.2459e-04	1	1	1	0.0196	0.0577	0.1921	-0.0693	0.0571	6.6704e-04	518
2	3.6812	-3.4096	1.7634	164	165	159	-1.1516e-04	2.5750e-05	-5.4800e-06	1	1	3	0.0235	0.0040	0.0563	-0.1336	0.0782	1.1813e-04	579
3	3.6808	-3.4093	1.7617	164	165	159	6.4610e-05	-9.2750e-05	1.3459e-04	1	1	3	0.0235	-0.0184	-0.0743	0.0588	0.0983	1.7576e-04	646
4	3.6809	-3.4094	1.7601	164	165	159	-1.0538e-04	1.6690e-05	-2.5390e-05	1	1	3	0.0392	0.0862	0.2881	0.0221	0.0651	1.0967e-04	692
5	3.6810	-3.4094	1.7586	164	165	159	-1.2517e-04	2.5750e-05	1.4540e-05	1	1	3	0.0314	0.0506	0.1872	-0.0430	0.0608	1.2862e-04	735

(b)

Figure 6. (a) The set of attributes/features contained in each row (corresponding to a single point in the cloud) of the matrix used to train the models. (b) A fragment of an example result matrix, showing the full set of calculated features for a pair of point clouds.

3 Results

In order to analyze the results of binary and multi-class classification, the following research strategy was adopted. First, statistical metrics were calculated based on error matrices and evaluated. Next, the feature rankings generated using the MRMR and ReliefF algorithms were analyzed. To enable comparison of these two algorithms, the coefficients were standardized. Finally, the classification results of all clouds were evaluated numerically and visually.

3.1 Binary classification analysis

The binary analysis consisted of finding a set of point cloud features that would allow for the most accurate identification of displaced prism fragments. The same neural network model was trained using all four datasets for testing purposes. The trained models were then used for binary classification of all nine measured point clouds. In this classification, the model needed to detect the change rather than determine its magnitude. During model training, 5-fold cross-validation and additional standardization of input data sets were used. The data set was divided into five parts, four of which were used to train the model and one to validate it. This process was repeated five times, each time using a different validation set.

Analysis of statistical factors

The statistical indicators obtained after training the algorithm based on artificial neural networks achieved remarkably high values, with an average classification accuracy of 99.1. On the other hand, it is worth noting that accuracy alone can be misleading, especially with unbalanced data sets. Kappa (Cohen's Kappa agreement coefficient) was equal to 1, which means that the classifier was in full agreement with the reference classification, exceeding random agreement. The Precision coefficient was close to 1. This means that most of the points classified as displaced actually be-

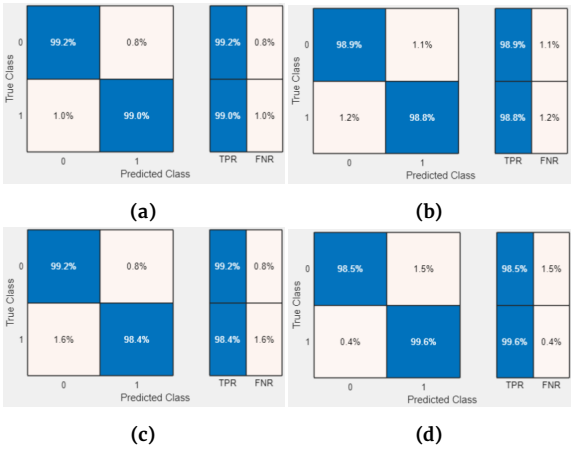


Figure 7. Confusion matrices for binary classification: (a) Neural Network model (described in Section 2.6); (b) Tree model (with maximum number of splits 100); (c) KNN model (number of neighbors is 10); (d) Gaussian SVM model (with kernel scale 0.9). These results were obtained for Data set 4.

longed to this class – there was a low risk of false positives. The Recall coefficient (sensitivity/detectability) reached a value of 0.99, meaning that the neural networks effectively detected almost all cases of displacement, with a low risk of undetected displacement (false negatives). The F1-score hovered around 0.99, indicating an excellent balance between precision and sensitivity. In theory, this means that the model performs well both in detecting displacements and in avoiding misclassifications.

The statistical coefficients described above were obtained for a neural network model trained on Data set 2 and Data set 4. It should be noted that several machine learning models were investigated during data analysis and algorithm development. For example, Figure 7 shows confusion matrices obtained after training four models using the same Data set 4: (a) Neural Network (described in Section 2.5), (b) Tree (with maximum number of splits 100), (c) KNN (number of neighbors is 10), and (d) Gaussian SVM (with kernel scale 0.9). Despite the approximate percentage numbers of point groups, it can be concluded that the results of neural network training are superior. Table 4 and Table 5, in turn, present more precise accuracy statistics for the results of training models on Data sets 2 and Data set 4, respectively. In addition to neural network statistics, statistics for Tree, KNN, and Gaussian SVM models are also presented. Comparing the results obtained in both variants, it can be concluded that the issue addressed is valid. The size and diversity of training fields affect the classification process. As a result of training on a smaller dataset, better accuracy indicators were achieved, including Accuracy, Precision, Recall, and F1. An insufficiently large training dataset resulted in a higher risk of overfitting and underestimated detection effectiveness for new or characteristic data in later stages of the study. However, after increasing the size and diversity of the training fields, more reliable results were obtained that were resistant to classification errors for classes with fewer cases. Based on the results obtained, a model based on artificial neural networks was selected for further analysis. These algorithms model nonlinear relationships between data and work well with large data sets. Additionally, they are universal and flexible, as they are highly effective in both binary and multi-class classification tasks. Considering previous observations, it was decided to use larger training fields.

The above statistical parameters (Accuracy, Precision, Recall, and F1) are determined only by the structure of the test fields. Therefore, it can be concluded that they represent the a priori accuracy of the machine learning model.

Feature ranking

As part of the analysis of the impact of features, two rankings were compared: the first one generated using the MRMR method, and the second one generated using the ReliefF method. The standardized results for each variant are summarized in Table 6.

MRMR rated the following features highest: displacement vector length, changes in local curvature, point density, and changes in normal vectors. This means that these features provided the model with the most relevant data for training. In ReliefF, on the other hand, the highest places were taken by the normal vector components: ΔNz, ΔNx, and ΔNy. It was therefore demonstrated that, on a local scale, these features are key to the model. The density feature was at the bottom of the ReliefF ranking, i.e., it proved to be the least important for the algorithm, which may be due to the greater spatial and radiometric diversity of the data.

Based on a graphical comparison of the relationships between pairs of features (as shown in Figure 8), it was found that the strongest correlations exist between density (column_19 in figure (a)) and displacement vector length (column_18 in figure (a)); as well as between density (column_19 in figure (b)) and local curvature change (column_17 in figure (b)). The graphs show the points from Data set 4, which was used to train the neural network for binary classification. The relationship between two features is stronger if the points do not have a random distribution, but instead form regular clusters/groups of points belonging to the same class.

Analysis of results

The analysis of binary classification results was performed for four training field variants, as shown in Table 7. The first variant, based solely on data from station 1, confirmed high prediction accuracy, but the model had limited ability to detect deformed points, especially with larger displacements (9 and 13 mm). The share of class 1 in pairs of epochs 1–2 and 0–2 was 0.15% and 0.11%, respectively.

Expansion of training fields in the second variant: improved the results for data from station 2 (the share of class 0 increased to 90.18%), but at the same time worsened the classification for station 3 (the share of class 0 fell to 74.54% and 84.13% for displacements of 9 mm and 13 mm). The removal of the density feature in variant three resulted in a significant improvement in the results for station 2 (the share of class 0 increased to 96.85%), but in other cases, a decrease in classification confidence and a higher percentage of points incorrectly assigned to class 1 were observed. The best results were obtained in variant four. The use of training fields covering all sites and epochs allowed for the highest effectiveness to be achieved. The share of class 0 exceeded 99%, and the average classification probabilities were close to 1. In all analyses, the weakest results concerned data from site 2.

Visualization of results

Four variants of binary point cloud classification were evaluated as part of the study. The analysis aimed to determine the impact of training set selection on the effectiveness of rosette prism displacement detection.

The results obtained using the model trained on Data set 1 indicated correct identification of most areas displaced by 4 mm, but the model was unable to fully detect changes of 9 and 13 mm. The repeatability of deformation detection was observed regardless of its actual extent. The best results were obtained for station 1, while the data from stations 2 and 3 contained numerous classification errors. The potential causes of these errors are the measurement geometry and the variable density of point clouds.

Data set 2 included training data from all stations, which allowed for a greater variety of recordings to be included. The results of the classification performed on the basis of Data set 2 are presented in Figure 9. Overall, it showed better results than Data set 1. Nevertheless, incorrect identifications of displacements were still observed, especially in the case of extreme stations.

Disabling the point density feature in Data set 3 resulted in a re-

Table 4. Evaluation of the classification accuracy of each class (Precision, Recall, F1) in binary classification. The table contains statistics for the proposed neural network model (described in Section 2.6), as well as for comparison with the following models: tree (with a maximum number of divisions of 100), KNN (number of neighbors is 10), and Gaussian SVM (with kernel scale 0.9). These results were obtained for Data set 2.

Class	TP	FP	FN	Accuracy	Precision	Recall	F1	p0	pe	kappa
Neural Network (described in section 2.6.)										
0	3050	4	3	-	1.00	1.00	1.00	-	0.49	-
1	1301	3	4	-	1.00	1.00	1.00	-	0.09	-
All	4351	7	7	1.00	-	-	-	1.00	0.58	1.00
Tree (with maximum number of splits 100)										
0	3049	0	4	-	1.00	1.00	1.00	-	0.49	-
1	1305	4	0	-	1.00	1.00	1.00	-	0.09	-
All	4354	4	4	1.00	-	-	-	1.00	0.58	1.00
KNN (number of neighbors is 10)										
0	3041	31	12	-	0.99	1.00	0.99	-	0.50	-
1	1268	12	31	-	0.99	0.98	0.98	-	0.09	-
All	4309	43	43	0.99	-	-	-	0.99	0.58	0.98
Gaussian SVM (with kernel scale 0.9)										
0	2964	0	89	-	1.00	0.97	0.99	-	0.48	-
1	1305	89	0	-	0.94	1.00	0.97	-	0.10	-
All	4269	89	89	0.98	-	-	-	0.98	0.57	0.95

Table 5. Evaluation of the classification accuracy of each class Precision, Recall, F1) in binary classification. The table contains statistics for the proposed neural network model (described in Section 2.6), as well as for comparison with the following models: tree (with a maximum number of divisions of 100), KNN (number of neighbors is 10), and Gaussian SVM (with kernel scale 0.9). These results were obtained for Data set 4.

