Automatic training data generation in Deep Learning-aided semantic segmentation of Heritage buildings

Arnadi Murtiyoso, Francesca Matrone, Massimo Martini, Andrea Lingua, Pierre Grussenmeyer, Roberto Pierdicca

AUTOMATIC TRAINING DATA GENERATION IN DEEP LEARNING-AIDED SEMANTIC SEGMENTATION OF HERITAGE BUILDINGS

A. Murtiyoso[1*], F. Matrone[2], M. Martini[3], A. Lingua[2], P. Grussenmeyer[4], R. Pierdicca[5]

[1]Forest Resources Management Group, Institute of Terrestrial Ecosystems, Department of Environmental Systems Science, ETH Zürich, Switzerland - arnadidhestaratri.murtiyoso@usys.eth.ch
[2]Department of Environment, Land and Infrastructure Engineering, Politecnico di Torino, Torino, Italy - (francesca.matrone, andrea.lingua)@polito.it
[3]Department of Information Engineering, Università Politecnica delle Marche, Ancona, Italy - m.martini@pm.univpm.it
[4]Université de Strasbourg, CNRS, INSA Strasbourg, ICube Laboratory UMR 7357, Photogrammetry and Geomatics Group, Strasbourg, France - pierre.grussenmeyer@insa-strasbourg.fr
[5]Department of Civil and Building Engineering and Architecture, Università Politecnica delle Marche, Ancona, Italy - r.pierdicca@staff.univpm.it

Commission II, WG II/8

KEY WORDS: Training data, Automation, Deep learning, Point cloud, Heritage, Semantic segmentation.

ABSTRACT:

In the geomatics domain the use of deep learning, a subset of machine learning, is becoming more and more widespread. In this context, the 3D semantic segmentation of heritage point clouds presents an interesting and promising approach for modelling automation, in light of the heterogeneous nature of historical building styles and features. However, this heterogeneity also presents an obstacle in terms of generating the training data for use in deep learning, hitherto performed largely manually. The current generally low availability of labelled data also presents a motivation to aid the process of training data generation. In this paper, we propose the use of approaches based on geometric rules to automate to a certain degree this task. One object class will be discussed in this paper, namely the pillars class. Results show that the approach managed to extract pillars with satisfactory quality (98.5% of correctly detected pillars with the proposed algorithm). Tests were also performed to use the outputs in a deep learning segmentation setting, with a favourable outcome in terms of reducing the overall labelling time (-66.5%). Certain particularities were nevertheless observed, which also influence the result of the deep learning segmentation.

1. INTRODUCTION

As the documentation of heritage objects is undertaken more and more in 3D, point cloud data has become ubiquitous in the heritage community. With the advent of laser scanners and advanced photogrammetric processing, the documentation process is becoming more and more streamlined. The next issue of interest in the point cloud processing community is how to annotate the geometric point cloud with the addition of semantic attributes. This is required when the point cloud is needed for analysis, modelling, and predictions. A semantically annotated point cloud can thereafter be used to create information-rich 3D GIS and/or HBIM (Heritage Building Information Models) (Campanaro et al., 2016).

Machine learning (ML), and more precisely deep learning (DL) techniques, has witnessed a surge in overall interest in this age of big data (Bello et al., 2020). The possibility to use large quantities of data to train the computer to perform semantic segmentation automatically is indeed a very interesting concept and currently a promising field of research, as it provides a robust segmentation result with quick processing time. However, the main bottleneck problem in implementing DL techniques is mainly related to the availability of labelled datasets (Maalek et al., 2019). In the case of heritage point cloud, this problem is exacerbated by the diversity of classes and architectural features, as well as the general lack of labelled datasets. As such, the usual way to obtain training data is by manual annotation (Malinverni et al., 2019).

Considering these issues, a combination of more traditional segmentation based on geometric axioms and DL techniques will be presented in this paper. In particular, deep learning techniques will allow validating objects segmented by traditional methods. The algorithmic segmentation uses the functions in the Matlab toolbox M_HERACLES to generate training data for the DL technique. The toolbox is a set of Matlab functions (https://github.com/murtiad/M_HERACLES, accessed 4 October 2021) specifically developed to perform semantic segmentation on heritage objects (Murtiyoso and Grussenmeyer, 2020).

While the geometric approach may be used to directly generate segmented point cloud, many limitations are inherent in this method. Indeed, this type of approach mainly uses case-specific hard-coded prior knowledge and geometric rules in each function, therefore limiting its use for a holistic semantic segmentation. This rigidity is however counterbalanced by rapid processing time and a straightforward and open nature as opposed to the more closed system of DL techniques. As such, we postulate that both geometric rules-based and DL methods have their own advantages and disadvantages which may play well to support each other.

The main idea proposed in this paper is therefore to use the two semantic segmentation approaches, namely the geometric rules-based and DL methods, to complement each other. Our proposed method therefore tries to automate as much as possible the semantic segmentation workflow in the case of

* Corresponding author

heritage buildings, from the generation of DL training data up to the use of DL itself for the abovementioned task. In this case, the DL framework is developed by testing the data on a specific neural network, namely the DGCNN (Dynamic Graph Convolutional Neural Network) (Wang et al., 2019), specially modified for the semantic segmentation in the field of cultural heritage (Matrone et al., 2020a; Pierdicca et al., 2020). M_HERACLES will be used to detect objects from the 3D scene (with the specific class of pillars is chosen for this study). Comparison of its results as opposed to manually labelled data will then be performed. Both the manual and automatic (i.e., resulting from M_HERACLES) results will then be used as input training data for our DGCNN framework. The results will then be compared in terms of processing time and quality. The paper does not aim to improve on the results of the neural network; rather, the main objective is to determine whether the use of geometric rules-aided automatic training data may help achieve similar results as manual annotation with expected gain in overall labelling time.

