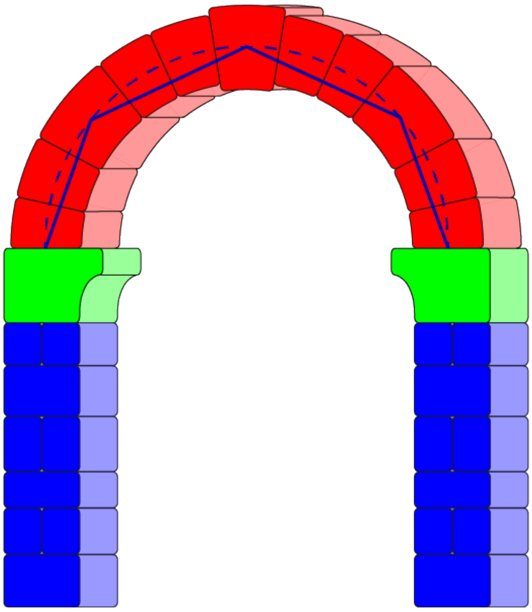


SEG4D

Segmentation for Cultural Heritage Diagnosis

This software has been funded by the Comunidad de Madrid through the call Research Grants for Young Investigators from Universidad Politécnica de Madrid

Version: Alpha

USER GUIDE

Damage evaluation

Index

2 Damage evaluation

In this tab you can perform a damage mapping of the construction supported by differente processing algoritmis. These algorithms include artifitial intelligence approaches.

IMPORTANT: You must first prepare your point cloud to work correctly with this software. It is highly recommended to perform a construction system segmentation before carrying out this stage. The section «Pre-processing» explains how to do it.

Types of damages

According with the publication «*Detection of damage in heritage constructions based on 3D point clouds. A systematic review*», the following strategies could be used to detect damage:

Geometric-based methods (those based on geometrical relationships of the point cloud):

Strategies based on manual mapping (G-MM)

Strategies based on sections and curve fitting algorithms (G-SC)

Strategies based on the computation of the point-to-point distance (G-p2pd)

Strategies based on the computation of the point-to-primitive distance (G-p2prd)

Strategies based on the computation of the point-to-3D model distance (G-p23Dd)

Strategies based on geometrical features (G-GF)

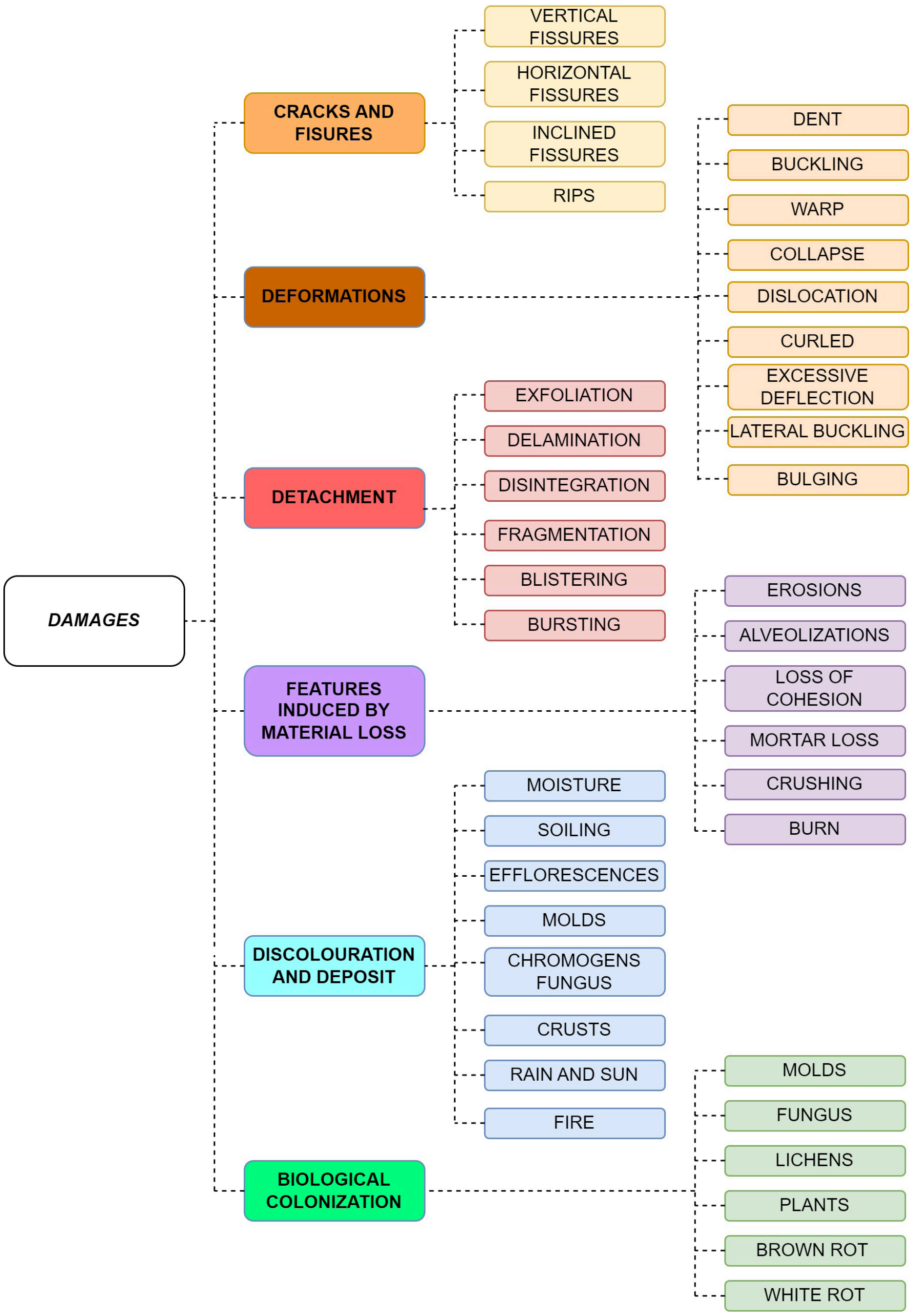
Radiometric-based methods (those based on colour or intensity ratios of the point cloud):

Strategies based on manual mapping (R-MM)

Strategies based on threshold segmentation (R-TS)

Strategies based on clustering segmentation (R-CS)

The rest of the section will show a set of summary sheets with typologies of damage and strategies to detect it and to help you with the type of strategy to use depending on the type of damage, whether it is geometric or radiometric.



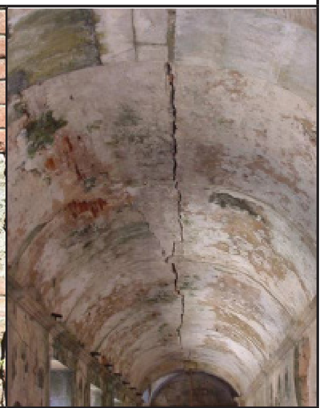
1.1. *Timber, masonry, metallic and concrete damages*

Cracks and fisures

Title	Cracks and fisures
Types of damage	Vertical fissures, horizontal fissures, inclined fissures, rips
Short description	An opening or slit that crosses the surface of single or contiguous vertical, horizontal, inclined or curved (discharging arch) structural elements along which it has started to split without breaking them apart. They may not always be structural in nature, but may affect non-structural elements.

https://blogdaengenhariacivil.wordpress.com/2015/01/26/trincas-e-fissuras/

https://blog.mcvaldezorras.com/2019/08/como-detectar-defectos-estructurales-vivienda.html



http://www.iaa-conservation.org.il/pdf/engineers2011/11_Case%20studies%20on%20monuments.pdf