Class	TP	FP	FN	Accuracy	Precision	Recall	F1	p0	pe	kappa
Neural Network (described in section 2.6.)										
0	14095	134	115	-	0.99	0.99	0.99	-	0.25	-
1	13865	115	134	-	0.99	0.99	0.99	-	0.25	-
All	27960	249	249	0.99	-	-	-	0.99	0.50	0.98
Tree (with maximum number of splits 100)										
0	14047	166	163	-	0.99	0.99	0.99	-	0.25	-
1	13833	163	166	-	0.99	0.99	0.99	-	0.25	-
All	27880	329	329	0.99	-	-	-	0.99	0.50	0.98
KNN (number of neighbors is 10)										
0	14102	221	108	-	0.98	0.99	0.99	-	0.26	-
1	13778	108	221	-	0.99	0.98	0.99	-	0.24	-
All	27880	329	329	0.99	-	-	-	0.99	0.50	0.98
Gaussian SVM (with kernel scale 0.9)										
0	13992	59	218	-	1.00	0.98	0.99	-	0.25	-
1	13940	218	59	-	0.98	1.00	0.99	-	0.25	-
All	27932	277	277	0.99	-	-	-	0.99	0.50	0.98

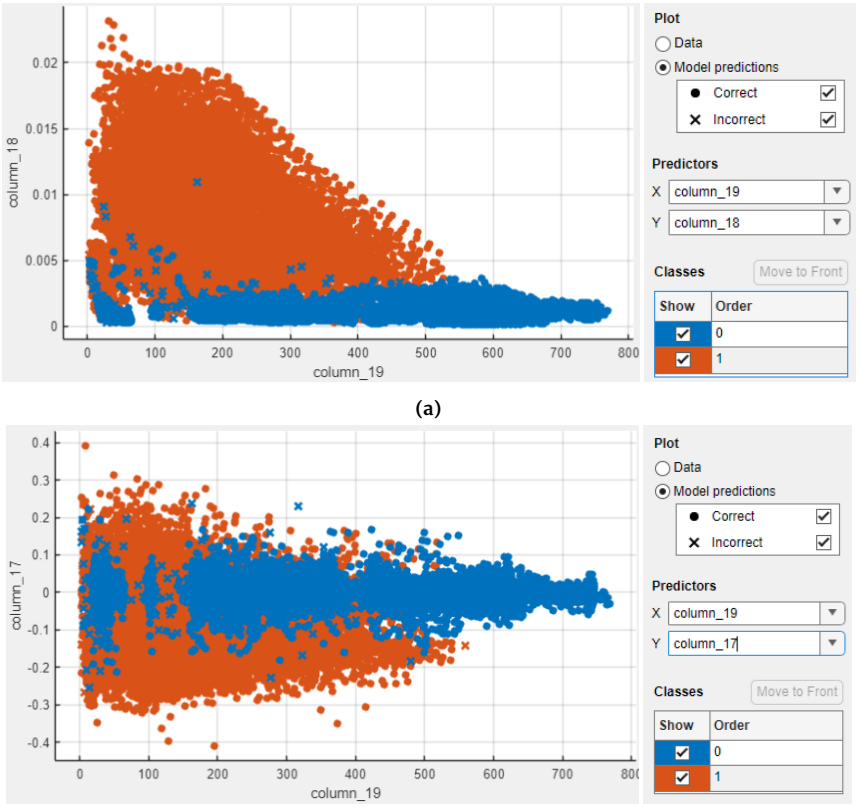


Figure 8. Graphical representation of the relationship between pairs of features: (a) density (column_19) and displacement vector length (column_18); (b) density (column_19) and changes in local curvature (column_17) for binary classification

Table 6. Ranking of features obtained using MRMR and ReliefF algorithms for Data sets 4 – two classes

Feature	Standardized MRMR score	Standardized ReliefF score
displacement vector length: |ΔXYZ|	1.0	0.4
changes in local curvature: ΔK	0.8	0.3
point density: Density	0.7	0.2
changes in normal vectors: ΔNz	**0.8**	**1.0**
change in intensity: ΔI	0.8	0.5
changes in normal vectors: ΔNx	0.6	0.8
changes in coordinates: ΔZ	0.7	0.4
changes in coordinates: ΔY	0.6	0.2
changes in color: ΔG	0.6	0.4
changes in normal vectors: ΔNy	0.7	0.6
changes in coordinates: ΔX	0.3	0.4
changes in color: ΔB	0.3	0.3
changes in color: ΔR	0.2	0.2

duction in surface classification errors and improved classification of the prisms themselves. However, local point errors still occurred. The best binary classification results were obtained after training neural networks using Data set 4. The classification results for all point cloud pairs are presented in Figure 10.

The complete Data set 4, which contained data recorded from all stations and for all simulated displacements, proved to be the most effective. The model correctly classified the prisms on the rosette. The number of misclassified points was the lowest among all the variants analyzed. The few errors were mainly due to differences in registration density.

The results of the experiment demonstrate that the quality and diversity of the training set significantly influence the effectiveness of deformation classification using neural networks. Models based on a wide range of data enable more accurate detection of changes. Limitations resulting from the geometry of the measurement and the non-uniformity of point cloud density require optimization of computational methods.

The parameters Accuracy, Precision, Recall, and F1 are reliable when comparing models trained using the same set of features and test fields. For example, based on the results in Table 4, it can be concluded that the Neural Network model trained on Data set 2 is better than the Gaussian SVM model trained on Data set 2. Similarly, based on the results presented in Table 5, it can be concluded that the Neural Network model trained on Data set 4 has an advantage over the Gaussian SVM model trained on Data set 4. As shown by the results of the classification of the cloud representing the entire rosette, the size of the test field and the selection of learning features are of fundamental importance for the effectiveness of the model. For the Neural Network model trained on Data set 2, the Accuracy, Precision, Recall, and F1 parameters were better than for the Neural Network model trained on Data set 4. However, the results of classifying the cloud representing the entire rosette using

Table 7. Summary of binary classification results, including the percentage share of points assigned to individual classes (0,1): Data set 1 – training field from P2 prism, data from the first station between series 0 and 1; Data set 2 – training field for P2, P5, P6, and P8, data from all stations between series 0 and 1; Data set 3 – training field for P2, P5, P6, and P8, data from all stations between series 0 and 1; Data set 4 – training field for P2, P5, P6, and P8, data from all stations and series.

Simulated displacement [mm]	**Class 0** [%]			**Class 1** [%]		
	TLS station 2 left	**TLS station 1 central**	**TLS station 3 right**	**TLS station 2 left**	**TLS station 1 central**	**TLS station 3 right**
			Data set 1			
4	72.99	99.42	98.45	27.01	0.58	1.55
9	99.69	99.85	99.85	0.31	0.15	0.15
13	99.78	99.89	99.91	0.22	0.11	0.09
			Data set 2			
4	90.18	99.48	99.08	9.82	0.52	0.92
9	99.72	99.89	74.54	0.28	0.11	25.46
13	99.83	99.94	84.13	0.17	0.06	15.87
			Data set 3			
4	96.85	99.19	98.27	3.15	0.81	1.73
9	97.78	97.76	98.21	2.22	2.24	1.79
13	97.37	98.69	98.40	2.63	1.31	1.60
			Data set 4			
4	95.71	99.36	99.03	4.29	0.64	0.97
9	99.21	99.35	99.31	0.79	0.65	0.69
13	99.15	98.14	99.21	0.85	1.86	0.79

the Neural Network classifier trained on Data set 4 outperform those of classifying the cloud representing the entire rosette using the Neural Network classifier trained on Data set 2.

To visualize the classification results, Figure 11 shows a fragment of the point cloud covering prism P1. These are the results of binary classification obtained using a neural network model trained on Data set 4. This prism was located above the scanner, which meant that the laser beam struck the surface of the object at an oblique (non-perpendicular) angle. Points classified as class 1 (simulated displacements) are marked in red, while points without displacements are marked in blue.

In contrast, Figure 12 presents analogous classification results for prism P9, which was located at the same height as the scanner. In this case, the laser beam fell on the surface of the object at a right angle (approximately 90 degrees). Both tables show differences in classification effectiveness, resulting from the measurement geometry (angle of incidence of the beam) and the position of the TLS stations relative to the tested fragment of the object, among other things.

It should be noted that the P1 prism was not moved in each measurement series. It was the highest prism in the entire rosette, located on the micrometric table. The fixed prisms, permanently attached to the wall, were constructed differently from the movable prisms located on the micrometric tables. To standardize the experimental conditions, the analyses focused on prisms of the same construction. Therefore, the tables present the results of the classification of a point cloud fragment representing a prism on a micrometric table in two variants: in Figure 11 – non-displaced, and in Figure 12 – displaced.

As a reminder, the position of prism P1 was changed by only 4 mm during the simulation of one measurement epoch from the second station. The effect of this displacement can be observed by analyzing the first fragment of the point cloud in Figure 11 (first row, first figure) with the remaining fragments of the point cloud. The detected points displaced on prism P1 may indicate the influence of the station position relative to the measured object. In the case of this fragment of the rosette, there may be a problem with the laser beam slipping on the edges of the prism housing, which may result in classification errors.

To confirm the above statements, Table 8 and Table 9 present a summary of the binary classification results. They contain the percentage share of points assigned to classes (0,1) and the average probability of class prediction for point cloud fragments, which are presented in Figures 11 and 12, i.e., prism P1 and its surroundings, and prism P9 and its surroundings. The classifier was a neural network model trained on Data set 4 – test field for P2, P5, P6, P8, data from all stations and series. Statistical data of point cloud fragments representing prism P1 confirm its displacement by 4 mm during scanning from station 2-left.

For the results obtained for point clouds representing prism P1, it should be noted that, firstly, the percentage of points with class 0 prediction increases with growing displacement, while the percentage of points with class 1 prediction decreases. Secondly, the average probability of class 0 prediction rises with increasing displacement, as does the average probability of class 1 prediction. These trends are consistent across all stations.

For the results obtained for point clouds representing the P9 prism, it should be noted that, firstly, the percentage of points with a class 0 prediction decreases with increasing displacement, while the percentage of points with a class 1 prediction rises. Secondly, both the average probability of class 0 and class 1 prediction increase with growing displacement. The same patterns are observed for all stations.

3.2 Multi-class classification analysis

The multi-class analysis involved identifying a set of point cloud features that would enable the most accurate sizing of the displaced prism fragments. The neural network model was trained using Data set 4. The trained model was then used to classify all nine measured point clouds. For this classification, the model needed to detect the change and accurately determine its magnitude. Therefore, four classes were defined at the outset: 0 mm simulated displacement (blue color), 4 mm simulated displacement (green color), 9 mm simulated displacement (yellow color), and 13 mm simulated displacement (red color). During model training, 5-fold cross-validation and additional standardization of input data sets were used.