2. STATE OF THE ART

Semantic segmentation as a research topic stems as a logical consequence to the use of point clouds as 3D archive. The unorganised nature possessed by point clouds by default requires classification in order to permit a better understanding of the scene (Poux et al., 2016). In the architecture and civil engineering (ACE) domain, this has evolved into the need for a "scan-to-BIM" process with Building Information Models (BIM) as the final product (Macher et al., 2017; Xiong et al., 2013). As regards to heritage point clouds, semantic segmentation enables the otherwise purely geometric data to receive tangible semantic information (Murtiyoso and Grussenmeyer, 2020) and thus may ultimately aid the creation of Heritage Building Information Models (HBIM) (Chiabrando et al., 2016).

Various approaches to point cloud processing may be taken to perform this task. Grilli et al., (2017) categorised these approaches into region growing methods (Bassier et al., 2017a), edge-based reconstruction (Boulaassal et al., 2007), model-fitting (Sanchez and Zakhor, 2012), machine learning-based methods (Bassier et al., 2017b), and hybrid approaches. In an article by Nguyen and Le (2013), a dual distinction between segmentation by machine learning and by the use of geometric axioms was made, also called "constraints" or hard-coded knowledge. Furthermore, Bassier et al. (2017b) refer to the latter as heuristic approach in contrast to machine learning; in essence an algorithmic approach to the problem of 3D point classification. With the advent of big data, DL as a part of the machine learning approach is today considered as another viable approach (Matrone et al., 2020a). Two methodologies encountered in the literature for the semantic segmentation task of point clouds concern deep learning and rules-based approaches.

2.1 Deep learning for point cloud segmentation

Three-dimensional point clouds are currently used in many applications thanks to the recent wide availability of 3D scanners and reconstruction techniques such as lidar, Structure-from-Motion (SfM) and other sensors like Kinect and Xtion. These are 3D unstructured vectors that compute 3D coordinates and other characteristics such as reflection, colour and normal. One of the pioneering research that used deep learning to process raw point clouds is presented in Qi et al. (2017a), where the PointNet architecture is used to process point clouds for semantic segmentation and classification. One limitation of this approach is its inability to extract the local features of point clouds, learning only global features through the max-pooling layer. To overcome this problem, the PointNet++ architecture (Qi et al., 2017b) was developed to allow the encoding of local features by dividing locally the point cloud. The authors of the architecture propose essentially hierarchical feature learning for object classification and semantic segmentation of 3D point clouds.

A recent approach inspired by PointNet and PointNet++ is the Dynamic Graph Convolutional Neural Networks (DGCNN) (Wang et al., 2019). Unlike PointNet++, DGCNN extracts local features by using an EdgeConv layer. DGCNN builds a neighbourhood graph that allows exploiting the local geometric structures, defining a link between the central point chosen and the edge vector connecting its neighbours to itself. The main advantage is that DGCNN presents a robust result to the variation of the input to obtain satisfactory results in both indoor and outdoor scenes.

In the Digital Cultural Heritage (DCH) domain, the work of Pierdicca et al. (2020) attempted to exploit an improved DGCNN architecture in order to semantically segment 3D point clouds, which is both interesting and useful for the automatic interpretation of architectural objects. Furthermore, the authors proposed a novel approach using additional point cloud features. A completely new dataset involving both indoor and outdoor scenes was used, belonging to different historical periods and different styles. The dataset has been manually labelled by experts to increase its level of trustworthiness.

However, creating a huge amount of labelled point clouds through manual annotation is very time-consuming and often impractical. This issue becomes more exacerbated by the complex variations in architectural styles and details in the case of heritage sites. A possible solution to this problem is the creation of synthetic training datasets, even though this procedure is less common in the DCH domain (Pierdicca et al., 2019). Pellis et al. (2021) also proposed reprojection of 3D point cloud labelling into the 2D space of photogrammetric images in order to augment 2D semantic segmentation training data in a DCH context. Other recent approaches for the generation of new data include techniques based on generative models, in particular generative adversarial networks (GAN) (Goodfellow et al., 2016).

In this work the DGCNN-Mod (Pierdicca et al., 2020) network will be used to validate the scenes segmented by rules-based approach provided by M_HERACLES. This particular network was chosen primarily because it was recently implemented in the DCH domain and showed promising results (Malinverni et al., 2019). Furthermore, as an initial experiment and proof of concept this paper will focus mainly on the pillars class in some cultural heritage scenes.