Interrelations between damages	Displacement/deformation Leaning Bulging/buckling Detachment Material loss Biological colonization (non-structural)
Type of strategy to detect the damage	G-P2PRD, G-GF, G-P2PD, R-MM, R-TS
Some references of interest	[1] L.J. Sánchez-Aparicio, F.L. del Blanco-García, D. Mencías-Carrizosa, P. Villanueva-Llauradó, J.R. Aira-Zunzunegui, D. Sanz-Arauz, R. Pierdicca, J. Pinilla-Melo, J. Garcia-Gago, Detection of damage in heritage constructions based on 3D point clouds. A systematic review, Journal of Building Engineering 77 (2023). https://doi.org/10.1016/J.JOBE.2023.107440. [2] R.A. Galantucci, F. Fatiguso, Advanced damage detection techniques in historical buildings using digital photogrammetry and 3D surface anlysis, J. Cult. Herit. 36 (2019) 51–62, https://doi.org/10.1016/j.culher.2018.09.014. [3] R.L. Wood, M.E. Mohammadi, Feature-based point cloud-based assessment of heritage structures for nondestructive and noncontact surface damage detection, Heritage 4 (2021) 775–793, https://doi.org/10.3390/HERITAGE4020043. [4] J. De Matías, F. Berenguer, J.P. Cortés, J.J. De Sanjosé, A. Atkinson, Laser scanning for the geometric study of the alcántara bridge and coria cathedral, in: Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci., the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Trento, Italy, 2013, pp. 51–56, https://doi.org/10.5194/isprsarchives-xl-5-w1-51-2013. [5] C. Suchocki, Comparison of time-of-flight and phase-shift TLS intensity data for the diagnostics measurements of buildings, Materials 13 (2020), https://doi. org/10.3390/MA13020353.

Deformations

Title	Deformations
Types of damage	Dent, buckling, warp, collapse, dislocation, curled, excessive deflection, lateral buckling, bulging
Short description	Alterations in the position, shape, or dimensions of a structure or structural component, caused by the forces acting upon them. They can manifest as displacements, bending, twisting, or changes in shape, without necessarily compromising the integrity of the material.


https://advanceblastcleaning.co.uk/timber/


https://www.icomos.org/en


https://syrian-heritage.org/about-us/built-heritage-documentation/catalogue-of-damage/

Interrelations between damages	Bending Structural cracks Detachment Material loss Rot
Type of strategy to detect the damage	G-MM, G-SC, G-P2PRD, G-GF, G-P23DD, G-P2PD
Some references of interest	[1] L.J. Sánchez-Aparicio, F.L. del Blanco-García, D. Mencías-Carrizosa, P. Villanueva-Llauradó, J.R. Aira-Zunzunegui, D. Sanz-Arauz, R. Pierdicca, J. Pinilla-Melo, J. Garcia-Gago, Detection of damage in heritage constructions based on 3D point clouds. A systematic review, Journal of Building Engineering 77 (2023). https://doi.org/10.1016/J.JOBE.2023.107440. [2] G. Teza, A. Pesci, Geometric characterization of a cylinder-shaped structure from laser scanner data: development of an analysis tool and its use on a leaning bell tower, J. Cult. Herit. 14 (2013) 411–423, https://doi.org/10.1016/J.CULHER.2012.10.015. [3] M. Batur, O. Yilmaz, H. Ozener, A case study of deformation measurements of istanbul land walls via terrestrial laser scanning, IEEE J. Sel. Top. Appl. Earth Obs. Rem. Sens. 13 (2020) 6362–6371, https://doi.org/10.1109/JSTARS.2020.3031675. [4] J. P. Sestras, S. Roșca, Ștefan Bilașco, S. Naș, S.M. Buru, L. Kovacs, V. Spalevi´c, A.F. Sestras, Feasibility assessments using unmanned aerial vehicle technology in heritage buildings: rehabilitation-restoration, spatial analysis and tourism potential analysis, Sensors 20 (2020), https://doi.org/10.3390/S20072054. [5] J.W. Fang, Z. Sun, Y.R. Zhang, Tls-fem integrated structural deformation analysis on the beamless hall at Nanjing, China, in: Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. - ISPRS Arch., International Society for Photogrammetry and Remote Sensing, Beijing, China, 2021, pp. 215–220, https://doi.org/10.5194/isprs-Archives-XLVI-M-1-2021-215-2021.

DETACHMENT

Title	Detachment
Types of damage	Exfoliation, delamination, disintegration, fragmentation, blistering, bursting
Short description	Separation, loss of material layers or components from a substrate, often due to various factors such as mechanical stress, chemical actions, or environmental conditions. It can manifest as the detachment of thin layers, raised elevations, loss of surface material, fragmentation, cracking, or corrosion, impacting the integrity and appearance of the affected material.

https://syrian-heritage.org/about-us/built-heritage-documentation/catalogue-of-damage/

https://syrian-heritage.org/about-us/built-heritage-documentation/catalogue-of-damage/

https://syrian-heritage.org/about-us/built-heritage-documentation/catalogue-of-damage/

Interrelations between damages	Material loss Fragmentation Crust Moist area Loss of cohesion
Type of strategy to detect the damage	G-GF, R-MM, R-TS
Some references of interest	[1] L.J. Sánchez-Aparicio, F.L. del Blanco-García, D. Mencías-Carrizosa, P. Villanueva-Llauradó, J.R. Aira-Zunzunegui, D. Sanz-Arauz, R. Pierdicca, J. Pinilla-Melo, J. Garcia-Gago, Detection of damage in heritage constructions based on 3D point clouds. A systematic review, Journal of Building Engineering 77 (2023). https://doi.org/10.1016/J.JOBE.2023.107440. [2] Z. Hadavandsiri, D.D. Lichti, A. Jahraus, D. Jarron, Concrete preliminary damage inspection by classification of terrestrial laser scanner point clouds through systematic threshold definition, ISPRS Int. J. Geo-Inf. 8 (2019), https://doi.org/10.3390/IJGI8120585. [3] M.L. Turco, M. Mattone, F. Rinaudo, Metric survey and bim technologies to record decay conditions, Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. - ISPRS Arch. 42 (2017) 261–268, https://doi.org/10.5194/ISPRS-ARCHIVES-XLII-5-W1-261-2017. [4] Q. Li, X. Cheng, Damage detection for historical architectures based on TLS intensity data, in: Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. - ISPRS Arch., the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Prague, Czech Republic, 2018, pp. 915–921, https://doi.org/10.5194/ISPRS-ARCHIVES-XLII-3-915-2018. [5] M. Kedzierski, P. Walczykowski, M. Wojtkowska, A. Fryskowska, Integration of point clouds and images acquired from a low-cost nir camera sensor for cultural heritage purposes, Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. - ISPRS Arch. 42 (2017) 407–414, https://doi.org/10.5194/ISPRS-ARCHIVES-XLII-2-W5-407-2017.