Analysis of statistical factors

The statistical parameters obtained after the machine learning procedure show that neural networks are a reliable tool for detecting displacements, even when classified into multiple categories. It

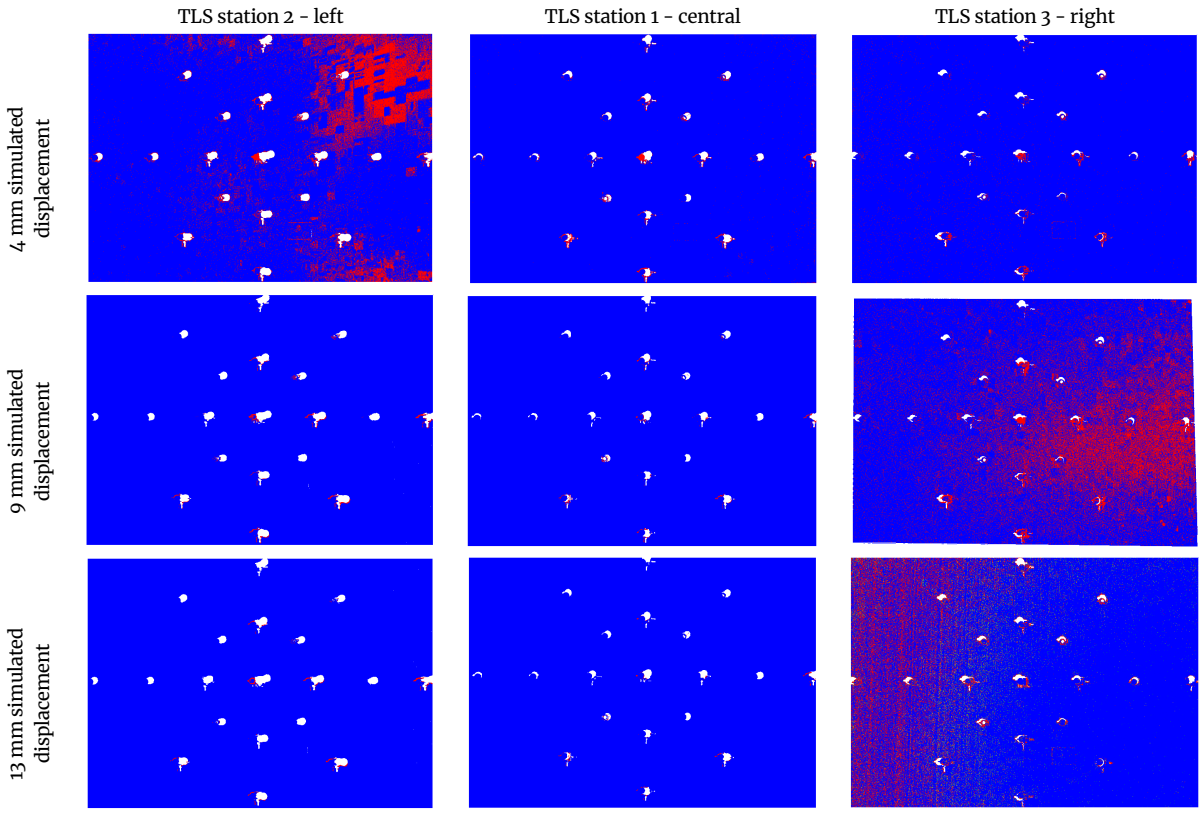


Figure 9. Visualization of binary classification results obtained using a neural network model trained on input Data set 2 – full rosette: simulated displacements are red color – class 1, no simulated displacements are blue color – class 0

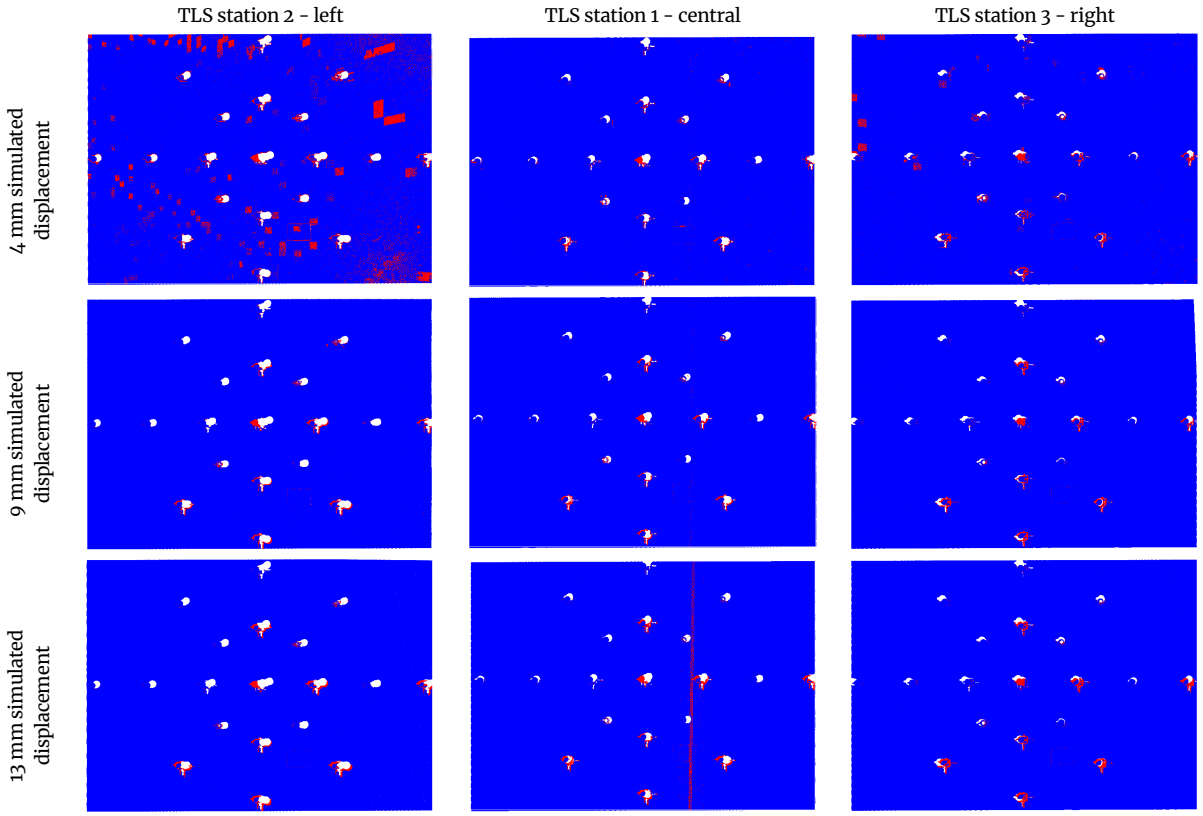


Figure 10. Visualization of binary classification results obtained using a neural network model trained on input Data set 4 – full rosette: simulated displacements are red color – class 1, no simulated displacements are blue color – class 0

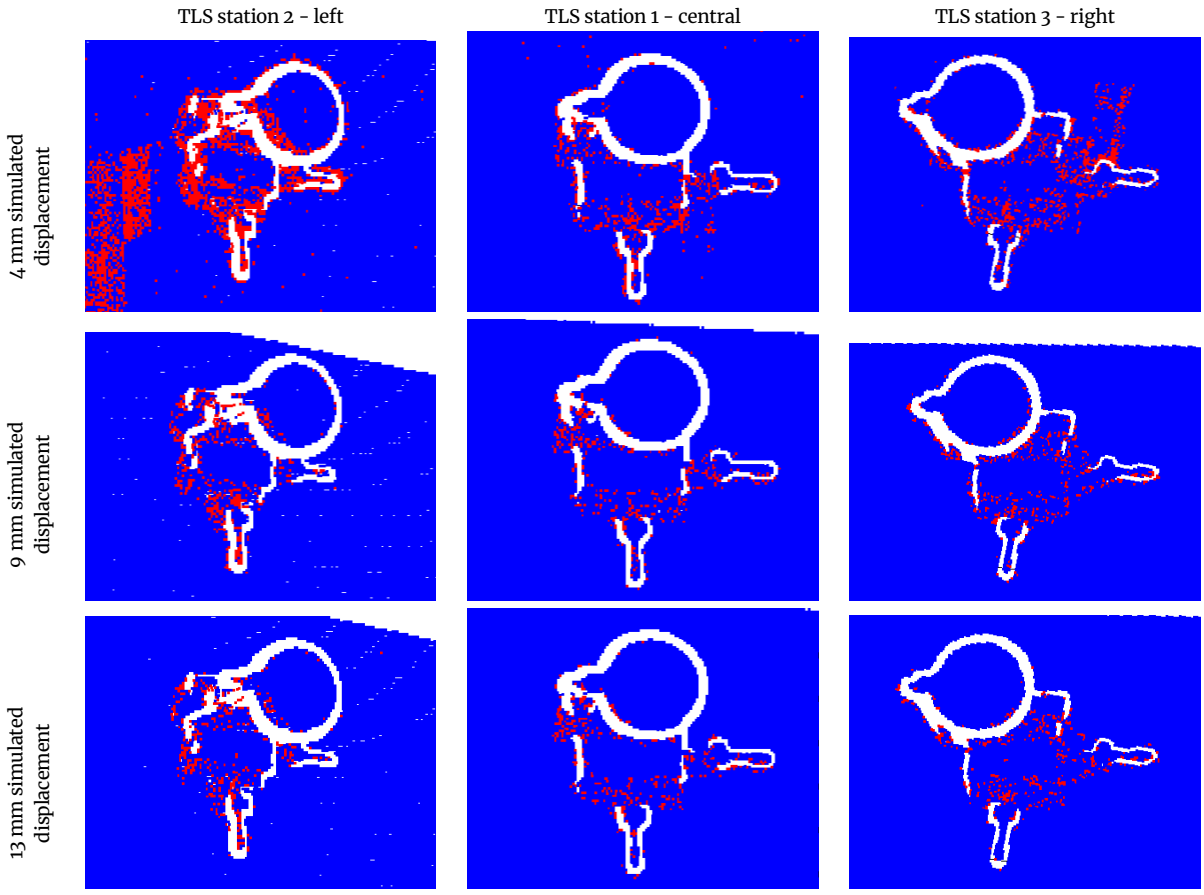


Figure 11. Visualization of binary classification results obtained using a neural network model trained on input Data set 4 – zoom on prism P1, which was located above the scanner: simulated displacements are red color – class 1, no simulated displacements are blue color – class 0

Table 8. Summary of binary classification results, including the percentage share of points assigned to classes (0.1) and the average probability of class prediction: classifier – neural network trained on Data set 4 – test field for P2, P5, P6, and P8, data from all stations and series. Statistical data of point cloud fragments representing prism P1, shown in Figure 11.

		TLS station 2 - left		TLS station 1 - central		TLS station 3 - right	
		part of cloud predicted for class [%]	average class probability	part of cloud predicted for class [%]	average class probability	part of cloud predicted for class [%]	average class probability
4 mm displacement	class 0	85.14	0.93	97.29	0.99	96.23	0.98
	class 1	14.86	0.83	2.71	0.86	3.77	0.84
9 mm displacement	class 0	96.68	1.00	98.20	1.00	97.91	1.00
	class 1	3.32	0.94	1.80	0.96	2.09	0.95
13 mm displacement	class 0	97.40	1.00	98.45	1.00	98.25	1.00
	class 1	2.60	0.96	1.55	0.97	1.75	0.97

Table 9. Summary of binary classification results, including the percentage share of points assigned to classes (0.1) and the average probability of class prediction: classifier – neural network trained on Data set 4 – test field for P2, P5, P6, P8, data from all stations and series. Statistical data of point cloud fragments representing prism P9, shown in Figure 12.