2.2 Geometric rules-based approach to the problem of point classification

The rules-based approach, as has been mentioned previously, employs geometric rules and constraints to detect and classify certain architectural features from the point cloud scene. The rules are generally prior knowledge hard coded into the algorithm (Maalek et al., 2019) and may consist of simple axioms (Macher et al., 2016; Murtiyoso and Grussenmeyer, 2019a) to more complex ontological networks (Poux et al., 2018). This heuristic approach can be rapidly employed without the need for training and is thus adaptable for simpler cases (Nguyen and Le, 2013); however it may suffer from higher

noise rate and rigidity, which leads to its limited use when compared to other approaches based on ML/DL.

Several types of geometric axioms can be used to detect very specific object classes. For example, in Riveiro et al. (2016) the authors detected planar surfaces for wall detection in an outdoor setting. In Rodríguez-Cuenca et al. (2015), the authors attempted to detect pole-like objects from an unorganised point cloud. Poux et al. (2017) took advantage of other point cloud features to perform segmentation, while in Poux et al. (2018) and Drap et al. (2017) more complex ontological relations were designed. Murtiyoso and Grussenmeyer (2019b) used pre-existing GIS layers to perform a similar task in a smaller-scale data setup.

In this paper, a part of the toolbox M_HERACLES was used to perform semantic segmentation for the class of pillars, i.e., structural supports. Structural supports such as columns present a particular interest for the heritage community, as often they present a valuable example of historical engineering and architectural design. Several study have been done in the field of structural support automatic 3D modelling (Maalek et al., 2019), but most focus on simple pillars or supports. In this regard, automation for heritage-related structural support remains difficult due to the many different types linked to the architectural style. M_HERACLES was specifically conceived to tackle this problem using rules-based approaches (Murtiyoso and Grussenmeyer, 2020).

As has been previously established, this study will be focused on the semantic segmentation of pillars. M_HERACLES will therefore be used to detect and classify pillars from the available datasets. These automatically segmented scenes will be used thereafter to train the DGCNN-Mod for the semantic segmentation task and the results will be compared against those obtained from manual labelling.

3. METHODOLOGY

The tests were conducted on a small, labelled dataset of point clouds of cultural heritage to quantitatively compare the labelling time and the consequent results of the neural network with the two different inputs, namely the dataset manually annotated and the one automatically segmented via M_HERACLES. The toolbox performs segmentation in the case of pillars using geometrical rules, namely the circularity of the cross-sections. Moreover, the scenes of the dataset were divided into training, validation, and test. More specifically, we want to understand the behaviour of the neural network with respect to the two test scenes with completely different architectural styles, namely European and Southeast Asian.

Among the five selected case studies (Figure 1), three are in the European architectural style: the "Sala delle Colonne" of Valentino Palace in Turin, Italy (**"Valentino"**) and two point clouds from the Sacro Monte di Varallo site (**"Ghiffa"** and **"Pilato"**). These sites are all included in the UNESCO World Heritage List. Meanwhile, the Southeast Asian dataset comprises of two pavilions (**"Kasepuhan_1"** and **"Kasepuhan_2"**) from the Kasepuhan Palace, a 15th century royal complex in Cirebon, Indonesia. All of these point clouds are part of the public ArCH dataset (Matrone et al., 2020b) (http://archdataset.polito.it/, accessed 5 October 2021).

Two distinct experiments were performed for the purposes of this paper. In the first experiment, the automatic point cloud labelling using the M_HERACLES toolbox will be discussed. In order to quantitatively analyse the results, the results were compared against manually labelled data to assess the proposed rules-based method's performance in terms of statistical qualities and processing time. The main objective of this first experiment is to test M_HERACLES's reliability for automatic point cloud labelling.

The second experiment is the more elaborate of the two. In this experiment, a real application of M_HERACLES's results was fed into a DL network to test its reliability in terms of use for DL training data generation. These results were then compared to the ones acquired from the manual labelling process as a point of reference.

KASEPUHAN_1	KASEPUHAN_2	VALENTINO	PILATO	GHIFFA
325,822 points	598,384 points	4,188,066 points	16,200,442 points	17,798,012 points
8 pillars	20 pillars	6 pillars	20 pillars	13 pillars
TLS	TLS	TLS	TLS+UAV	TLS+UAV
Outdoor	Outdoor	Indoor	Outdoor/Indoor	Outdoor

Figure 1. The point clouds used for the experiments' section, with basic metadata as retrieved from the ArCH website. The number of pillars corresponds to free-standing pillars, i.e. does not include engaged columns.

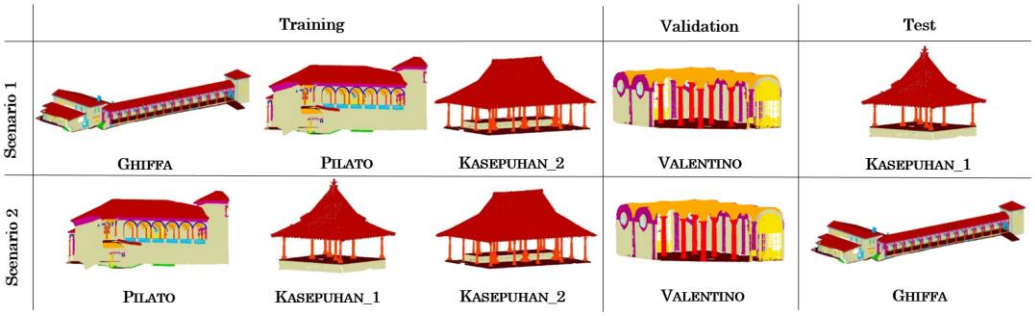
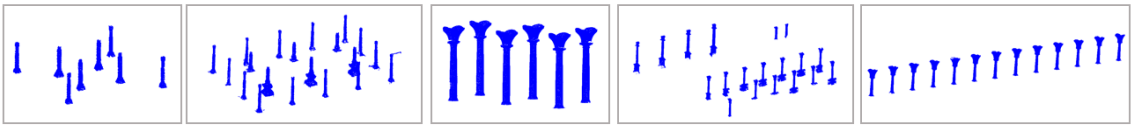


Figure 2. Configuration of the different datasets used in the two scenarios for the second experiment.