Material loss

Title	Material loss
Types of damage	Erosions, alveolizations, loss of cohesion, mortar loss, crushing, burn
Short description	Deterioration or removal of substance from the surface or structure of a material, resulting in changes to its appearance, integrity, or functionality. This loss can occur due to various factors such as erosion, cavitation, chemical reactions, thermal exposure, mechanical damage, or intentional actions. It may lead to smoothed surfaces, formation of cavities, surface damage, loss of particle bond, or intentional tool marks, impacting the material's aesthetic, structural, or functional properties.

https://syrian-heritage.org/about-us/built-heritage-documentation/catalogue-of-damage/

https://syrian-heritage.org/about-us/built-heritage-documentation/catalogue-of-damage/

https://svipolylatur.org/product_details/117002240.html

Interrelations between damages	Detachment Biological colonization Material loss Pitting Disintegration
Type of strategy to detect the damage	G-MM, G-SC, G-P2PD, G-P2PRD, G-P23DD, G-GF, R-MM, R-TS
Some references of interest	[1] L.J. Sánchez-Aparicio, F.L. del Blanco-García, D. Mencías-Carrizosa, P. Villanueva-Llauradó, J.R. Aira-Zunzunegui, D. Sanz-Arauz, R. Pierdicca, J. Pinilla-Melo, J. Garcia-Gago, Detection of damage in heritage constructions based on 3D point clouds. A systematic review, Journal of Building Engineering 77 (2023). https://doi.org/10.1016/J.JOBE.2023.107440. [2] S. Takhirov, K.M. Mosalam, M.A. Moustafa, L. Myagkova, B. Quigley, Laser scanning, modeling, and analysis for damage assessment and restoration of historical structures, in: COMPDYN 2015 - 5th ECCOMAS Themat. Conf. Comput. Methods Struct. Dyn. Earthq. Eng., National Technical University of Athens, Crete Island, Greece, 2015, pp. 2375–2395, https://doi.org/10.7712/120115.3545.1384. [3] G. Tucci, G. Bartoli, M. Betti, V. Bonora, M. Korumaz, A.G. Korumaz, Advanced procedure for documenting and assessment of cultural heritage: from laser scanning to finite element, in: IOP Conf. Ser. Mater. Sci. Eng., Institute of Physics Publishing, 2018, https://doi.org/10.1088/1757-899X/364/1/012085. [4] G. Teza, A. Pesci, Evaluation of the temperature pattern of a complex body from thermal imaging and 3D information: a method and its MATLAB implementation, Infrared Phys. Technol. 96 (2019) 228–237, https://doi.org/10.1016/J.INFRARED.2018.11.029. [5] R. Nespeca, L. De Luca, Analysis, thematic maps and data mining from point cloud to ontology for software development, in: Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. - ISPRS Arch., International Society for Photogrammetry and Remote Sensing, Prague, Czech Republic, 2016, pp. 347–354, https://doi.org/10.5194/isprsarchives-XLI-B5-347-2016.

Discouloration and deposit

Title	Discouloration and deposit
Types of damage	Moisture, soiling, efflorescence, crusts, patinas, chromogens fungus, molds, rain and sun, fire.
Short description	Changes or accumulations on the surface of materials, often altering their colour, texture, or appearance. These can result from various sources such as atmospheric pollutants, biological growth, soluble salts, natural aging, or industrial emissions. They may manifest as crusts, darkening, whitish powdery layers, or chromatic modifications, impacting the aesthetic and sometimes structural integrity of the material.

https://syrian-heritage.org/about-us/built-heritage-documentation/catalogue-of-damage/

https://syrian-heritage.org/about-us/built-heritage-documentation/catalogue-of-damage/

https://pixabay.com/es/photos/par%C3%ADs-el-museo-del-louvre-lions-gate-2373702/

Interrelations between damages	Biological colonization Encrustration Material loss Detachment Lichens
Type of strategy to detect the damage	G-P2PRD, G-GF, G-P2PD, R-MM, R-TS
Some references of interest	[1] L.J. Sánchez-Aparicio, F.L. del Blanco-García, D. Mencías-Carrizosa, P. Villanueva-Llauradó, J.R. Aira-Zunzunegui, D. Sanz-Arauz, R. Pierdicca, J. Pinilla-Melo, J. Garcia-Gago, Detection of damage in heritage constructions based on 3D point clouds. A systematic review, Journal of Building Engineering 77 (2023). https://doi.org/10.1016/J.JOBE.2023.107440. [2] E. Valero, A. Forster, F. Bosché, E. Hyslop, L. Wilson, A. Turmel, Automated defect detection and classification in ashlar masonry walls using machine learning, Autom. ConStruct. 106 (2019), 102846, https://doi.org/10.1016/J.AUTCON.2019.102846. [3] J. Taraben, G. Morgenthal, Integration and comparison methods for multitemporal image-based 2D annotations in linked 3D building documentation, Rem. Sens. 14 (2022), https://doi.org/10.3390/RS14092286. [4] M. Balzani, F. Maietti, L. Rossato, 3D data processing toward maintenance and conservation. the integrated digital documentation of casa de vidro, in: ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci., Copernicus GmbH, 2019, pp. 65–72, https://doi.org/10.5194/isprs-archives-XLII-2-W9-65-2019. [5] J.S. Pozo-Antonio, I. Puente, M.F.C. Pereira, C.S.A. Rocha, Quantification and mapping of deterioration patterns on granite surfaces by means of mobile LiDAR data, Meas. J. Int. Meas. Confed. 140 (2019) 227–236, https://doi.org/10.1016/j.measurement.2019.03.066.

Biological colonization

Title	Biological colonization
Types of damage	Mould, fungus, lichens, plants, brown rot, white rot
Short description	Growth and establishment of various organisms on surfaces or within materials, leading to visible changes in appearance, texture, or structural integrity. These organisms can include fungi, algae, cyanobacteria, or plants, which may form colonies, cushions, or vegetation. Colonization can result in discoloration, deterioration, or degradation of the substrate, impacting both natural and man-made materials.

https://elrobledegarciaz.blogspot.com/2014/10/portera.html

https://www.pgdiag.fr/images/stories/pgdiag/agentsdedgradation.pdf

https://es.123rf.com/imagenes-de-archivo/moho.html

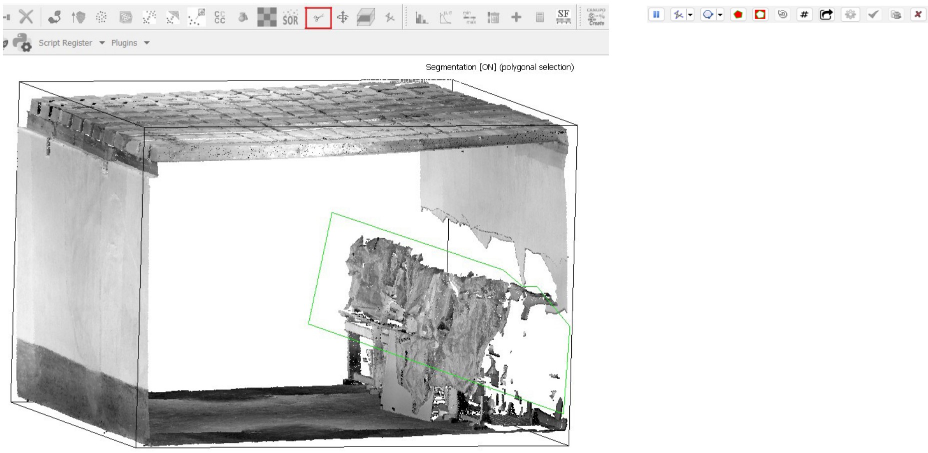
Interrelations between damages	Soiling Discolouration Material loss Detachment Cracks
Type of strategy to detect the damage	G-P2PRD, G-GF, G-P2PD, R-MM, R-TS
Some references of interest	[1] L.J. Sánchez-Aparicio, F.L. del Blanco-García, D. Mencías-Carrizosa, P. Villanueva-Llauradó, J.R. Aira-Zunzunegui, D. Sanz-Arauz, R. Pierdicca, J. Pinilla-Melo, J. Garcia-Gago, Detection of damage in heritage constructions based on 3D point clouds. A systematic review, Journal of Building Engineering 77 (2023). https://doi.org/10.1016/J.JOBE.2023.107440. [2] C. Suchocki, M. Damiecka-Suchocka, J. Katzer, J. Janicka, J. Rapi´nski, P. Stałowska, Remote detection of moisture and bio-deterioration of building walls by time-of-flight and phase-shift terrestrial laser scanners, Rem. Sens. 12 (2020) 1708, https://doi.org/10.3390/RS12111708, 2020 Vol 12 Page 1708. [3] R.L. Wood, M.E. Mohammadi, Feature-based point cloud-based assessment of heritage structures for nondestructive and noncontact surface damage detection, Heritage 4 (2021) 775–793, https://doi.org/10.3390/HERITAGE4020043. [4] J.S. Pozo-Antonio, I. Puente, M.F.C. Pereira, C.S.A. Rocha, Quantification and mapping of deterioration patterns on granite surfaces by means of mobile LiDAR data, Meas. J. Int. Meas. Confed. 140 (2019) 227–236, https://doi.org/10.1016/j.measurement.2019.03.066. [5] S. Del Pozo, J. Herrero-Pascual, B. Felipe-García, D. Hernández-López, P. Rodríguez-Gonzálvez, D. González-Aguilera, Multi-sensor radiometric study to detect pathologies in historical buildings, in: Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. - ISPRS Arch., International Society for Photogrammetry and Remote Sensing, Avila, Spain, 2015, pp. 193–200, https://doi.org/10.5194/isprsarchives-XL-5-W4-193-2015.