		TLS station 2 - left		TLS station 1 - central		TLS station 3 - right	
		part of cloud predicted for class [%]	average class probability	part of cloud predicted for class [%]	average class probability	part of cloud predicted for class [%]	average class probability
4 mm displacement	class 0	88.62	0.91	92.76	0.99	93.54	0.99
	class 1	11.38	0.86	7.24	0.93	6.46	0.95
9 mm displacement	class 0	86.98	0.99	91.56	1.00	91.80	1.00
	class 1	13.02	0.98	8.44	0.98	8.20	0.98
13 mm displacement	class 0	83.91	1.00	89.80	1.00	89.66	1.00
	class 1	16.09	0.99	10.20	0.99	10.34	0.99

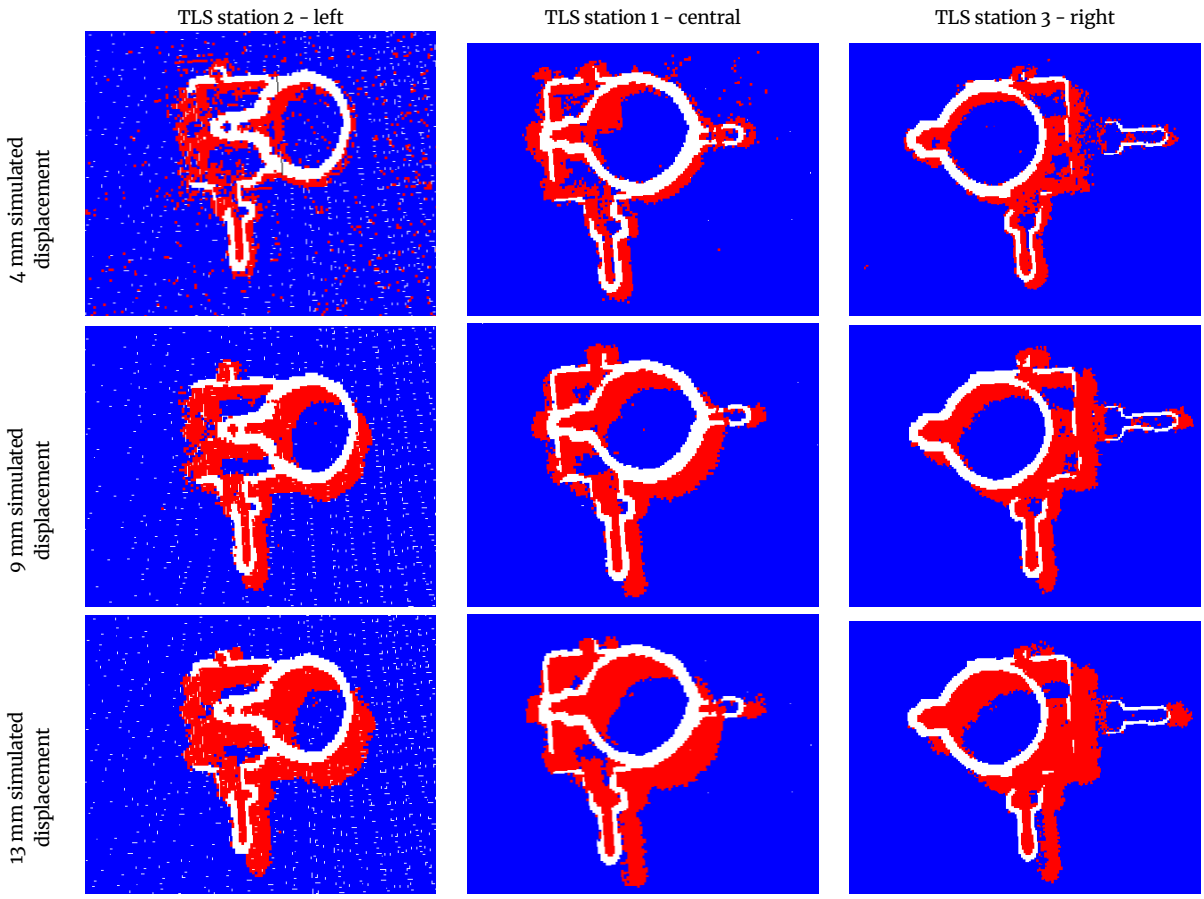


Figure 12. Visualization of binary classification results obtained using a neural network model trained on input Data set 4 – zoom on prism P9, which is located at the same height as the scanner (approximately perpendicular angle of incidence of the laser beam on the wall surface): simulated displacements are red color – class 1, no simulated displacements are blue color – class 0

was particularly important to maintain a high balance between precision and sensitivity, which reduces the risk of errors. The classification accuracy was 96.0%, indicating that most points were correctly assigned to their respective classes. A Kappa coefficient of 0.94 indicates standard classification agreement with the reference data. A Precision coefficient of 0.94 means that the model generates few false positives. The Recall coefficient of 0.94 indicates a low percentage of false negatives. The F1-score coefficient of 0.95 shows that the model performs well in terms of both classification accuracy and completeness of detection of all cases.

The confusion matrix for the multi-class classification neural network model described in Section 2.6 is shown in Figure 13a. Confusion matrices for simpler classifiers on the same feature sets are also presented to prove that the neural network was indeed necessary. For comparison, the relationships between the true and predicted classes are shown for Figure 13b the Tree model (with a maximum number of splits of 100), Figure 13c the KNN model (with 10 neighbors), and Figure 13d the Gaussian SVM model (with a kernel scale of 0.9).

Table 10 shows performance measures for each class (Precision, Recall, F1) and balanced accuracy or macro-averaged F1 for the entire dataset in multi-class classification. The results are presented for the target neural network model, as well as for comparison with the following models: Tree (with a maximum number of splits of 100), KNN (with 10 neighbors), and Gaussian SVM (with a kernel scale of 0.9). The best reliability measures were obtained for the neural network model described in Section 2.6.

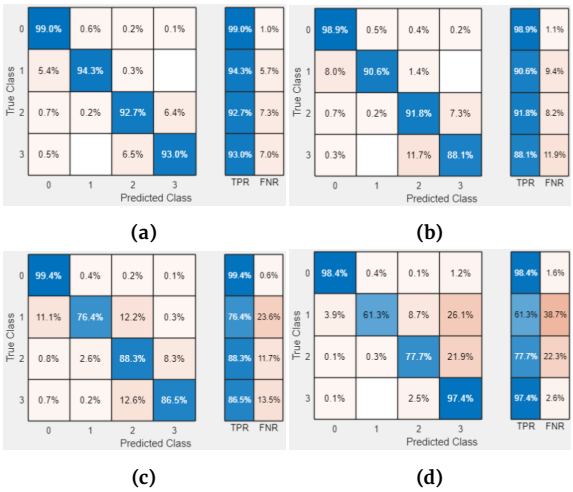


Figure 13. Confusion matrices for multi-class classification: (a) Neural Network model (described in Section 2.6); (b) Tree model (with maximum number of splits 100); (c) KNN model (number of neighbors is 10); (d) Gaussian SVM model (with kernel scale 0.9). These results were obtained for Data set 4.

Table 10. Evaluation of the classification accuracy of each class (Precision, Recall, F1), as well as balanced accuracy and average macro F1 in multi-class classification. The table contains statistics for the proposed Neural Network model (described in Section 2.6), as well as for comparison with the following models: Tree (with a maximum number of splits 100), KNN (number of neighbors is 10), and Gaussian SVM (with kernel scale 0.9). These results were obtained for Data set 4.

Class	TP	FP	FN	Precision	Recall	F1	Macro-averaged F1	Balanced Accuracy	p0	pe	kappa
Neural Network (described in section 2.6.)											
0–0 mm	14073	146	137	0.99	0.99	0.99	-	-	-	0.25	-
1–4 mm	1230	99	75	0.93	0.94	0.93	-	-	-	0.00	-
2–9 mm	4867	516	386	0.90	0.93	0.92	-	-	-	0.04	-
3–13 mm	6920	358	521	0.95	0.93	0.94	-	-	-	0.07	-
All	27090	1119	1119	-	-	-	**0.95**	**0.94**	0.96	0.36	0.94
Tree (with maximum number of splits 100)											
0–0 mm	14053	161	157	0.99	0.99	0.99	-	-	-	0.25	-
1–4 mm	1182	85	123	0.93	0.91	0.92	-	-	-	0.00	-
2–9 mm	4824	939	429	0.84	0.92	0.88	-	-	-	0.04	-
3–13 mm	6552	413	889	0.94	0.88	0.91	-	-	-	0.07	-
All	26611	1598	1598	-	-	-	**0.92**	**0.92**	0.94	0.36	0.91
KNN (number of neighbors is 10)											
0–0 mm	14118	240	92	0.98	0.99	0.99	-	-	-	0.26	-
1–4 mm	997	207	308	0.83	0.76	0.79	-	-	-	0.00	-
2–9 mm	4640	1123	613	0.81	0.88	0.84	-	-	-	0.04	-
3–13 mm	6437	447	1004	0.94	0.87	0.90	-	-	-	0.06	-
All	26192	2017	2017	-	-	-	**0.88**	**0.88**	0.93	0.36	0.89
Gaussian SVM (with kernel scale 0.9)											
0–0 mm	13977	63	233	1.00	0.98	0.99	-	-	-	0.25	-
1–4 mm	800	66	505	0.92	0.61	0.74	-	-	-	0.00	-
2–9 mm	4082	315	1171	0.93	0.78	0.85	-	-	-	0.03	-
3–13 mm	7248	1658	193	0.81	0.97	0.89	-	-	-	0.08	-
All	26107	2102	2102	-	-	-	**0.84**	**0.86**	0.93	0.36	0.88

Feature ranking

To assess the importance of individual features in the multi-class classification process, their significance was analyzed using the MRMR and ReliefF algorithms (Table 11).

The MRMR algorithm ranked the distance to the nearest neighbor and the difference in curvature highest. This was followed by point density, the difference in the normal vector in the Z axis, and the difference in reflection intensity. In turn, the ReliefF algorithm indicated point density as the dominant feature, which gained a clear advantage over the others. The following features were ranked next: difference in the normal vector in the Z axis, difference in the normal vector in the X axis, and distance to the nearest neighbor.

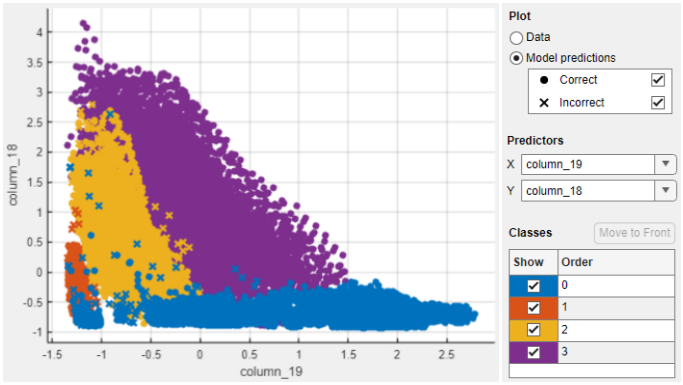
A comparison with the feature ranking for binary classification revealed similarities and differences. In both classifications, the displacement vector length and changes in local curvature features were highly significant. Features related to the normal vector (especially ΔNz and ΔNx) and change in intensity (ΔI) also ranked high, although their position differed depending on the algorithm. Color features (ΔR, ΔG, ΔB) and the coordinates of individual axes (ΔX, ΔY, ΔZ) were relatively unimportant in both classifications.

In the binary classification, the ReliefF algorithm gave the highest rating to the change in the normal vector in the Z axis. This suggests that for a simpler task – distinguishing between displaced and non-displaced points – features describing local changes in surface orientation were important. In the multi-class classification, ReliefF ranked the Density feature as the highest. For the MRMR algorithm, |ΔXYZ| dominated in both cases; however, in the multi-class classification, Density also gained greater importance, which slightly altered the hierarchy compared to the binary classification. Features related to intensity (ΔI) were crucial in the binary classification, with their importance decreasing in the multi-class classification.