KASEPUHAN_1	KASEPUHAN_2	VALENTINO	PILATO	GHIFFA
8 pillars out of 8	20 pillars out of 20	6 pillars out of 6	19 pillars out of 20	13 pillars out of 13
25.1 secs	38.3 secs	307.8 secs	528.2 secs	254.54 secs

Figure 3. Visual result of the automatic segmentation and labelling for all five datasets.

Pursuing the aim of ensuring the repeatability, the second experiment is further divided into two scenarios in which different datasets were used for the training, validation, and testing phases. The configurations of the datasets used in these two scenarios are described in Figure 2.

Scenes containing an adequate number of columns with different characteristics and dimensions were chosen for the training of the network. For the test, on the other hand, the main criterion was the representativeness of different and diverse architectural styles, to test the generalization of the method. Indeed, Scenario 1 will perform the test on a Southeast Asian data while Scenario 2 will be applied to the European one.

The DL framework used in this experiment is the DGCNN-Mod developed by Pierdicca et al. (2020) with the geometric coordinates (XYZ), radiometric component (RGB) and the normal vectors as inputs. The normal vector was previously computed using the third-party software CloudCompare. The 3D features have not been added in this case (Matrone et al., 2020a), since the purpose of the tests is to investigate, by way of a comparison, if and how the automatic rules-based annotation affects the final prediction. The DGCNN-Mod training was set with hyperparameters identical to the values tested in Pierdicca et al. (2020), namely block dimensions of 1x1m, 4096 subsampled points for each block and a stride value of 1. Tests with 2x2m block size, 8192 points subsampled and stride 1 (in order to have overlapping blocks) have been carried out as well, but as no relevant differences emerged, only the prior ones will be shown and discussed in this paper. An NVIDIA RTX 2080 TI 11 GB, 128 GB RAM with processor Intel(R) Xeon(R) Silver 4214 CPU @ 2.20GHz was used for DL-related processing.

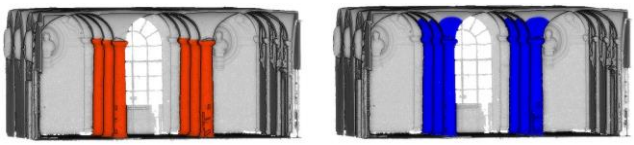
Finally, in this second experiment for each scenario the training will be performed twice: the first uses manually annotated point clouds while in the second the "column" class of the scene was replaced by the one predicted automatically by M_HERACLES as presented at the end of the first experiment.

4. RESULTS AND DISCUSSIONS

This section will be divided into two subsections, each describing the findings related to the first and second experiments respectively. The first subsection will describe the results of the automatic segmentation and labelling of pillars in the five available datasets using functions taken from the M_HERACLES toolbox. The second subsection will describe results from the second experiment, namely the use of outcomes from the previous subsection as training data for the developed DL algorithm and the assessment of its performance in comparison to manually labelled training data. Two scenarios will be presented in this second part of the section.

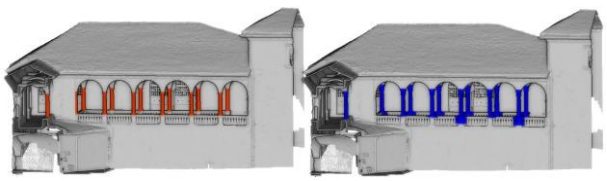
4.1 Automatic segmentation and labelling

As far as the rules-based segmentation and labelling is concerned, results show that the algorithm managed to properly detect the correct number of pillars in 4 out of the 5 datasets (FIGURE 3). In the case of the Pilato dataset, one pillar failed to be detected by M_HERACLES since it was attached to a wall segment. Similarly, the algorithm was unable to detect the engaged columns attached to the walls in the Valentino dataset, but otherwise successfully predicted the six free standing pillars at the centre of the scene. In terms of processing time, all datasets were processed in a reasonably fast processing time as also shown in FIGURE 3. This is arguably faster than manual labelling and moreover requires less human intervention and thus human-induced error. The experiment was performed by using an Intel(R) Xeon(R) E5645 2.4 GHz CPU. A more detailed comparison especially in terms of processing time vis-à-vis manual labelling shall be performed further in the text.

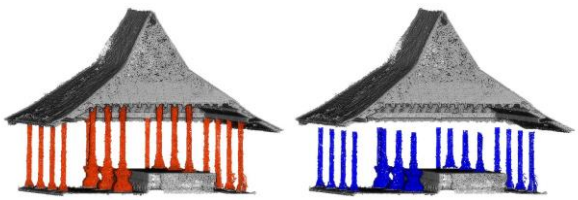


(a) Section of the Valentino's Castle courtroom

(b) SMV - Pilato Palace, South façade



(c) SMV - Pilato Palace, North façade



(d) Section of the Kasepuhan_2 Palace

Figure 4. Sample of pillars from each dataset, showing results of manual labelling as reference (red, left-hand side) and those from M_HERACLES (blue, right hand side).