Pre-processing

Cleaning the point cloud
First of all, you will have to remove from your point cloud everything that is not «architectural», e.g. people, furniture, trees, etc. To do this, you can use the «scissors» tool to cut out everything you are not interested in.

Workflow:

1. Edit
2. Segment
3. Inside
4. Offside
5. Cut



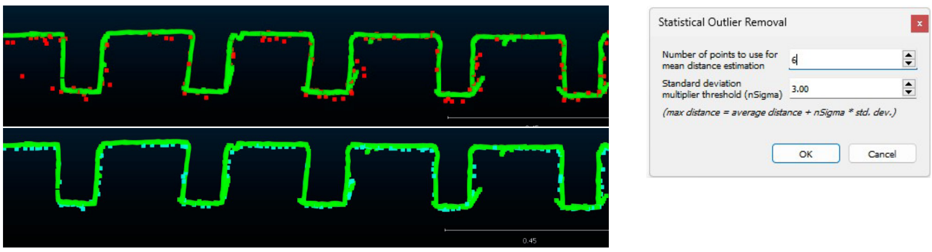
1.2. *Use of «scissors» tool.*

Then, to leave the point cloud cleaner, we recommend that you use the S.O.R. (Statiscal Outlier Removal) tool to remove all the scattered points that do not correspond to your point cloud.

Recommended parameters:
- *Number of points:* 6
- *Standard deviation:* 3

Workflow:

1. Tools
2. Clean
3. S.O.R. filter



1.3. *Use of «S.O.R.» tool.*

Segmentation
In the next step you will have to segment your point cloud by construction systems using the «Construction system segmentation» tab in order to make it possible to detect the damage at a later stage.

Features Computation
Finally, you will need to compute the features of your point cloud, which will give it the geometric or statistical information necessary for the artificial intelligence to work properly. Features are computed on a local basis. To compute them you can already do it in the software, in the first section «Feature computation».

Geometrical features

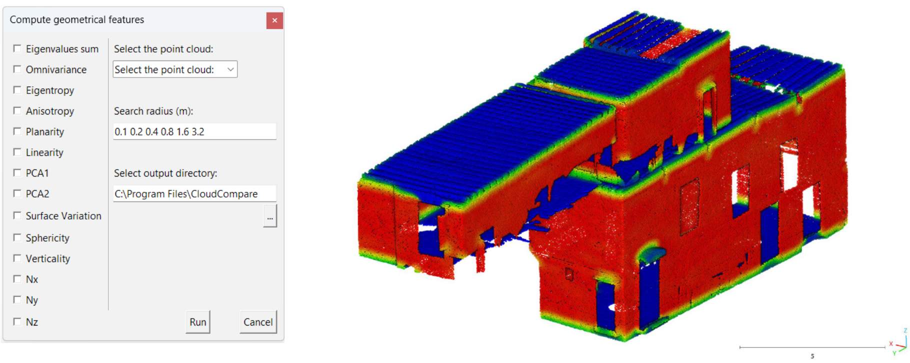
This tool allows you to compute several geometric features on one or several point clouds (in a batch). Geometric features that can be computed this way are: Sum of eigenvalues, Omnivariance, Eigenentropy, Anisotropy, Planarity, Linearity, PCA 1, PCA 2, Surface variation, Sphericity, Verticality, Nx, Ny and Nz.

Firstly, choose the geometric feature(s) that should be computed (several features can be selected/checked at the same time).

Then set the 'Radius search' on which the selected features will be computed. In this step you can compute all the features selected with one or more search radius. If you want to compute with more radius search use "Spaces" to separate each radius search. Then choose an output directory and execute the algorithm with "Run".

Statistical features

This tool, as well as the geometrical features extraction, allows you to compute several textural features on one or several point clouds (in a batch). The behaviour of certain statistical measures provides information on the distribution and characteristics of the data, thus enabling the definition of certain areas with insufficient content to be improved. Stadistical features that can be computed this way are:



Workflow:

1. Select point cloud

2. Select geom. features

3. Select radius search

4. Select output

1.4. Computing geometrical features to point cloud

- *Mean value:* Represents the trend by the sum of all neighbouring point values divided by the total number.

- *Standard deviation:* Measures the dispersion of the points around the mean from the average of the squared differences between each point and the mean of its neighbours.

- *Range:* Distance between the maximum value and the minimum value of the scalar.

- *Energy:* Statistical measure of randomness in the population of scalar fields.

- *Entropy:* It is also known as the angular second moment or uniformity.

- *Kurtosis:* This variable Measures the stability of the distribution, which relates to the normal distribution of the scalar values.

- *Skewness:* It is a measure of the degree of asymmetry in the distribution. Positive assimmetry indicates a distribution with a long right tail, while negative skewness indicates a long-left tail.

- *Skewness with binary values:* Is a feature which is «1» if the skewness is possitive and «0» if is negative.- *Skewness with binary values:* Positive and negative values.

Firstly, choose the textural feature(s) that should be computed (several features can be selected/checked at the same time).

Then set the 'Radius search' on which the selected features will be computed. In this step you can compute all the features selected with one or more search radius. If you want to compute with more radius search use "Spaces" to separate each radius search. Then choose an output directory and execute the algorithm with "Run".

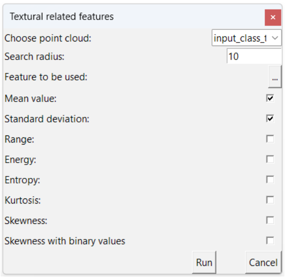
Workflow:

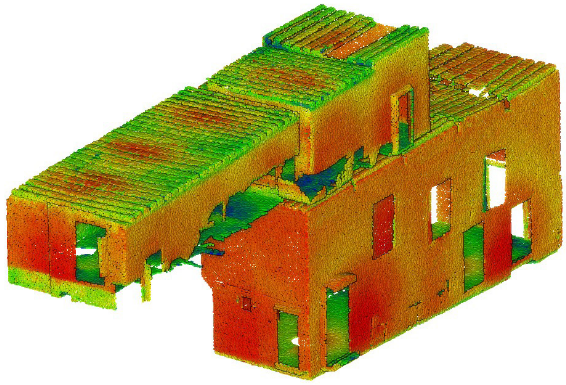
1. Select point cloud

2. Select stad. features

3. Select radius search

4. Select output





1.5. Computing stadistical features from point cloud

Color Conversion

This function simply converts the channel R,G & B into another cromatic combination. You can choose between these options:

- *HSV*: Hue Saturation Value

- *YCbCr*: «Y» represents the luma component and the «Cb» and «Cr» signals are the blue difference and red difference chrominance components, respectively.