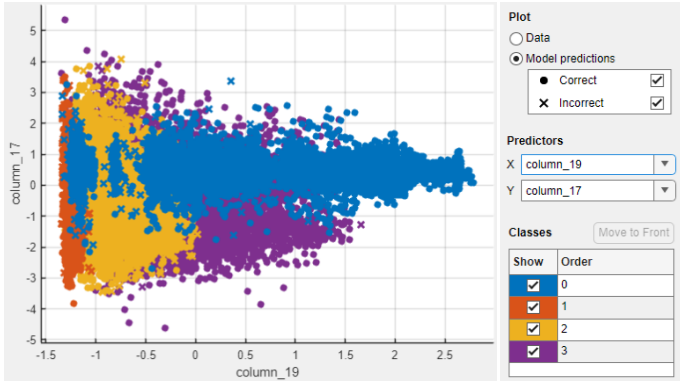
Figure 14 shows the relationship between density (column_19 in

Table 11. Ranking of features obtained using MRMR and ReliefF algorithms for Data sets 4 – four classes

Feature	Standardized MRMR score	Standardized ReliefF score
displacement vector length: |ΔXYZ|	**1.0**	**0.3**
changes in local curvature: ΔK	0.8	0.3
point density: Density	**0.8**	**1.0**
changes in normal vectors: ΔNz	0.8	0.4
change in intensity: ΔI	0.7	0.0
changes in normal vectors: ΔNx	0.7	0.4
changes in coordinates: ΔZ	0.6	0.3
changes in coordinates: ΔY	0.5	0.3
changes in color: ΔG	0.4	0.2
changes in normal vectors: ΔNy	0.4	0.3
changes in coordinates: ΔX	0.3	0.3
changes in color: ΔB	0.3	0.2
changes in color: ΔR	0.3	0.2



(a)



(b)

Figure 14. Graphical representation of the relationship between pairs of features: (a) density (column_19) and displacement vector length (column_18); (b) density (column_19) and changes in local curvature (column_17) for multi-class classification

Figure 14a) and the length of the displacement vector (column_18 in Figure 14a), as well as between density (column_19 in Figure 14b) and local curvature change (column_17 in Figure 14b). The graphs show the points of Data set 4, which was used to train the neural network for multi-class classification. Points in the same class form regular clusters or groups. The relationship between these two features is strong.

Analysis of results

The results of multi-class classification, performed on Data set 4, confirmed the effectiveness of the approach based on diverse training fields. Table 12 shows the percentage share of points assigned to individual deformation classes. The results indicate that the vast majority of points were assigned to class 0 (over 97–99% in all cases). The share of points in the other classes was extremely low, usually below 1%. Table 13 shows the average probabilities of assigning points to individual classes (0–3). This complements the data in Table 12, showing not only the percentage of points in the classes, but also the certainty of the model decisions.

The high average probability for class 0 (close to 1) confirms that the model was very confident in assigning most points to no deformation. In contrast, for classes corresponding to deformations (1–3), the probabilities are noticeably lower, indicating that the model had difficulty in clearly recognizing these cases.

Visualization of results

The analysis of the results showed that the model effectively distinguished between individual displacement classes. According to the adopted color scale, blue corresponded to class 0 (no displacement), green corresponded to class 1 (4 mm displacement), yellow corresponded to class 2 (9 mm displacement), and red corresponded to class 3 (13 mm displacement). The color scheme was retained in the point cloud visualizations presented in Figure 15.

The model correctly assigned points to the appropriate classes. For example, in a cloud simulating displacements of 4 mm, most of the displaced points were classified as class 1. Similarly, in a cloud simulating displacements of 9 mm, they were classified as class 2, and in a cloud simulating displacements of 13 mm, as class 3. The differences were clearly visible in the color scheme of the visualizations.

However, isolated cases of misclassification were observed. For position No. 1, in the point cloud simulating a displacement of 9 mm, prism P7 was assigned to class 3 instead of class 2. This may have been caused by an imprecise movement of the micrometric table. As a result, with a radius r equal to the deformation value used to calculate the local point density, the model could assign the object to a higher class.

As in the binary classification, in the case of stations 2 and 3, located at a greater angle to the object under investigation, areas were incorrectly classified as displacements. However, they differed in location.

Figures 16 and 17 present the results of multi-class classification obtained from a neural network model trained on Data set 4. The visualizations show fragments of the point cloud located in the area of two prisms: P1 – placed above the scanner, where the laser beam fell on the surface at a non-perpendicular angle, and P9 – placed at the same height as the scanner, where the laser beam fell at a perpendicular angle. In each visualization, the points were colored according to the displacement class assigned by the model. In the overall assessment, the model correctly identified the areas corresponding to the individual displacement values. However, some points were more difficult to classify unambiguously. For the P9 prism (Figure 17), which was located at the height of the scanner, the color distribution was more uniform and clearer. The perpendicular angle of incidence of the beam enabled better displacement mapping, confirming the effectiveness of classification

Table 12. Summary of multi-class classification results, including the percentage share of points assigned to individual classes (0, 1, 2, 3)

Simulated displacement [mm]	Class 0 - displacement equal 0 mm [%]			Class 1 - displacement equal 4 mm [%]			Class 2 - displacement equal 9 mm [%]			Class 3 - displacement equal 13 mm [%]		
	TLS station 2 - left	TLS station 1 - central	TLS station 3 - right	TLS station 2 - left	TLS station 1 - central	TLS station 3 - right	TLS station 2 - left	TLS station 1 - central	TLS station 3 - right	TLS station 2 - left	TLS station 1 - central	TLS station 3 - right
					Data set 4							
4	97.69	99.60	99.51	**2.27**	**0.33**	**0.41**	0.07	0.04	0.08	0.00	0.00	0.00
9	99.40	99.52	99.40	0.04	0.01	0.01	**0.31**	**0.54**	**0.40**	0.02	0.16	0.19
13	99.21	99.42	99.28	0.00	0.00	0.00	0.05	0.18	0.03	**0.60**	**0.53**	**0.68**

Table 13. Average probabilities of class occurrence (0, 1, 2, 3)

Simulated displacement [mm]	Class 0 - displacement equal 0 mm [%]			Class 1 - displacement equal 4 mm [%]			Class 2 - displacement equal 9 mm [%]			Class 3 - displacement equal 13 mm [%]		
	TLS station 2 - left	TLS station 1 - central	TLS station 3 - right	TLS station 2 - left	TLS station 1 - central	TLS station 3 - right	TLS station 2 - left	TLS station 1 - central	TLS station 3 - right	TLS station 2 - left	TLS station 1 - central	TLS station 3 - right
					Data set 4							
4	0.97	**1.00**	**1.00**	0.72	**0.86**	**0.86**	0.74	0.78	0.80	0.87	0.84	0.88
9	**1.00**	**1.00**	**1.00**	**0.80**	0.81	0.81	**0.91**	**0.88**	**0.86**	0.70	0.87	0.81
13	**1.00**	**1.00**	**1.00**	0.79	0.79	0.75	0.80	0.81	0.81	**0.90**	**0.96**	**0.97**

in conditions conducive to optimal measurement geometry.

Based on the results in Figure 16, it can be concluded that the algorithm correctly detected a displacement of 4 mm on the point cloud recorded from the second station. The remaining fragments contained misclassified point-like displacements. Incorrectly detected points moved on prism P1 may indicate that the measurement geometry affects the classification results. In addition, the effect shown in Figure 16 may be exacerbated by the problem of the laser beam slipping on the edges of the measured objects.

Tables 14 and 15 present a summary of the multi-class classification results. They contain the percentage share of points assigned to classes (0, 1, 2, 3) and the average probability of predicting the class for point cloud fragments shown in Figures 16 and 17, i.e., prism P1 and its surroundings, and prism P9 and its surroundings. The classifier was a neural network model trained on Data set 4 – test field for P2, P5, P6, and P8, data from all stations and series. Statistical data of point cloud fragments representing prism P1 confirm its displacement by 4 mm during scanning from station 2-left.

Unfortunately, the results of multi-class classification do not exhibit the same clear correlations as those obtained in binary classification. For the results obtained for point clouds representing prism P1, it should be noted that, firstly, the percentage of points with a class 0 prediction increases with growing displacement, as does the average probability of a class 0 prediction. Secondly, the percentage of points with a class 1, 2, or 3, and the average probability of a class 1, 2, or 3 prediction do not show a clear trend depending on the increase in displacement.

For the results obtained for point clouds representing prism P9, it should be noted that the percentage of points with a class 0 forecast decreases with rising displacement, while the average probability of a class 1, 2, or 3 forecast does not show a clear trend depending on the increase in displacement.

4 Discussion

The conducted research showed that the use of neural network models enables effective classification of displacements recorded using TLS point clouds. An important observation is that multi-class classification not only identifies deformation, but also allows for the estimation of its magnitude. This can be particularly important in engineering applications, such as monitoring the deformation of building walls, where it is crucial to detect not only displacement but also its magnitude. The subject matter of the article aligns with the modern trend of utilizing artificial intelligence for various surveying applications (Medina and Paffenroth, 2021; Zhao et al., 2022; Gojcic et al., 2020). The literature indicates a growing adoption of TLS techniques for monitoring buildings, infrastructure, and tunnels, as well as an increasing frequency of attempts to integrate them with classification algorithms and neural networks (Teng et al., 2022). An example of a directly related work, in which the authors construct a network architecture for recognizing deformations of building elements based on point clouds, is Chen et al. (2018). The results of this work confirm the feasibility of classifying geometric deformations in 3D clouds using neural networks. Another example is the description of tunnel cross-section deformation monitoring using methods based on TLS and 3D classification, as in the work of Camara et al. (2024), who use advanced techniques to track changes in the cross-section and stability of the object.

Both binary and multi-class analysis allowed for the correct assignment of points to the appropriate categories. In addition, 5-fold cross-validation in Classification Learner was performed by partitioning observations at the point level. Consequently, samples from the same prism and measurement epoch may appear in both the training and validation folds, potentially yielding overly optimistic performance estimates due to strong sample correlation. Nevertheless, the results obtained indicate that the position of the station relative to the object affects classification. The visualization results for individual prisms confirmed the high effectiveness of the model, while also revealing its limitations. Classification errors occurred more frequently in the case of stations with a non-perpendicular laser beam incidence angle. Under such conditions, points with higher measurement noise were recorded. These observations are consistent with the literature (Soudarissanane et al., 2011). TLS studies show that the angle of incidence significantly increases measurement noise (especially above ~60°). This reduces

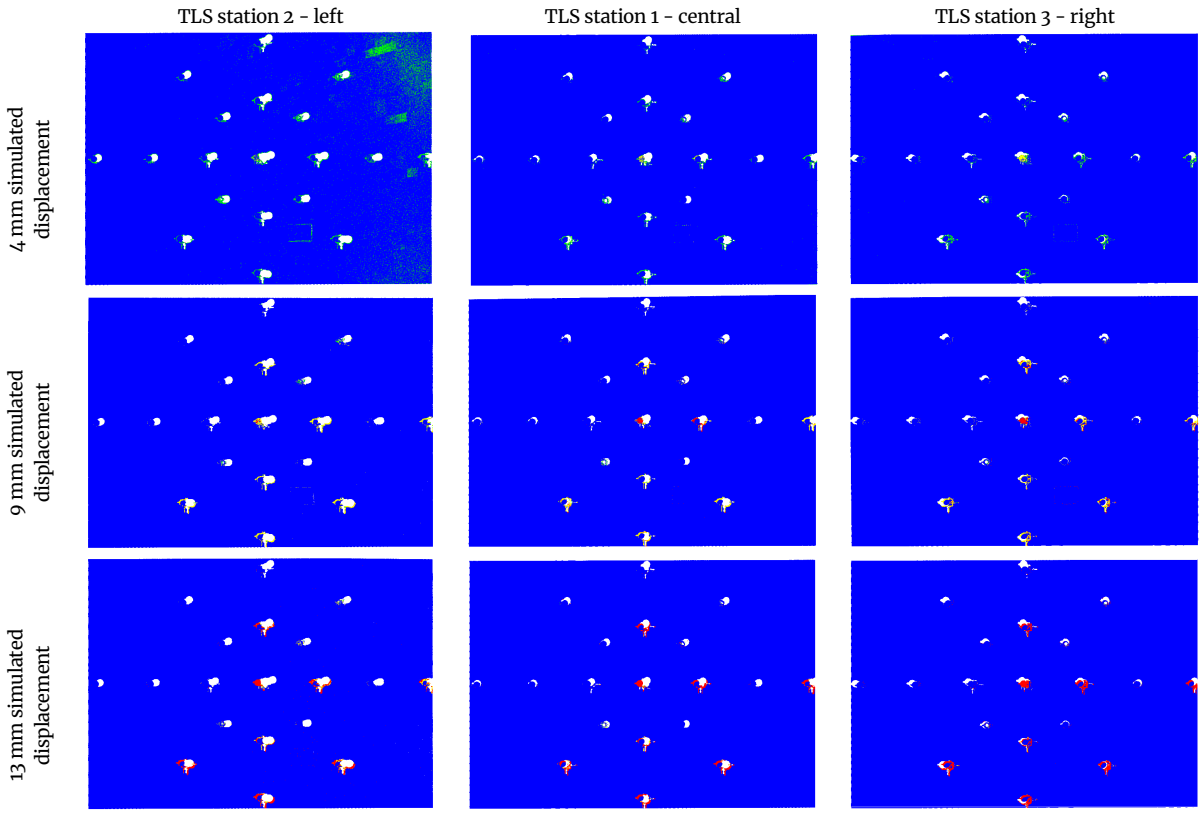


Figure 15. Visualization of multi-class classification results obtained using a neural network model trained on input Data set 4 – full rosette