Object	*Point count*		*True Positives*	*Erroneous points*		*%P*	*%R*	*%F1*
	Manual	*M_HERACLES*		*False positives*	*False negatives*			
Kasepuhan_1	24 607	16 227	16 219	8	8 388	99.95	65.91	79.44
Kasepuhan_2	40 942	32 230	31 481	749	9 461	97.68	76.89	86.05
Valentino	228 516	303 439	202 777	100 662	25 739	66.83	88.74	76.24
Pilato	408 771	744 802	386 481	358 321	22 290	51.89	94.55	67.01
Ghiffa	403 749	642 923	339 817	303 106	63 932	52.86	84.17	64.93
					Mean	**73.84**	**82.05**	**74.73**

Table 1. Table presenting the results of automatic pillar detection and classification using M_HERACLES for the five available datasets. %P, %R and %F1 indicate respectively Precision, Recall and F1-Score metrics.

Annotation	*Kasepuhan_1*	*Kasepuhan_2*	*Valentino*	*Pilato*	*Ghiffa*
Manual	3 mins	5 mins	8 mins	17 mins	14 mins
M_HERACLES	0.25 mins	0.5 mins	5 mins	6 mins	4 mins

Table 2. Comparison of annotation time.

As far as the processing time is concerned, lower point count seems to influence the overall duration. However, the number of detected objects is also an important factor. Indeed, the processing time for Valentino is much higher in spite of it having the fewest pillars because M_HERACLES attempted to detect (and rejected) candidate pillars from the surrounding scene i.e., walls and engaged columns. In reality, for the Valentino scene M_HERACLES detected 20 potential pillars, of which only 6 were retained. However, this also shows one of the shortcomings of the rules-based approach. Indeed, the ground truth received from manual labelling also classes engaged columns as "pillars". M_HERACLES, however, failed to take this into account and instead generated only free-standing pillars in its result.

Quantitatively speaking, the algorithm managed to obtain an average recall of 82.05% across the five datasets, even though the average precision is lower at 73.84% (Table 1). It is also worth noting that in Table 1 the point count of for manual labelling and M_HERACLES is different, because the manual labelling presents the reference ground truth and as such may be assumed as free from erroneous classification.

Furthermore, as may be inferred from Table 1, the two Kasepuhan datasets gave higher precision values with lower recall rates, while the other three datasets received a better recall score at the expense of the precision. This is due mainly to the nature of the datasets. The two Kasepuhan pavilions are particular as they do not possess ceilings; the roofs having been suppressed automatically by M_HERACLES beforehand (Murtiyoso and Grussenmeyer, 2019a). On the contrary, the other three Italian datasets possess either ceilings or arches at the top of each column. The 2.5D approach used by M_HERACLES means that it encountered problems when dealing with different classes of objects in the vertical space (Figure 4). Figure 4 further demonstrates the limitations of the current M_HERACLES implementation (blue) of the algorithmic approach in 2.5D. In Valentino, a part of the ceiling was also labelled as “pillar”. A similar phenomenon occurred with the arches and the pedestal in Pilato.

Consequently, for the European point clouds Type I error (false positives) was more dominant while for the Kasepuhan Type II error (false negatives) was more important; hence explaining the precision and recall values for both cases. The normalised F1 score value showed less discrepancy between the different architectural styles, with a mean value of 74.73%.

Furthermore, this type of semantic segmentation is less robust to noise; indeed, the systematic higher recall value is validated by the fact that M_HERACLES employs geometric rules to perform the detection. In this regard, due to the systematic hard knowledge embedded within the algorithm, the precision, recall, and F1 score values depend strongly on the choice of geometric parameters inputted into the algorithm. The main goal of this step is not to create the final semantic segmentation, but rather to accelerate the training data generation for further DL processing.

4.2 Comparison of processing time

Table 2 shows a comparison between the annotation times required for the different scenes using manual means and automatically using M_HERACLES. This simple comparison shows that manual annotation generally takes more time than the M_HERACLES algorithm. Furthermore, manual annotation also encounters the problem in that it requires an expert in domain to be able to correctly perform the labelling, as opposed to rules-based approaches in which this knowledge is already hard-coded into the algorithm.

If we consider the overall results, manual annotation generally guarantees more accurate predictions. However, the question that we would like to answer in this regard is whether this increase in accuracy at the expense of time and expertise can be replaced by rules-based segmentation, particularly within the context of using them further as input for neural networks. It can be beneficial to find a balance or a compromise between the quality of the results and the pre-processing times. The following subsection shall try to answer this question by feeding the results of the two approaches (manual and M_HERACLES) into our DL network and analysing the quality of the resulting semantic segmentation.