- *YIQ*: «Y» represents the luminance information, «I» and «Q» represent the chrominance information, orange-blue and purple-green range respectively.

- *YUV*: «Y» represents the luminance information, «U» and «V» represent the chrominance information, red and blue range respectively.

Machine Learning

Supervised machine learning means that you will need a training model for it to work properly. In contrast, unsupervised does not require a training model. For supervised, it is advisable to divide the point cloud in such a way that 25% is for training and the remaining is for testing. Take both point clouds in the same folder.

Supervised Machine Learning

A) The main function of Feature selection tab is to select the best features for the next step, Classification. You must have the features computated in the Scalar Fields of the point cloud before applying the feature selection. You can compute the features in the "Features Computation" section.

In this tab, you can choose the point cloud, apply the selected selectors, choose the features that you want to work with, choose the number of features that you want to obtain, the number of folds for cross-validation and choose output directory.

- *Selectors*: Current version's default available selectors are ['kbest_f','rfe_lr','rfe_svm','rfe_tree','rfe_rf','rfecv_svm','rfecv_tree','rfecv_rf']. (NOTE: SVM based selectors are highly sensitive to the number of features(high-dimension) and training records number, i.e.rfe_svm and rfecv_svm. When features number > 50 w/ records number over 50K, better exclude these 2 selectors, otherwise will result in long processing time.)

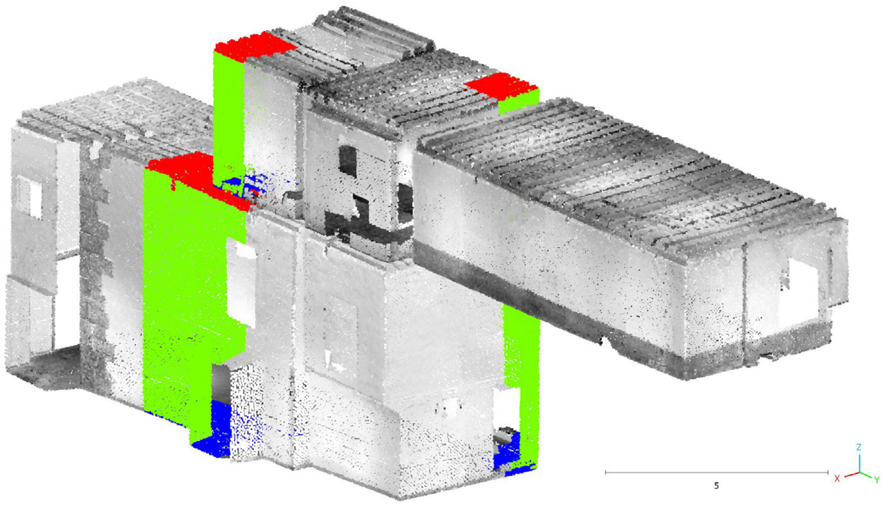
- *Number of features to consider*: Set the number of features want to select out

- *Folds for cross-validation*: Number of folds for cross-validation.

The result is a Document File (.txt) with the names of the features selected.

[1] https://optimal-flow.readthedocs.io/en/latest/autoFS.html#dynafs-clf.

B) In the classification tab, firstly you have to divide the point cloud into training (25%) and testing (75%). Make sure that all classes are present in both training and testing.



Workflow:

1. Select point cloud

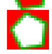
2. Viewing Tools

2.1. Set Top View 

3. Edit

3.1. Segment 

3.2. Inside 

3.3.Offside 

3.4. Cut 

1.6. Point cloud division (training and testing)

This tab works with different algorithms:

Random Forest

Parameters:

- *The number of trees in the forest*.

- *The function to measure the quality of a split:* Supported criteria are "gini" for the Gini impurity and "log_loss" and "entropy" both for the Shannon information gain, see Mathematical formulation. Note: This parameter is tree-specific.

-*The maximum depth of the tree:* If None, then nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples.

[2] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

- *The minimum number of samples required to split an internal node:* If int, then consider min_samples_split as the minimum number. If float, then min_samples_split is a fraction and ceil(min_samples_split * n_samples) are the minimum number of samples for each split.

- *The minimum number of samples required to be at a leaf node:* A split point at any depth will only be considered if it leaves at least min_samples_leaf training samples in each of the left and right branches.

- *The minimum weighted fraction of the sum total of weights (of all the input samples) required to be at a leaf node:* Samples have equal weight when sample_weight is not provided.

-*The number of features to consider when looking for the best split:* Note: the search for a split does not stop until at least one valid partition of the node samples is found, even if it requires to effectively inspect more than max_features features.

- *Maximum number of leaf nodes:* Grow trees with max_leaf_nodes in best-first fashion. Best nodes are defined as relative reduction in impurity. If None then unlimited number of leaf nodes.

- *Impurity to split the node:* A node will be split if this split induces a decrease of the impurity greater than or equal to this value.

- *Use of bootstrap*: Whether bootstrap samples are used when building trees. If False, the whole dataset is used to build each tree.

- *Complexity parameter used for Minimal Cost:* The subtree with the largest cost complexity that is smaller than ccp_alpha will be chosen. By default, no pruning is performed.

Support Vector Machine

[3] https://scikit-learn.org/stable/modules/svm.html

Parameters:

- *Regularization parameter:* The strength of the regularization is inversely proportional to C. Must be strictly positive. The penalty is a squared l2 penalty.

- *Kernel type:* Specifies the kernel type to be used in the algorithm. If none is given, 'rbf' will be used. If a callable is given it is used to pre-compute the kernel matrix from data matrices; that matrix should be an array of shape (n_samples, n_samples).

- *Degree of the polynomial kernel function:* Must be non-negative. Ignored by all other kernels.

- *Kernel coefficient:* If gamma='scale' (default) is passed then it uses 1 / (n_features * X.var()) as value of gamma, if 'auto', uses 1 / n_features and if float, must be non-negative.

- *Independent term in kernel function:* It is only significant in 'poly' and 'sigmoid'.

- *Use shrinking heuristics:* If the number of iterations is large, then shrinking can shorten the training time.

- *Enable probability estimates:* This must be enabled prior to calling fit, will slow down that method as it internally uses 5-fold cross-validation, and predict_proba may be inconsistent with predict.

- *Tolerance for stopping criterion.*

- *Class weight:* Set the parameter C of class i to class_weight[i]*C for SVC. If not given, all classes are supposed to have weight one.

- *Max number of iterations:* Hard limit on iterations within solver, or -1 for no limit.

- *Returned function:* Whether to return a one-vs-rest ('ovr') decision function of shape (n_samples, n_classes) as all other classifiers, or the original one-vs-one ('ovo') decision function of libsvm which has shape (n_samples, n_classes * (n_classes - 1) / 2).

- *Break ties:* If true, decision_function_shape='ovr', and number of classes > 2, predict will break ties according to the confidence values of decision_function; otherwise the first class among the tied classes is returned.

Logistic Regression

[4] https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

Parameters:

- *Penalty function*: None: no penalty is added; 'l2': add a L2 penalty term and it is the default choice; 'l1': add a L1 penalty term; 'elasticnet': both L1 and L2 penalty terms are added.

- *Constrained problem (dual):* Dual formulation is only implemented for l2 penalty with liblinear solver. Prefer dual=False when n_samples > n_features.

- *Tolerance for stopping criterion.*

- *Inverse of regularization strength:* Must be a positive number. Like in support vector machines, smaller values specify stronger regularization.

- *Add a bias to the model:* Specifies if a constant should be added to the decision function.

- *Intercept scaling:* In this case, x becomes [x, self.intercept_scaling], i.e. a "synthetic" feature with constant value equal to intercept_scaling is appended to the instance vector.