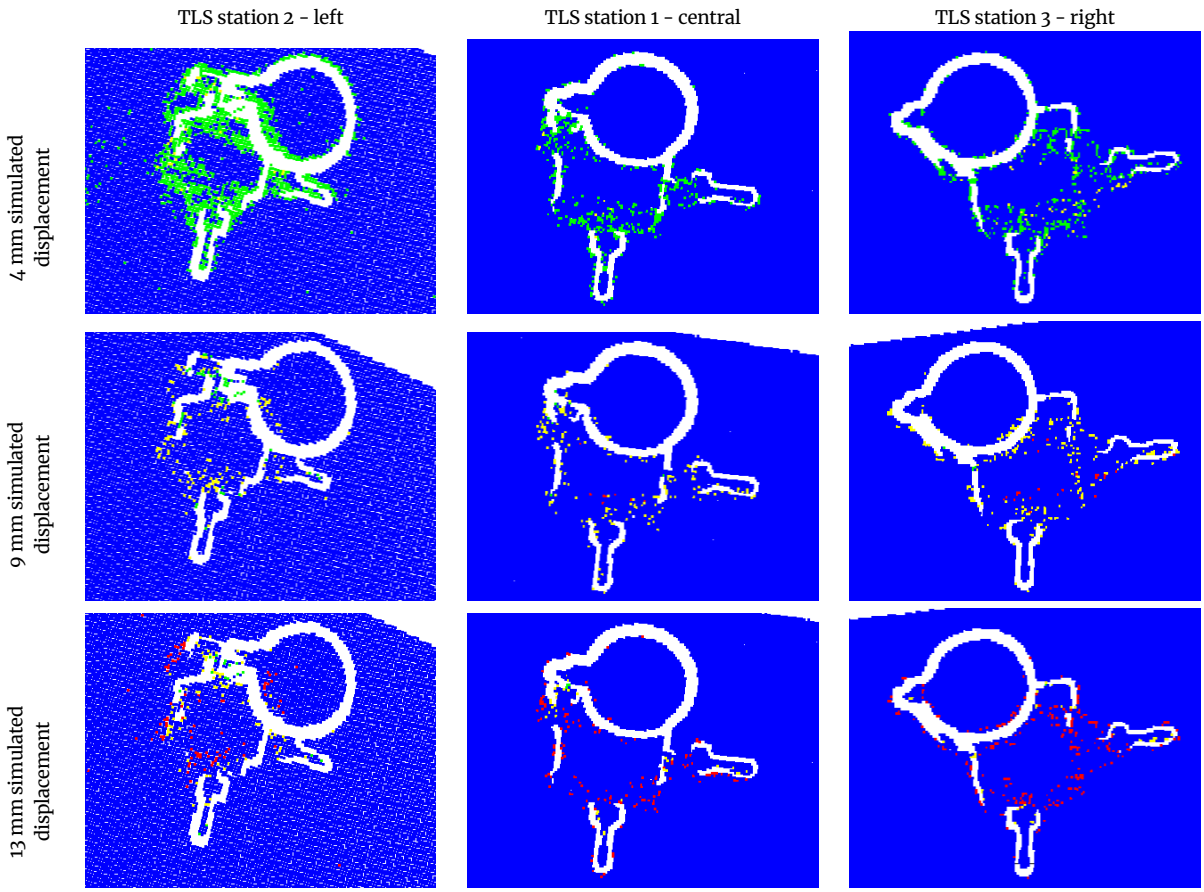


Figure 16. Visualization of multi-class classification results obtained using a neural network model trained on input Data set 4 – zoom on prism P1, which was located above the scanner

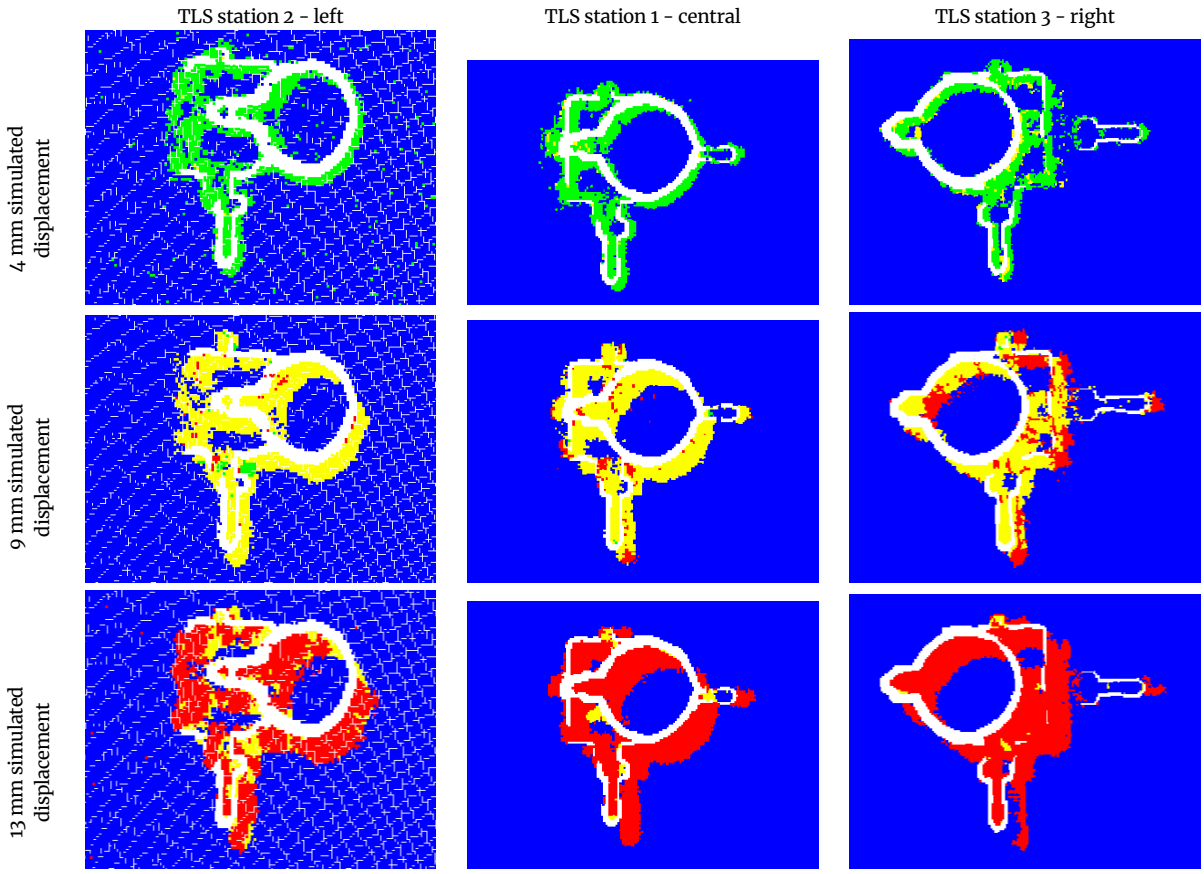


Figure 17. Visualization of multi-class classification results obtained using a neural network model trained on input Data set 4 – zoom on prism P9, which is located at the same height as the scanner (approximately perpendicular angle of incidence of the laser beam on the wall surface)

Table 14. Summary of multi-class classification results, including the percentage share of points assigned to classes (0,1,2,3) and the average probability of class prediction: classifier – neural network trained on Data set 4 – test field for P2, P5, P6, and P8, data from all stations and series. Statistical data of point cloud fragments representing prism P1, shown in Figure 16.

		TLS station 2 - left		TLS station 1 - central		TLS station 3 - right	
		part of cloud predicted for class [%]	average class probability	part of cloud predicted for class [%]	average class probability	part of cloud predicted for class [%]	average class probability
4 mm displacement	class 0	90.56	0.95	98.05	0.99	97.76	0.99
	class 1	9.25	0.84	1.90	0.79	2.02	0.79
	class 2	0.19	0.77	0.05	0.65	0.23	0.71
	class 3	0.00	0.00	0.00	0.00	0.00	0.00
9 mm displacement	class 0	98.52	1.00	99.13	1.00	98.78	1.00
	class 1	0.42	0.83	0.04	0.82	0.02	0.96
	class 2	1.06	0.85	0.79	0.83	1.09	0.86
	class 3	0.00	0.00	0.04	0.62	0.11	0.72
13 mm displacement	class 0	98.56	1.00	99.18	1.00	98.71	1.00
	class 1	0.13	0.80	0.01	0.61	0.00	0.79
	class 2	0.68	0.83	0.16	0.81	0.18	0.79
	class 3	0.63	0.80	0.64	0.84	1.11	0.87

Table 15. Summary of multi-class classification results, including the percentage share of points assigned to classes (0,1,2,3) and the average probability of class prediction: classifier – neural network trained on Data set 4 – test field for P2, P5, P6, and P8, data from all stations and series. Statistical data of point cloud fragments representing prism P9, shown in Figure 17.

		TLS station 2 - left		TLS station 1 - central		TLS station 3 - right	
		part of cloud predicted for class [%]	average class probability	part of cloud predicted for class [%]	average class probability	part of cloud predicted for class [%]	average class probability
4 mm displacement	class 0	90.31	0.95	94.29	0.99	94.91	1.00
	class 1	9.63	0.87	5.54	0.90	4.47	0.89
	class 2	0.06	0.71	0.17	0.76	0.62	0.76
	class 3	0.00	0.00	0.00	0.00	0.00	0.00
9 mm displacement	class 0	88.52	1.00	92.30	1.00	92.39	1.00
	class 1	0.36	0.82	0.04	0.74	0.02	0.74
	class 2	10.72	0.93	6.91	0.89	4.88	0.83
	class 3	0.40	0.74	0.75	0.71	2.72	0.80
13 mm displacement	class 0	84.60	0.99	89.60	1.00	90.58	1.00
	class 1	0.00	0.00	0.00	0.00	0.00	0.00
	class 2	3.27	0.75	0.50	0.79	0.17	0.77
	class 3	12.12	0.87	9.91	0.97	9.25	0.98

the precision of coordinates and derived features (e.g., normals, intensities). This naturally leads to greater difficulties in classifying displaced points, as local geometric features were more difficult to reproduce correctly.