4.3 Application in deep learning training

This subsection will describe the use of the results from the previous subsection as input in DL training and how it can affect the semantic segmentation of point clouds. Two types of input scenes are used for training neural networks: manually labelled data and automatic labelled data obtained from M_HERACLES. For the experiments the DGCNN-Mod, a state-of-the-art neural network designed for the cultural heritage domain, will be used. As has already been explained, this work focuses only on the pillars/columns class objects. Therefore, only results related to this class will be shown and compared. Visual results and the main metrics for the segmentation task will be shown for every test. As has been previously established, experiments were performed on two different scenarios.

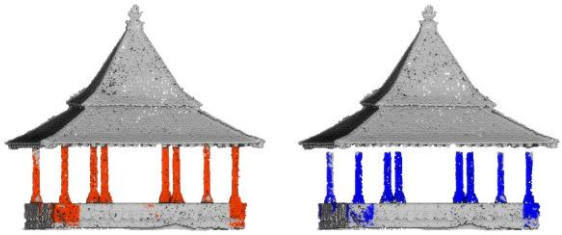


Figure 5. Predicted scenes of the first scenario on Kasepuhan_1. On the left side (red) the semantically segmented column class from the training using manually labelled data is shown. The results shown on the right (blue) used results derived from M_HERACLES for the training process.

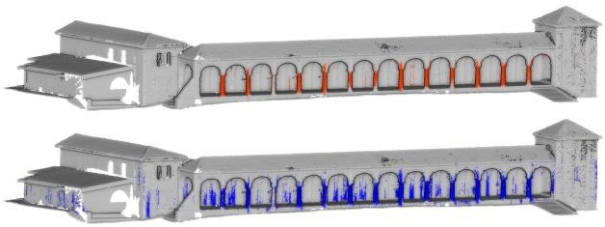


Figure 6. Predicted scenes for the second scenario. On the top side the result (red) from the manually labelled scenes used as training; on the bottom results (blue) from training using data derived from M_HERACLES.

4.3.1 Scenario 1: Result of the semantic segmentation as performed on the Kasepuhan_1 dataset can be seen in Figure 5. The results show a good level of identification of the column class in both cases. More specifically, from Figure 5 it can be seen how with the manually annotated data the columns are classified along their entire length. Meanwhile in the case of M_HERACLES results used as training data; the upper part of the columns was not included in the training set due to the algorithm's cropping of the ceilings. This resulted in the upper parts of the columns not included by the final prediction.

Another interesting point to note is that in the manually annotated scene, parts of the base of the pillars are recognised as the column class, while in the case of automatic annotation this misclassification is seen to be of a lesser extent. Theoretically speaking, the result from both approaches should be similar since no column base was included in either class. The reduced number of points in M_HERACLES's case due to its inherent denoising functions may have inadvertently reduced this effect for the automatic training data. Furthermore, since the neural networks are by nature opaque in some of their parts, it was not possible at this point to define the nature and source of this particular error.

When considering the metrics of the single class of columns as displayed in Table 3, the precision and the F1-score are greater in the case of manual labelling. The result is inverted in the case of the recall score. High precision score in this case indicates that manual labelling allows the training of a network that manages to correctly predict the column class. On the other hand, the M_HERACLES-based method with the higher recall score, creates a more generic network able to predict other classes outside of the column class.

4.3.2 Scenario 2: For the second scenario a starker difference between the two training datasets can be observed. As can be seen in Figure 6, semantic segmentation results derived from automatic annotation is visually noisier and tended to classify engaged columns on the walls behind to the arcade of the Ghiffa scene as columns.

Figure 7 further attempts to investigate this result in more detail. Indeed, the presence of some parts of the mouldings visible in the upper part of the columns in the case of the automatic automation may have led to these results.

The upper mouldings included by M_HERACLES in the training dataset it provided have many similar shape and geometric features as those of the engaged columns and half-pilasters. Furthermore, this behaviour is accentuated by the fact that only the radiometric component and normals have been used as input features. Indeed, the latter is shown to have significant influence in predicting the semantic segmentation results.

Quantitatively speaking, the general metrics as displayed in Table 4 are lower than those presented in the article by Matrone et al. (2020a), due to the lower number of scenes in the training set. Furthermore, compared to the previous scenario and considering solely those of the class of columns, a greater deviation between the manual and automatic-derived results in terms of Precision, Recall, F1-Score and IoU can also be observed. This result comes despite the relatively noise-free result from M_HERACLES, thus adding further to the argument put forward in the previous paragraph in which the misclassification of mouldings as columns generated a significant error.

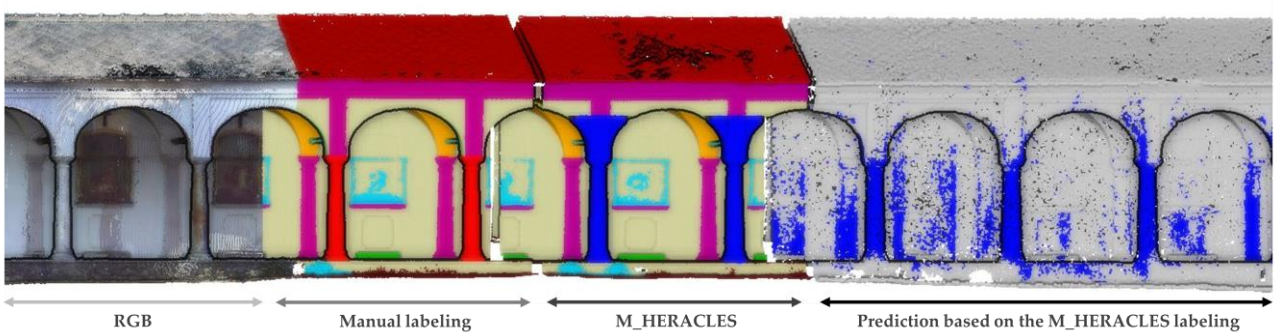


Figure 7. From left to right: detail of the point cloud in RGB, manual annotation of the columns, automatic annotation and finally the prediction results. Notice how the upper mouldings labelled as columns by M_HERACLES are very similar to the engaged columns on the wall behind.