- *Class weight:* If not given, all classes are supposed to have weight one.

- *Type of solver:* Algorithm to use in the optimization problem. Default is 'lbfgs'. To choose a solver, you might want to consider the following aspects: For small datasets, 'liblinear' is a good choice, whereas 'sag' and 'saga' are faster for large ones; For multiclass problems, only 'newton-cg', 'sag', 'saga' and 'lbfgs' handle multinomial loss; 'liblinear' is limited to one-versus-rest schemes; 'newton-cholesky' is a good choice for n_samples >> n_features, especially with one-hot encoded categorical features with rare categories.

- *Max number of iterations:* Maximum number of iterations taken for the solvers to converge.

- *Type of multiclass fitting strategy:* If the option chosen is 'ovr', then a binary problem is fit for each label. For 'multinomial' the loss minimised is the multinomial loss fit across the entire probability distribution, even when the data is binary. 'multinomial' is unavailable when solver='liblinear'. 'auto' selects 'ovr' if the data is binary, or if solver='liblinear', and otherwise selects 'multinomial'.

- *Elastic-Net mixing parameter:* With 0 <= l1_ratio <= 1. Only used if penalty='elasticnet'. Setting l1_ratio=0 is equivalent to using penalty='l2', while setting l1_ratio=1 is equivalent to using penalty='l1'. For 0 < l1_ratio <1, the penalty is a combination of L1 and L2.

- *Number of cores to use:* Number of CPU cores used when parallelizing over classes if multi_class='ovr'". This parameter is ignored when the solver is set to 'liblinear' regardless of whether 'multi_class' is specified or not. -1 means using all processors.

[5] https://epistasislab.github.io/tpot/

Auto Machine Learning

It helps you out with the most tedious part of machine learning by intelligently exploring thousands of possible pipelines to find the best one for your data.
Parameters:

- *Number of generations:* Number of iterations to the run pipeline optimization process. It must be a positive number or None. If None, the parameter "Maximum total time" must be defined as the runtime limit. Generally, it will work better when you give it more generations (and therefore time) to optimize the pipeline.

-*Population size:* Number of individuals to retain in the genetic programming population every generation. Must be a positive number. Generally, it will work better when you give it more individuals with which to optimize the pipeline.

- *Mutation rate:* Mutation rate for the genetic programming algorithm in the range [0.0, 1.0]. This parameter tells the GP algorithm how many pipelines to apply random changes to every generation. Mutation rate + crossover rate cannot exceed 1.0. We recommend using the default parameter unless you understand how the mutation rate affects GP algorithms.

- *Crossover rate:* This parameter tells the genetic programming algorithm how many pipelines to «breed» every generation.

- *Function used to evaluate the quality:* See the section on scoring functions for more details.

- *Cross-validation*: Cross-validation strategy used when evaluating pipelines.

- *Maximum total time*: How many minutes TPOT has to optimize the pipeline. If not None, this setting will allow TPOT to run until "Maximum total time" minutes elapsed and then stop. It will stop earlier if generations is set and all generations are already evaluated.

- *Maximum time per evaluation*: How many minutes TPOT has to evaluate a single pipeline. Setting this parameter to higher values will allow TPOT to evaluate more complex pipelines, but will also allow TPOT to run longer. Use this parameter to help prevent TPOT from wasting time on evaluating time-consuming pipelines.

- *Number of generations without improvement*: Ends the optimization process if there is no improvement in the given number of generations.

C) Prediction tab consists of predicting the classification with the previous training file. In the previous step, it launched a PKL file with the training configuration, which is necessary for this step. This algorithm works with Scikit-Learn library.

Is mandatory to send in the buttons the required information: firstly, you have to choose the point cloud to predict, then select the feature and the PKL file launched in the classification step. Then choose an output directory and execute the algorithm with "Run".

Unsupervised Machine Learning

A) Training tab works with different algorithms:

K-Means

Parameters:

- *Number of clusters:* The number of clusters to form as well as the number of centroids to generate.

- *Number of iterations:* Maximum number of iterations of the k-means algorithm for a single run.

- *Optimization strategy*: Due to the fact that the optimal number of clusters is not known in advance, by selecting this option, it recommends the optimal number of clusters. The available options are: elbow method and silhouete method, Calinski Harabasz, Davies Bouldin.

Elbow method: The KElbowVisualizer implements the "elbow" method to help data scientists select the optimal number of clusters by fitting the model with a range of values for "K". https://www.scikit-yb.org/en/latest/api/cluster/elbow.html

Silhouete Coefficient: Is used when the ground-truth about the dataset is unknown and computes the density of clusters computed by the model. https://www.scikit-yb.org/en/latest/api/cluster/silhouette.html

Calinski Harabasz index: It is also known as the Variance Ratio Criterion. The score is defined as ratio of the sum of between-cluster dispersion and of within-cluster dispersion. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.calinski_harabasz_score.html#sklearn.metrics.calinski_harabasz_score

Davies Bouldin index: The score is defined as the average similarity measure of each cluster with its most similar cluster, where similarity is the ratio of within-cluster distances to between-cluster distances. Thus, clusters which are farther apart and less dispersed will result in a better score. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.davies_bouldin_score.html

- *Maximum number of clusters:* Depending on the choice of the optimization strategy, set a maximum number of clusters.

- *Minimum number of clusters:* Depending on the choice of the optimization strategy, set a minimum number of clusters.

Fuzzy-K-Means

Parameters:

The parameters are the same as K-Means (previous).

Hierarchical Clustering

[6] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

[7] https://fda.readthedocs.io/en/latest/modules/ml/autosummary/skfda.ml.clustering.FuzzyCMeans.html

[8] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

Parameters:

- *Number of clusters:* The number of clusters to find. It must be None if distance threshold is not None.

- *Metric for calculating the distance between instances:* Can be "euclidean", "l1", "l2", "manhattan", "cosine", or "precomputed". If linkage is "ward", only "euclidean" is accepted. If "precomputed", a distance matrix is needed as input for the fit method.

- *Linkage criterion:* The linkage criterion determines which distance to use between sets of observation. The algorithm will merge the pairs of cluster that minimize this criterion.

'ward' minimizes the variance of the clusters being merged.

'average' uses the average of the distances of each observation of the two sets.

'complete' or 'maximum' linkage uses the maximum distances between all observations of the two sets.

'single' uses the minimum of the distances between all observations of the two sets.

- *Linkage distance threshold*: The linkage distance threshold at or above which clusters will not be merged. If not None, number of clusters must be None and compute full tree must be True.

- *Computes distances between clusters:* Even if distance threshold is not used. This can be used to make dendrogram visualization, but introduces a computational and memory overhead.

[9] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

DBSCAN

Parameters:

- *Epsilon*: The maximum distance between two samples for one to be considered as in the neighborhood of the other. This is not a maximum bound on the distances of points within a cluster.

- *Minimum number of points to create a cluster*: The number of samples (or total weight) in a neighborhood for a point to be considered as a core point. This includes the point itself. If min samples is set to a higher value, DBSCAN will find denser clusters, whereas if it is set to a lower value, the found clusters will be sparser.

[10] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.OPTICS.html

OPTICS

Parameters:

- *The number of samples in a neighborhood to be considered as cluster:* Also, up and down steep regions can't have more than min_samples consecutive non-steep points.

- *Epsilon:* The maximum distance between two samples for one to be considered as in the neighborhood of the other.

- *Metric for distance computation*: Any metric from scikit-learn or scipy.spatial.distance can be used. If metric is a callable function, it is called on each pair of instances (rows) and the resulting value recorded. See the documentation for scipy.spatial.distance for details on these metrics.