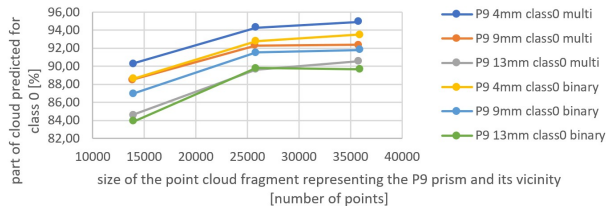
When establishing the relationship between the angle of reflection of the laser beam from the surface of the object and misclassified points, the authors do not relate this effect to the intensity of reflection. Therefore, the analysis did not focus on assessing the impact of signal quality or measurement accuracy in a strictly geometric sense. However, the studies indicate that the angle of incidence of the laser beam has a significant impact on the uneven density of the point cloud, which is the input data for the classification process. As the position of the TLS scanner changes, so does the geometry of the object observation and, consequently, the local density of the recorded points. In the case of the analyzed wall, the highest point cloud density was observed in areas where the laser beam fell at an angle close to a right angle. As the angle of incidence decreased, the point cloud density gradually decreased. This finding is directly reflected in the latest publications (Liu et al., 2025) and in the results of this study.

As shown in Figure 18, the sizes of the point cloud fragments representing the P9 prism and its immediate surroundings differ, even though they were extracted using the same spatial range. These differences are attributed to the use of three separate scanning stations located in different parts of the laboratory to extract the analyzed fragments, which resulted in variations in surface sampling densities.

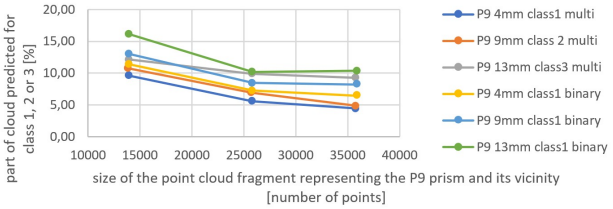
Additionally, it was observed that in both binary and multi-class classification, as the number of points in the analyzed cloud fragment increases, the percentage of points assigned to class 0 rises, whilst the percentage of points classified to classes 1, 2, and 3 decreases. This indicates a significant dependence of prediction results on the local density of the point cloud, which can lead to systematic interpretation errors in areas with heterogeneous sampling.

Irregular point cloud density is a key limitation of the devised methodology, but it also highlights an important direction for further development. The premise for further research in this area is the high ranking of features describing point density in the feature importance ranking, which confirms their significant impact on the classification process and the final analysis results.

Another significant limitation of the developed methodology is the need to determine the neighborhood radius a priori. It must be declared at the initialization stage of the algorithm that generates features describing the local geometry of the point cloud. The choice of this parameter directly affects the characteristics of the derived



(a)



(b)

Figure 18. Statistics of point cloud fragments representing the P9 prism within the vicinity of the same range. This is the relationship between the size of the cloud fragment (defined by the number of points in the cloud fragment) and the percentage of points with a prediction of: (a) class 0, (b) classes 1, 2, or 3. Each graph shows the results of both binary and multi-class classification

descriptors and, consequently, the classification effectiveness. The lack of a universal neighborhood radius suitable for different types of objects, point cloud densities, or scales of analyzed displacements necessitates empirical selection, limiting the universality of the method. In the study, radii equal to the values of the displacements were adopted, as their magnitudes were known. In real-world conditions, however, displacements are unknown. Future work should analyze the impact of the neighborhood radius on the classification results and assess the methodology under practical conditions.

It is worth noting that the primary objective of this research was to automate the detection of small displacements. The problem, therefore, is a strong imbalance between classes in the input set, with the "displacement" class being significantly smaller than the "no displacement" class. Such disproportion is typical in change and displacement detection, but it poses a significant challenge for machine learning. Models trained on unbalanced data tend to favor the dominant class, leading to reduced effectiveness in indicating actual displacements, particularly those with small amplitudes.

Class imbalance affects not only the model training process, but also its ability to generalize when classifying target point clouds, increasing the risk of overlooking areas that are actually subject to displacement. As a result, the obtained results may be characterized by high overall accuracy, but low sensitivity with respect to the less numerous class.

The conducted research confirms that high overall accuracy of trained models does not necessarily mean the ability to detect changes (the model "learns" the dominant class). A small training sample can lead to overfitting, which is consistent with ML practice for 3D data. The literature recommends augmentation techniques or the use of models that learn multiscale features to improve generalization with small training sets (Griffiths and Boehm, 2019). The ranking of features, in which geometric features were given greater weight in multi-class classification, is also reflected in the literature on point cloud classification methods. Review papers on 3D classification emphasize the importance of local features, such as spatial relationships and neighborhood context, for more complex classification tasks (Zhang et al., 2023, 2025). This work uses traditional neural networks and feature selection methods. However, the literature is increasingly leaning towards deep architectures for 3D segmentation and classification. A review of deep learning methods in point cloud classification reveals that these approaches are increasingly outperforming classical models, particularly for complex and large datasets (Zhang et al., 2023).

In summary, the findings of this study confirm the potential of using machine learning for TLS data analysis. However, they also indicate the need for further work in the following areas: minimizing the impact of station geometry on classification quality, integrating additional noise filtering methods, and extending the analysis to other types of displacement.

5 Conclusions

The developed procedure, covering stages from data acquisition through the calculation of geometric and radiometric features, to the construction of classification models, allowed for the assessment of changes in building structures. The results obtained indicate that binary classification, which distinguishes between displaced and non-displaced points, achieved higher overall accuracy than multi-class classification. In engineering applications, the ability to estimate the scale of deformation can be of significant practical importance.

However, it should be emphasized that the classifier's effectiveness largely depended on the quality and diversity of the training fields prepared. Insufficient number of samples or unbalanced data led to overfitting of the models, which negatively affected the stability of the results.

Analysis of feature rankings revealed that different parameters had varying significance levels, depending on the classification type. In binary classification, features related to local surface geometry, such as normal vectors and curvature, have proven key. In multi-class classification, however, global features such as displacement vector length and point density played a more important role.

The research also revealed the importance of measurement geometry and the positioning of laser scanners relative to the object being examined. In the case of stations with a sharp beam-incidence angle (station 2-left), a decrease in point density and an increase in misclassifications were observed.

In summary, the developed methodology demonstrated significant potential in detecting the displacements of building elements. The research revealed certain limitations, such as determining a universal neighborhood radius size, standardizing point cloud density, which is caused by a change in the location of the TLS station relative to the object, and the need to adapt mathematical models to test fields with disproportionate class sizes. Specifically, the study showed that beam incidence angles exceeding 60° significantly increase measurement noise, which complicates the classification of displaced points. However, the results are promising. The methodology described in the article provides a basis for future research. Future work should focus on expanding the procedure to detect displacements perpendicular to the wall surface (out-of-plane), which requires an expanded set of input features. Additionally, it is necessary to validate the methodology on real-world engineering structures. Ultimately, it can be concluded that machine learning methods can become a valuable component of modern structural monitoring systems, supporting both the detection and quantitative assessment of displacements.

Acknowledgements

The presented material is part of research conducted as part of a master's thesis entitled "Analysis of selected Machine Learning algorithms for investigating wall deformations in building structures based on point cloud" conducted at the Faculty of Geodesy and Cartography of the Warsaw University of Technology.

References

Alnuaimi, A. F. and Albaldawi, T. H. (2024). An overview of machine learning classification techniques. *BIO Web of Conferences*, 97:00133, doi:10.1051/bioconf/20249700133.

Amamra, A. and Boumaza, K. (2019). A Comparative Study of Coarse to Dense 3D Indoor Scene Registration Algorithms. In *2019 International Conference on Advanced Electrical Engineering (ICAEE)*, page 1–6. IEEE, doi:10.1109/icaee47123.2019.9014836.

Bae, C., Cho, G. R., Bae, J., Park, S., Lee, M., Kim, S., and Park, J. H. (2025). Cylindrical Scan Context: A Multi-Channel Descriptor for Vertical-Structure-Aware LiDAR Localization. *Sensors*, 25(23):7223, doi:10.3390/s25237223.

Bao, N., Zhang, T., Huang, R., Biswal, S., Su, J., and Wang, Y. (2023). A Deep Transfer Learning Network for Structural Condition Identification with Limited Real-World Training Data. *Structural Control and Health Monitoring*, 2023:1–18, doi:10.1155/2023/8899806.

Bao, S., Fang, X., Bu, H., Yu, X., and Cai, Z. (2025). Monitoring of the Deformation of Deep Foundation Pit Using 3D Laser Scanning. *Buildings*, 15(8):1290, doi:10.3390/buildings15081290.

Becker, D., Raddatz, L., Roussel, C., and Klonowski, J. (2024). Analysis methods for deformation detection using TLS and UAS data on the example of a landslide simulation. *International Journal of Geo-Engineering*, 15(1), doi:10.1186/s40703-023-00203-z.

Besl, P. J. and McKay, N. D. (1992). Method for registration of 3-D shapes. In Schenker, P. S., editor, *Sensor Fusion IV: Control*

Paradigms and Data Structures, volume 1611, page 586–606. SPIE, doi:10.1117/12.57955.

Camara, M., Wang, L., and You, Z. (2024). Tunnel Cross-Section Deformation Monitoring Based on Mobile Laser Scanning Point Cloud. *Sensors*, 24(22):7192, doi:10.3390/s24227192.

Cennamo, A., Kaestner, F., and Kummert, A. (2021). A Neural Network Based System for Efficient Semantic Segmentation of Radar Point Clouds. *Neural Processing Letters*, 53(5):3217–3235, doi:10.1007/s11063-021-10544-4.

Chen, J., Cho, Y. K., and Ueda, J. (2018). Sampled-Point Network for Classification of Deformed Building Element Point Clouds. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, page 2164–2169. IEEE, doi:10.1109/icra.2018.8461095.

Chen, L., Feng, C., Ma, Y., Zhao, Y., and Wang, C. (2024). A review of rigid point cloud registration based on deep learning. *Frontiers in Neurorobotics*, 17, doi:10.3389/fnbot.2023.1281332.

Crall, J. (2023). The MCC approaches the geometric mean of precision and recall as true negatives approach infinity. *arXiv preprint arXiv:2305.00594*.

De Gélis, I., Corpetti, T., and Lefèvre, S. (2024). Change detection needs change information: Improving deep 3-D point cloud change detection. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–10, doi:10.1109/TGRS.2024.3359484.

De Gélis, I., Lefèvre, S., and Corpetti, T. (2023). DC3DCD: Unsupervised learning for multiclass 3D point cloud change detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 206:168–183, doi:10.1016/j.isprsjprs.2023.10.022.

Dey, S., Sharma, A., Rai, H., and Biswas, S. (2024). Comprehensive Analysis of Structural Defects in Various Structures Using TLS Data and Machine Learning. In *2024 IEEE International Conference on Future Machine Learning and Data Science (FMLDS)*, page 457–461. IEEE, doi:10.1109/fmlds63805.2024.00086.

Fontes, M., Bakoň, M., Cunha, A., and Sousa, J. J. (2025). Deep Learning Meets InSAR for Infrastructure Monitoring: A Systematic Review of Models, Applications, and Challenges. *Sensors*, 25(23):7169, doi:10.3390/s25237169.

Gili, J. A., Corominas, J., and Rius, J. (2000). Using Global Positioning System techniques in landslide monitoring. *Engineering Geology*, 55(3):167–192, doi:10.1016/s0013-7952(99)00127-1.

Gojcic, Z., Zhou, C., and Wieser, A. (2020). F2S3: Robustified determination of 3D displacement vector fields using deep learning. *Journal of Applied Geodesy*, 14(2):177–189, doi:10.1515/jag-2019-0044.