Metrics	*Annotation*	*Accuracy*	*Precision*	*Recall*	*F1-Score*	*IoU*
Overall	Manual	0.9246	0.9359	0.9264	0.9264	0.3379
	M_HERACLES	0.9183	0.9353	0.9184	0.9159	0.3982
Columns	Manual	-	0.7615	0.9097	0.8290	0.7079
	M_HERACLES	-	0.6233	0.9240	0.7445	0.5929

Table 3. Results of the tests performed on the first scenario.

Metrics	*Annotation*	*Accuracy*	*Precision*	*Recall*	*F1-Score*	*IoU*
Overall	Manual	0.7292	0.7671	0.7292	0.7360	0.3825
	M_HERACLES	0.6332	0.7115	0.6332	0.6625	0.3036
Columns	Manual	-	0.7227	0.6378	0.6776	0.5123
	M_HERACLES	-	0.3851	0.5093	0.4386	0.2808

Table 4. Results of the tests performed on the second scenario.

5. CONCLUSIONS

This article described an alternative methodology to be applied for automatic labelling of point clouds. The output of the analysed algorithm can be, in fact, used as input for deep learning techniques. Being a preliminary study, this research focused mainly on the column class to evaluate a possible extension of this methodology to other classes.

The results obtained show that the proposed method managed to cut processing time by up to an average of six times faster than traditional manual labelling. The first scenario of DL implementation showed that despite lower annotation accuracy of automatic approaches, in simpler settings the proposed approach managed to attain similar quality as manual labelling. However, in the second scenario the more complex nature of the data presented other challenges. Automatically derived training data was essentially faced with a systematic error in the form of misclassified points. In this case we can observe that annotation accuracy is also important, at least in this case and using this neural network architecture. Additionally, the importance and great relevance of normal vectors in class recognition was demonstrated particularly in this scenario.

Although the metrics derived from the DL training based on automatic annotation are lower than the manual ones in both scenarios (in Scenario 2 more so than in Scenario 1), it is also important to note that the rules-based approach is both modular and adjustable. By knowing specific problems, the M_HERACLES functions can be tuned to adapt to specific cases and can thereafter be easily and rapidly deployed in a repeated manner. Although this may seem counter-intuitive, we argue that the margin of processing time gained by the proposed method permits a further in-depth tune up of rules-based approaches to increase the results.

Most importantly, we argue that the methodology described in this paper can provide a compromise between the pursuit of the best neural network performances and the reduction of overall processing times. Indeed, in the context of DCH this is crucial due to the virtually countless types of potential objects for DL training. Such compromise is therefore an interesting solution to greatly reduce processing time by pertaining to the required geometric specifications.

Some critical issues related to the difficulty of error track back in the proposed neural network remain, as is expected from the black-box nature of deep learning architectures. In this case, this issue meant that errors related to semantic segmentation results were more difficult to ascertain and, in some cases, to prove. Furthermore, the rules-based approach used in this paper as represented by the M_HERACLES toolbox also faces issues such as the requirement for further parameter tune-up to improve the results. These two issues, respectively pertaining to ML-based and geometric rules-based approaches are known and continue to be interesting topics to explore in the future. Future developments of this work can be an expansion of the proposed method into other classes in the cultural heritage context.

REFERENCES

Bassier, M., Bonduel, M., Genechten, B. Van, Vergauwen, M., 2017a. Octree-Based Region Growing and Conditional Random Fields, in: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. pp. 25–30.

Bassier, M., Vergauwen, M., Van Genechten, B., 2017b. Automated Classification of Heritage Buildings for As-Built BIM using Machine Learning Techniques, in: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences. pp. 25–30.

Bello, S.A., Yu, S., Wang, C., Adam, J.M., Li, J., 2020. deep learning on 3D point clouds. Remote Sensing 12, 1729.

Boulaassal, H., Landes, T., Grussenmeyer, P., Kurdi, F., 2007. Automatic segmentation of building facades using terrestrial laser data, in: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. pp. 65–70.

Campanaro, D.M., Landeschi, G., Dell'Unto, N., Leander Touati, A.M., 2016. 3D GIS for cultural heritage restoration: A "white box" workflow. Journal of Cultural Heritage 18, 321–332.

Chiabrando, F., Sammartano, G., Spanò, A., 2016. Historical Buildings Models and Their Handling Via 3D Survey: From Points Clouds To User-Oriented Hbim, in: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. pp. 633–640.

Drap, P., Papini, O., Pruno, E., Nucciotti, M., Vannini, G., 2017. Ontology-based photogrammetry survey for medieval archaeology: Toward a 3D geographic information system (GIS), Geosciences.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep learning. MIT press.