- *Extraction method:* Used to extract clusters using the calculated reachability and ordering. Possible values are "xi" and "dbscan".

- *Minimum steepness:* Determines the minimum steepness on the reachability plot that constitutes a cluster boundary. For example, an upwards point in the reachability plot is defined by the ratio from one point to its successor being at most 1-xi. Used only when extraction method='xi'.

- *Minimum cluster size:* Minimum number of samples in an OPTICS cluster, expressed as an absolute number or a fraction of the number of samples (rounded to be at least 2). If None, the value of min_samples are used instead. Used only when cluster_method='xi'.

B) Prediction tab consists of predicting the classification with the previous training file. In the previous step, it launched a PKL file with the training configuration, which is necessary for this step. This algorithm works with Scikit-Learn library.

It is mandatory to send in the buttons the required information:

Firstly, you have to choose the point cloud to predict, then select the feature and the PKL file launched in the classification step. Then choose an output directory and execute the algorithm with "Run".

Results
For all the Machine Learning algorithms used, different graphs representing the values of the results obtained will be generated in the Output directory.

- *Classification matrix*: This matrix represents the level of accuracy and precision of the algorithm training. The above part represents it by levels of classification, and under it, the global values in percentage. Parameters:

Accuracy: Accuracy is a measure of the overall correctness of a model. It calculates the ratio of correctly predicted instances to the total instances.

Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. It measures the accuracy of positive predictions.

Recall: Recall is the ratio of correctly predicted positive observations to all observations in actual class. It measures the ability of the model to capture all the positive instances.

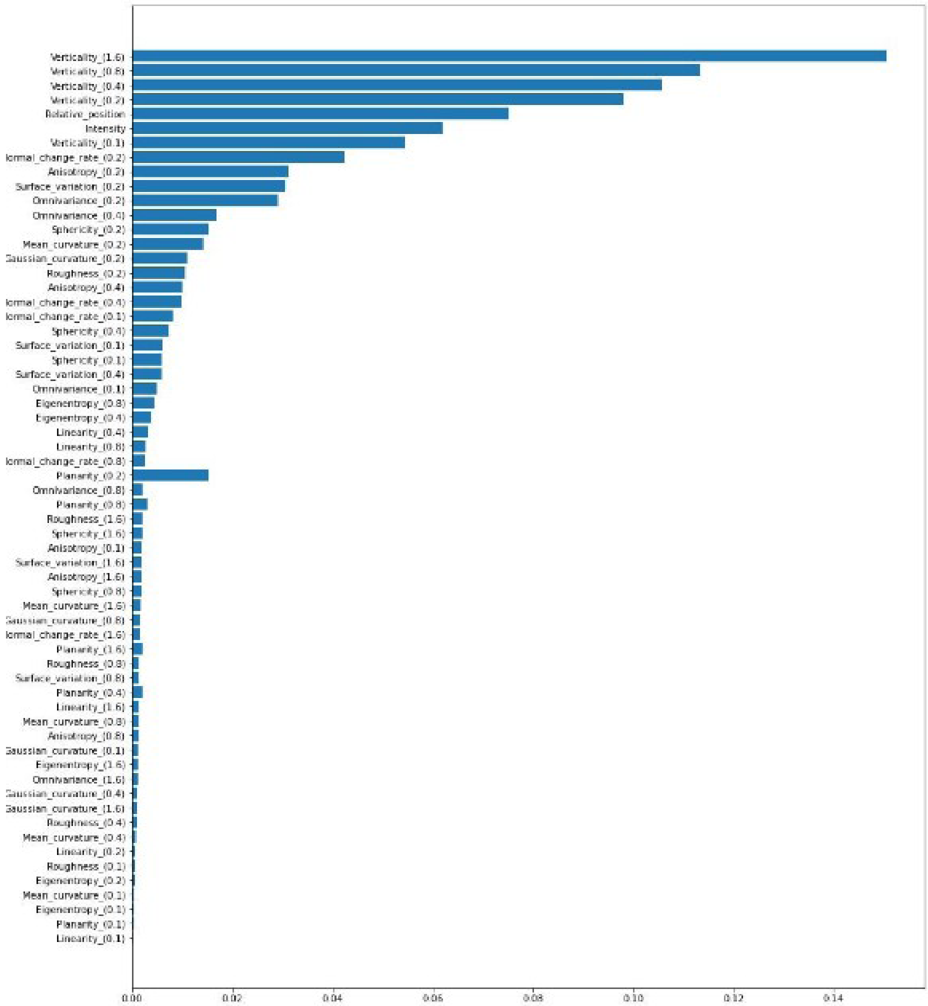
F1 Score: The F1 Score is the harmonic means of precision and recall. It provides a balance between precision and recall.

-*Importance* (only in Random Forest algorithm): This graph represents the importance of the features by scale, the higher a feature is, the more important it has been for the classification.

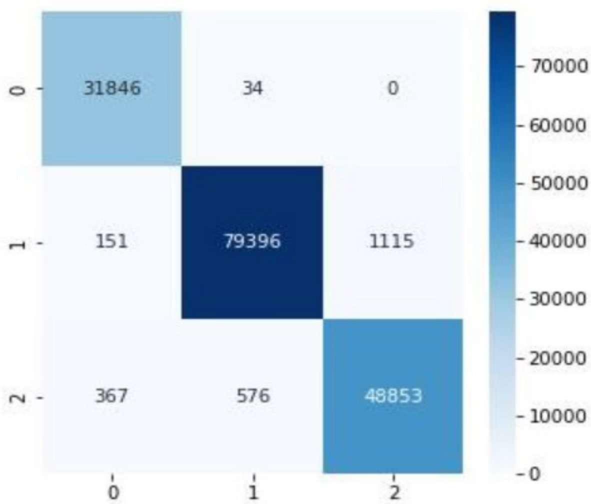
- *Confusion matrix:* This matrix represents the number of points of a level that has been confused with another level.

	precision	recall	f1-score	support
0.0	0.984	0.999	0.991	31880
1.0	0.992	0.984	0.988	80662
2.0	0.978	0.981	0.979	49796
accuracy			0.986	162338
macro avg	0.985	0.988	0.986	162338
weighted avg	0.986	0.986	0.986	162338