Griffiths, D. and Boehm, J. (2019). Weighted point cloud augmentation for neural network training data class-imbalance. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W13:981–987, doi:10.5194/isprs-archives-xlii-2-w13-981-2019.

Hardt, M. and Recht, B. (2021). Patterns, predictions, and actions: A story about machine learning. *arXiv preprint arXiv:2102.05242*.

Haykin, S. (2009). *Neural networks and learning machines*. Pearson Education, Inc., McMaster University, Hamilton, Ontario, Canada.

Heaton, J. (2017). Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning: The MIT Press, 2016, 800 pp, ISBN: 0262035618. *Genetic Programming and Evolvable Machines*, 19(1–2):305–307, doi:10.1007/s10710-017-9314-z.

Hexsel, B., Vhavle, H., and Chen, Y. (2022). DICP: Doppler iterative closest point algorithm. *arXiv preprint arXiv:2201.11944*.

Karsznia, K., Osada, E., and Muszyński, Z. (2023). Real-Time Adjustment and Spatial Data Integration Algorithms Combining Total Station and GNSS Surveys with an Earth Gravity Model. *Applied Sciences*, 13(16):9380, doi:10.3390/app13169380.

Karsznia, K., Świerczyńska, E., Książek, K., and Odziemczyk, W. (2025). Development of an expert system for the deformation monitoring of historical sites using Artificial Intelligence (AI). In *6th Joint International Symposium on Deformation Monitoring (JISDM) 7–9 April 2025, Karlsruhe, Germany*.

Knight, P. E., Muda, A. K., Yusof, N. M., and Muda, N. A. (2023). Enhancement of Embedded Feature Selection Method for 3D Molecular Structure of Amphetamine-Type Stimulants (ATS) Drugs. *International Journal of Computer Information Systems and Industrial Management Applications*, 15:9–9.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444, doi:10.1038/nature14539.

Li, J. and Pisztora, V. A. (2021). *Unsupervised and Semisupervised Learning*. Wiley StatsRef: Statistics Reference Online, doi:10.1002/9781118445112.stat08320.

Liu, M., Liu, G., Zhao, M., Zhang, X., Yang, K., and Chen, Y. (2025). Optimizing 3D Laser Scanning Parameters for Early-Stage Defect Detectability in Subgrade Condition Monitoring. *Sensors*, 25(23):7174, doi:10.3390/s25237174.

Liu, W., Sun, J., Li, W., Hu, T., and Wang, P. (2019). Deep Learning on Point Clouds and Its Application: A Survey. *Sensors*, 19(19):4188, doi:10.3390/s19194188.

Liu, W., Wu, H., and Chirikjian, G. S. (2021). LSG-CPD: Coherent point drift with local surface geometry for point cloud registration. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15293–15302.

Luo, G., He, B., Xiong, Y., Wang, L., Wang, H., Zhu, Z., and Shi, X. (2023). An Optimized Convolutional Neural Network for the 3D Point-Cloud Compression. *Sensors*, 23(4):2250, doi:10.3390/s23042250.

Łępicka, M. and Kornuta, T. (2017). Registration of RGB-D images: comparison of pairwise registration variants. *Pomiary Automatyka Robotyka*, 21(2):5–14, doi:10.14313/par_224/5.

Medina, F. P. and Paffenroth, R. (2021). *Machine Learning in LiDAR 3D Point Clouds*, pages 113–133. Springer International Publishing, Cham, doi:10.1007/978-3-030-79891-8_6.

Peng, H., Long, F., and Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, doi:10.1109/tpami.2005.159.

Peng, Y., Lin, S., Wu, H., and Cao, G. (2023). Point Cloud Registration Based on Fast Point Feature Histogram Descriptors for 3D Reconstruction of Trees. *Remote Sensing*, 15(15):3775, doi:10.3390/rs15153775.

Pomerleau, F., Colas, F., and Siegwart, R. (2015). A Review of Point Cloud Registration Algorithms for Mobile Robotics. *Foundations and Trends® in Robotics*, 4(1):1–104, doi:10.1561/2300000035.

Prakash, V. J. and Nithya, D. L. (2014). A Survey On Semi-Supervised Learning Techniques. *International Journal of Computer Trends and Technology*, 8(1):25–29, doi:10.14445/22312803/ijctt-v8p105.

Prószyński, W. and Kwaśniak, M. (2015). *Podstawy geodezyjnego wyznaczania przemieszczeń: pojęcia i elementy metodyki*. Oficyna Wydawnicza Politechniki Warszawskiej.

Rathod, Y. (2024). A Survey of Machine Learning Techniques for Artificial Intelligence. *International Journal of Computer Techniques*, 11(4):6–10.

Robnik-Šikonja, M. and Kononenko, I. (2003). Theoretical and Empirical Analysis of ReliefF and RReliefF. *Machine Learning*, 53(1–2):23–69, doi:10.1023/a:1025667309714.

Rusinkiewicz, S. and Levoy, M. (2001). Efficient variants of the ICP algorithm. In *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, IM-01, page 145–152. IEEE Comput. Soc, doi:10.1109/im.2001.924423.

Rusu, R. B., Blodow, N., and Beetz, M. (2009). Fast point feature histograms (FPFH) for 3D registration. In *2009 IEEE international conference on robotics and automation*, pages 3212–3217. IEEE.

Scaioni, M., Marsella, M., Crosetto, M., Tornatore, V., and Wang, J. (2018). Geodetic and Remote-Sensing Sensors for Dam Deformation Monitoring. *Sensors*, 18(11):3682, doi:10.3390/s18113682.

Si, H., Qiu, J., and Li, Y. (2022). A Review of Point Cloud Registration Algorithms for Laser Scanners: Applications in Large-

Scale Aircraft Measurement. *Applied Sciences*, 12(20):10247, doi:10.3390/app122010247.

Sirpa-Poma, J. W., Calle, J., Uscamayta-Ferrano, E., Molina-Carpio, J., Satgé, F., Cuentas Toledo, O., Duran, R., Pacheco Mollinedo, P., Hussain, R., and Pillco-Zolá, R. (2025). Development of Hourly Resolution Air Temperature Across Titicaca Lake on Auxiliary ERA5 Variables and Machine Learning-Based Gap-Filling. *Sensors*, 25(23):7165, doi:10.3390/s25237165.

Soudarissanane, S., Lindenbergh, R., Menenti, M., and Teunissen, P. (2011). Scanning geometry: Influencing factor on the quality of terrestrial laser scanning points. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(4):389–399, doi:10.1016/j.isprsjprs.2011.01.005.

Spears, B. K., Brase, J., Bremer, P.-T., Chen, B., Field, J., Gaffney, J., Kruse, M., Langer, S., Lewis, K., Nora, R., Peterson, J. L., Jayaraman Thiagarajan, J., Van Essen, B., and Humbird, K. (2018). Deep learning: A guide for practitioners in the physical sciences. *Physics of Plasmas*, 25(8), doi:10.1063/1.5020791.

Świerczyńska, E., Karsznia, K., Książek, K., and Odziemczyk, W. (2024). Investigating diverse photogrammetric techniques in the hazard assessment of historical sites of the Museum of the Coal Basin Area in Będzin, Poland. *Reports on Geodesy and Geoinformatics*, 118(1), doi:10.2478/rgg-2024-0016.

Syed, I. and Lokhande, V. (2024). An overview of the supervised machine learning. *International Research Journal of Modernization in Engineering Technology and Science*, doi:10.56726/irjmets51366.

Sztubecki, J., Topoliński, S., Mrówczyńska, M., Bağrıaçık, B., and Beycioğlu, A. (2022). Experimental Research of the Structure Condition Using Geodetic Methods and Crackmeter. *Applied Sciences*, 12(13):6754, doi:10.3390/app12136754.

Tareen, S. A. K. and Saleem, Z. (2018). A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK. In *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, page 1–10. IEEE, doi:10.1109/icomet.2018.8346440.

Teng, J., Shi, Y., Wang, H., and Wu, J. (2022). Review on the Research and Applications of TLS in Ground Surface and Constructions Deformation Monitoring. *Sensors*, 22(23):9179, doi:10.3390/s22239179.

Vakili, M., Ghamsari, M., and Rezaei, M. (2020). Performance analysis and comparison of machine and deep learning algorithms for IoT data classification. *arXiv preprint arXiv:2001.09636*.

Wojda, D. (2025). Analysis of selected machine learning algorithms applied to the study of wall deformation in buildings based on point clouds (in Polish). Master's thesis, Warsaw University of Technology.

Yang, J., Zhang, J., Cai, Z., and Fang, D. (2023). Novel 3D local feature descriptor of point clouds based on spatial voxel homogenization for feature matching. *Visual Computing for Industry, Biomedicine, and Art*, 6(1), doi:10.1186/s42492-023-00145-4.

Yang, L., Xu, S., Yang, Z., He, J., Gong, L., Wang, W., Li, Y., Wang, L., and Chen, Z. (2025). Fast Registration Algorithm for Laser Point Cloud Based on 3D-SIFT Features. *Sensors*, 25(3):628, doi:10.3390/s25030628.

Yuan, H., Li, G., Wang, L., and Li, X. (2025). Research on the Improved ICP Algorithm for LiDAR Point Cloud Registration. *Sensors*, 25(15):4748, doi:10.3390/s25154748.

Zaczek-Peplinska, J. and Popielski, P. (2013). *Terrestrial laser scanning technology as a tool for verification of the geometry of numerical models of behaviour of hydro technical structures (in Polish)*, pages 2018–230. Warszawa: Instytut Meteorologii i Gospodarki Wodnej – Państwowy Instytut Badawczy.

Zhang, H., Wang, C., Tian, S., Lu, B., Zhang, L., Ning, X., and Bai, X. (2023). Deep learning-based 3D point cloud classification: A systematic survey and outlook. *Displays*, 79:102456, doi:10.1016/j.displa.2023.102456.

Zhang, Y., Ding, C., and Li, T. (2008). Gene selection algorithm by combining reliefF and mRMR. *BMC Genomics*, 9(S2), doi:10.1186/1471-2164-9-s2-s27.

Zhang, Y., Wang, X., Lyu, X., Zhang, L., Song, W., and Zhang, R. (2025). Segment-Based SLAM Registration Optimization Algorithm Combining NDT and PL-ICP. *Sensors*, 25(23):7175, doi:10.3390/s25237175.

Zhao, G., Li, J., Xi, J., and Luo, L. (2024). An Efficient and Stable Registration Framework for Large Point Clouds at Two Different Moments. *Sensors*, 24(22):7174, doi:10.3390/s24227174.

Zhao, Y., Seo, H., and Chen, C. (2022). Displacement analysis of point cloud removed ground collapse effect in SMW by CANUPO machine learning algorithm. *Journal of Civil Structural Health Monitoring*, 12(2):447–463, doi:10.1007/s13349-022-00555-7.

Zhou, X., Zhu, Q., Zhang, Q., and Du, Y. (2025). Machine learning driven 3D point cloud displacement reconstruction and structural safety evaluation of retaining structures. *Measurement*, 256:118540, doi:10.1016/j.measurement.2025.118540.

Zhuang, Y., Chen, W., Jin, T., Chen, B., Zhang, H., and Zhang, W. (2022). A Review of Computer Vision-Based Structural Deformation Monitoring in Field Environments. *Sensors*, 22(10):3789, doi:10.3390/s22103789.

Zienkiewicz, O. C., Taylor, R. L., and Zhu, J. Z. (2013). *The Finite Element Method: Its Basis and Fundamentals*. Elsevier, doi:10.1016/b978-1-85617-633-0.00019-8.