Grilli, E., Menna, F., Remondino, F., 2017. A Review of Point Clouds Segmentation and Classification Algorithms, in: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. pp. 339–344.

Maalek, R., Lichti, D.D., Ruwanpura, J.Y., 2019. Automatic recognition of common structural elements from point clouds

for automated progress monitoring and dimensional quality control in reinforced concrete construction. Remote Sensing 11.

Macher, H., Landes, T., Grussenmeyer, P., 2017. From Point Clouds to Building Information Models: 3D Semi-Automatic Reconstruction of Indoors of Existing Buildings. Applied Sciences 7, 1–30.

Macher, H., Landes, T., Grussenmeyer, P., 2016. Validation of Point Clouds Segmentation Algorithms through their Application to Several Case Studies for Indoor Building Modelling, in: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. pp. 12–19.

Malinverni, E.S., Pierdicca, R., Paolanti, M., Martini, M., Morbidoni, C., Matrone, F., Lingua, A., 2019. Deep learning for semantic segmentation of point cloud, in: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. pp. 735–742.

Matrone, F., Grilli, E., Martini, M., Paolanti, M., Pierdicca, R., Remondino, F., 2020a. Comparing machine and deep learning methods for large 3D heritage semantic segmentation. ISPRS International Journal of Geo-Information 9.

Matrone, F., Grilli, E., Martini, M., Paolanti, M., Pierdicca, R., Remondino, F., 2020b. Comparing Machine and Deep Learning Methods for Large 3D Heritage Semantic Segmentation. ISPRS International Journal of Geo-Information 9, 535.

Matrone, F, Lingua, A., Pierdicca, R., Malinverni, E.S., Paolanti, M., Grilli, E., Remondino, F., Murtiyoso, A., Landes, T., 2020. A benchmark for large-scale heritage point cloud semantic segmentation. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 43, 1419–1426.

Murtiyoso, A., Grussenmeyer, P., 2020. Virtual disassembling of historical edifices: Experiments and assessments of an automatic approach for classifying multi-scalar point clouds into architectural elements. Sensors (Switzerland) 20, 2161.

Murtiyoso, A., Grussenmeyer, P., 2019. Automatic Heritage Building Point Cloud Segmentation and Classification Using Geometrical Rules, in: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. pp. 821–827.

Murtiyoso, A, Grussenmeyer, P., 2019. Point cloud segmentation and semantic annotation aided by GIS data for heritage complexes, in: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. pp. 523–528.

Nguyen, A., Le, B., 2013. 3D Point Cloud Segmentation : A survey, in: 2013 6th IEEE Conference on Robotics, Automation and Mechatronics (RAM). IEEE, pp. 225–230.

Pellis, E., Masiero, A., Tucci, G., Betti, M., Grussenmeyer, P., 2021. Towards an Integrated Design Methodology for H-Bim, in: Proceedings of the Joint International Event 9th ARQUEOLÓGICA 2.0 & 3rd GEORES, Valencia (Spain), 26–28 April 2021. pp. 389–398.

Pierdicca, R., Mameli, M., Malinverni, E.S., Paolanti, M., Frontoni, E., 2019. Automatic Generation of Point Cloud Synthetic Dataset for Historical Building Representation, in: International Conference on Augmented Reality, Virtual Reality and Computer Graphics. pp. 203–219.

Pierdicca, R., Paolanti, M., Matrone, F., Martini, M., Morbidoni, C., Malinverni, E.S., Frontoni, E., Lingua, A.M., 2020. Point Cloud Semantic Segmentation Using a Deep Learning Framework for Cultural Heritage. Remote Sensing 12, 1005.

Poux, F., Hallot, P., Neuville, R., Billen, R., 2016. Smart Point Cloud: Definition and Remaining Challenges IV, 20–21.

Poux, F., Neuville, R., Billen, R., 2017. Point cloud classification of tesserae from terrestrial laser data combined with dense image matching for archaeological information extraction, in: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences. pp. 203–211.

Poux, F., Neuville, R., Nys, G.A., Billen, R., 2018. 3D point cloud semantic modelling: Integrated framework for indoor spaces and furniture. Remote Sensing 10, 1–35.

Qi, Charles R, Su, H., Mo, K., Guibas, L.J., 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 652–660.

Qi, Charles Ruizhongtai, Yi, L., Su, H., Guibas, L.J., 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space, in: Advances in Neural Information Processing Systems. pp. 5099–5108.

Riveiro, B., Dejong, M.J., Conde, B., 2016. Automated processing of large point clouds for structural health monitoring of masonry arch bridges. Automation in Construction 72, 258–268.

Rodríguez-Cuenca, B., García-Cortés, S., Ordóñez, C., Alonso, M.C., 2015. Automatic Detection and Classification of Pole-Like Objects in Urban Point Cloud Data Using an Anomaly Detection Algorithm. Remote Sensing 7, 12680–12703. Sanchez, V., Zakhor, A., 2012. Planar 3D modeling of building interiors from point cloud data, in: Proceedings - International Conference on Image Processing, ICIP. pp. 1777–1780.

Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M., 2019. Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics (TOG) 38, 1–12.

Xiong, X., Adan, A., Akinci, B., Huber, D., 2013. Automatic creation of semantically rich 3D building models from laser scanner data. Automation in Construction 31, 325–337.