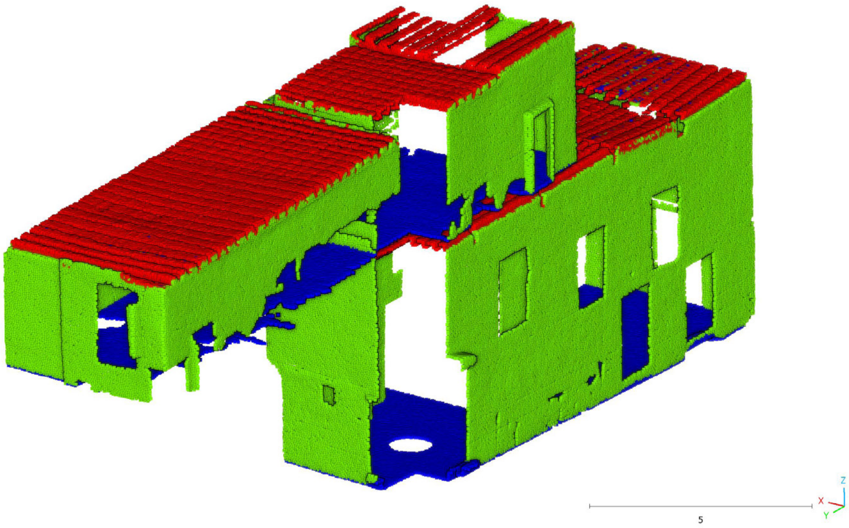
1.7. Classification matrix



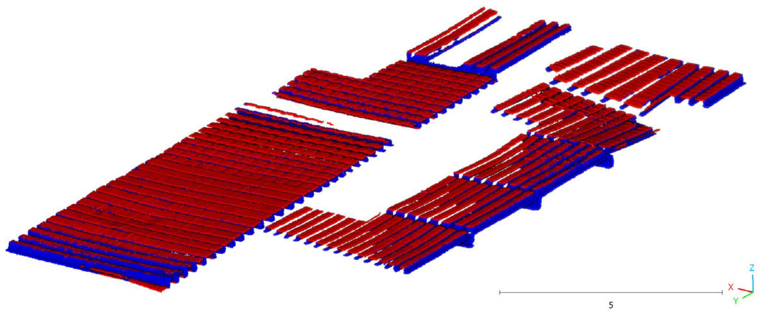
1.8. Importance



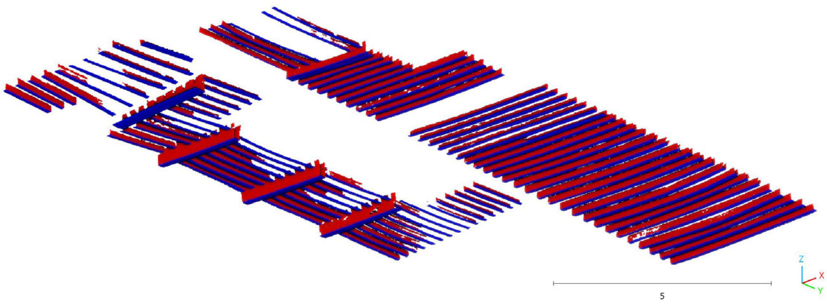
1.9. Confusion matrix



1.10. Classification result of the first level of classification



1.11. Classification result of the second level of classification



1.12. Classification result of the third level of classification

Deep Learning
Deep Learning works with Point Transformer algorithm.
IMPORTANT: **The first numbered class in the point cloud must start with 0, the next 1 and so on for the rest of the classes.**

A) The Training tab consists of, as well as Machine Learning, with a training and testing point cloud. First you have to choose them correctly (it is recommended to take 25% of the point cloud for training and 75% of the rest for testing). Then, as well as Machine Learning, choose the features you want to add to the training. GPU consumption means how much power from your computer you want to use. Choose the number of iterations to the run pipeline optimization process (it must be a positive number or None). Then choose an output directory and execute the algorithm with “Run”.

B) The Classification tab is the next step after the Training. In the previous step, it launched a PKL file with the training configuration, which is necessary for this step. Firstly, you have to choose the point cloud to classify, then select the feature and the PKL file launched in the classification step. Then choose a GPU consumption and an output directory and execute the algorithm with “Run”.

The results will be in the output folder.

Arches and vaults

This tool offers you the possibility to extract the deformations of arches and vaults by means of the analysis of a number of sections.

Analysis of arches

This algorithm allows to extract the longitudinal section of different arches and to estimate the best fitting arch based on the RANSAC method (RANdom SAmple Consensus). The algorithm will calculate this arc in each of the elements of the selected folder.

Parameters:

- *Thickness threshold:* Is the thickness of points to consider from the arc axis to estimate the calculation section.

- *Type of arch:* Is the type of arch to be estimated by the RANSAC method. Type of arch:

Circular arch
Pointed arch
Quarter arch

- *Threshold value for RANSAC fitting:* Is the maximum distance from a point to the best fitting arc below which the point is considered to belong to that arc and above which it is considered outlier.

- *Number of iterations for RANSAC fitting:* Is the maximum number of iterations to be used by RANSAC.

- *Minimum number of samples for fitting the model*

- *Fixed springing line:* Constraint that makes the arc conform to its starts. These starts are defined as a percentage of the total arc height.

- *Percentage of points to fit the curve from springs:* The percentage of the total height of the arc.

- *Load the points of the main section:* Option that allows to load the points considered to estimate the section.

- *Path for saving the data:* Path for saving the data. A section is saved with the points considered as inliers and outliers as well as the arc of best fit.

Analysis of vaults

Method that allows in first instance to extract the longitudinal axis of the vault and then to obtain sections that are analysed by the process «Analyse arches».

Parameters:

- *Choose the point clouds:* Point cloud used for the analysis.

- *Type of vault:* Types:

  Barrel vault

- *Space between sections:* Spacing between consecutive sections extracted perpendicular to the main axis of the vault.
- *Thickness of each section:* Is the thickness of points to be considered from the axis of the arch to estimate the calculation section.
- *Maximum z value for extracting the axis:* Is the maximum value in Z below which the points of the vault are considered as points for the calculation of its axis.
- *Minimum z value for extracting the axis:* Is the minimum Z value above which the points of the vault are considered as points for the calculation of its axis.
- *Smoothing parameter for the axis extraction:* Is the degree of smoothing used in the estimation of the main axis of the vault.
- *Expected clearance of the vault:* Is the approximate clearance of the vault.
- *Configuration of the arch estimator:* Same parameters as for «Analyse arches».
- *Path for saving the data:* Saves sections and best-fit arcs of each section along the main axis of the vault.

Analysis of deflection

This algorithm allows the analysis of deflections in beams. A folder containing the bottom face of the beams must be sectioned.
Basically, the algorithm calculates the main axes of each beam with the Minimum Bouding Box algorithm and then extracts a longitudinal section on which a fitting polynomial, the inflection points and maximum deflection are calculated.

Parameters:

- *Thickness threshold:* Is the thickness, from the central axis of the beam within which the points will be taken to calculate the polynomial defining the deflection.
- *Polynomic degree:* Is the degree of adjustment of the polynomial.
- *Maximum relative deflection (L/300; L/500):* The relative deflection limit. Above the limit a scalar of excess deflection will be included which will be true if the threshold is exceeded.
- *Type of input for the scalar field:* Is the type of input to be used for the scalar fields of the results. Type:

  Data: Extracted directly from the point cloud.

  Fit: Extracted from the fitted polynomial.

- *Load the points of the main axis:* If true the program will load the main axes of the beam.

Path for saving data: Place to save the outputs. These outputs are txt files with the calculated data and plots of the points of the main axis and the fitting polynomial, as well as the points of maximum deflection and inflection.

Analysis of inclination

It is necessary to select a folder with the elements separated by instances. The algorithm will calculate the inclination of each element.

Basically the algorithm makes a series of z-slices (with a specific thickness), and calculates the centre of gravity which is then used to estimate the slope.

Parameters:

- *Thickness threshold:* Is the thickness of each of these sections. More thickness means more points, but can also make the determination of the slope more inaccurate.

- *Step between sections:* Separation between consecutive sections.

- *Maximum inclination allowed:* Maximum inclination in degrees. So that if this inclination is exceeded, the element will have a scale with an excess of inclination in real terms.

- *Type strategy used to compute the center of gravity:* This parameter is used to determine the point of gravity of each section. The centre of gravity would be used to calculate the tilt through a straight line adjustment on each of the axes. The types are:

Points: Takes as input for the calculation of the centre of gravity the points themselves. It is inaccurate if it has notable data absences.

Convex hull: It estimates the centre of gravity of the polygon defined by the points of each section with the Convex Hull algorithm.

Min Bounding Rectangle: Estimate the minimum rectangle containing the points and then determine its centre of gravity. Accurate if you have a large percentage of the section captured, otherwise it is imprecise.

Circle fitting: The circle is fitted to the set of points in the section. This option is accurate even with missing data.

- *Path for saving the data:* The outputs are txt files with the calculated data and graphs showing the inclination in the x and y axis as well as a folder with the results of the centre of gravity estimation.

BIM Integration

This tool will be coming soon.