As-scanned Point Clouds Generation for Virtual Reverse Engineering of CAD Assembly Models

Jérémy Montlahuc[1], Ghazanfar Ali Shah[1], Arnaud Polette[1] and Jean-Philippe Pernot[1]

[1]Arts et Métiers, LISPEN, HeSam, Aix-en-Provence, France

Corresponding author: Jean-Philippe Pernot, jean-philippe.pernot@ensam.eu

Abstract. This paper introduces a new approach for the generation of as-scanned point clouds of CAD assembly models. The resulting point clouds incorporate various realistic artifacts that would appear if the corresponding real objects were digitalized with a laser scanner. Such a virtual Reverse Engineering technique can produce a huge amount of realistic point clouds much faster than using classical time-consuming Reverse Engineering techniques on real physical objects. Here, there is no need to use a laser scanner and the post-processing steps are automatic. Using this technique, it is easy to create large databases of point clouds automatically segmented and labeled from the CAD models and which can be used for supervised machine learning. The proposed approach starts by generating a triangle mesh wrapping the CAD assembly model to be reverse engineered. The resulting watertight mesh is then sampled to obtain a more realistic distribution of points. The occlusion phenomenon is then simulated using a hidden point removal algorithm launched from several viewpoints. A misalignment procedure can optionally be used to simulate the fact that in real-life Reverse Engineering the position and orientation of the laser scanner and/or real object would have been changed to get a different scanning viewpoint. The virtual Reverse Engineering process ends by generating noise and by inserting outliers. The approach is illustrated and validated on several industrial examples.

Keywords: as-scanned point clouds, scanning artifacts, surface wrapping, sampling, hidden point removal, noise and outliers, reverse engineering.
DOI: https://doi.org/10.14733/cadaps.2019.1171-1182

1 INTRODUCTION

Over the last years, the use of artificial intelligence techniques to analyze and process geometric models has become a new trend in computer sciences. This is notably true when considering the segmentation and classification of 3D point clouds [5][9]. However, such techniques require the access to large datasets which may also have to be labeled when considering supervised learning

techniques. Therefore, one of the key issues when defining such learning approach is to be able to rely on available and trustable large datasets. This is not straightforward as it can take a lot of time to generate and process all the data. Actually, in Reverse Engineering process [8] the acquisition and processing time can range from a few minutes to several hours [1]. This strongly depends on the adopted technology (e.g. LIDAR, laser scanner, structured-light scanner, RGB-D sensor), the acquisition procedure followed by the operator and the complexity of the object to be reverse engineered. For example, scanning a simple sonotrode from multiple viewpoints (Fig. 1.a1) and treating (e.g. cleaning, filtering, registration, simplification, meshing) the resulting point clouds (Fig. 1.a2) can require up to several tens of minutes. Thus, it becomes unreasonable to try to manually generate thousands of point clouds using classical Reverse Engineering techniques on more complex existing physical objects or environments. Moreover, to be able to analyze and understand the impact of both the type of sensors and adopted control parameters, it is also important to have access to multiple point clouds of the same object following various acquisition scenarios. This further explodes the number of required acquisitions and associated treatments, thus justifying the need to develop the fully virtual Reverse Engineering technique presented in this paper.

In the proposed virtual Reverse Engineering approach, point clouds are automatically generated from CAD models of parts or assemblies. Thus, our approach can make use of available databases containing a huge amount of CAD models (e.g. GrabCAD, TraceParts, 3DModelSpace). The resulting point clouds incorporate various realistic artifacts that would appear if the corresponding real objects were digitalized with a real acquisition device. If the CAD models are labeled or enriched with semantic information, the generated point clouds could easily inherit from the available information. The inheritance procedure is not developed in this paper which focuses on the way point clouds can be realistically generated. As the approach is fully parameterized, for a given CAD model, several point clouds (Fig. 1.b1 to 1.b4) can be generated to simulate different scanning conditions (e.g. type of acquisition device and associated control parameters, environmental conditions, adopted acquisition sequence). As a consequence, a large variety of as-scanned point clouds can be generated in few second.

Figure 1: Real sonotrode (a1) scanned to get point clouds that have been post-treated (a2). CAD model of a sonotrode (b1) virtually reverse engineered following three parameterizations of the proposed framework (b2-b4).

To generate realistic as-scanned point clouds, it is important to analyze and to understand the multiple artifacts which can appear during a real reverse engineering session. Actually, artifacts can result from more or less complex phenomena generated by, and/or between, the acquisition device (e.g. type, control parameters), the digitalized object (e.g. material, color, shape, size), the operator (e.g. acquisition strategy, experience), and the environment (e.g. light, temperature, vibration). However, even

if the origins can be multiple, the impacts on the resulting point clouds can be classified according to five main categories (Fig. 2): non-uniform sampling, missing data, misaligned point clouds, noisy data and outliers [2].

(a) (b) (c) (d) (e)

Figure 2: Different artifacts: (a) non-uniform sampling, (b) missing data, (c) misaligned scans, (d) noisy data and (e) outliers[2].

This paper introduces a modular framework for the generation of as-scanned point clouds incorporating the above mentioned artifacts. Section 2 introduces the overall framework and discusses some of its characteristics. The modules and associated control parameters are detailed in section 3. Section 4 presents the tested configurations, the adopted evaluation criteria as well as the results on several industrial examples. The last section concludes this paper and introduces several perspectives.

2 OVERALL FRAMEWORK

The proposed approach is composed of several modules (Fig. 3). It starts by generating a triangle mesh wrapping the CAD model to be reverse engineered. The CAD model can either be a single part or an assembly of several parts. The resulting watertight mesh is then sampled to obtain a more realistic distribution of points. The occlusion phenomenon is then simulated using a hidden point removal algorithm launched from several viewpoints. As a result, the resulting point cloud is incomplete and some data are missing. Then, a misalignment procedure can optionally be used to take into account the fact that in real-life reverse engineering the object is acquired from several viewpoints. Thus, this procedure can modify the position and orientation of the point clouds' reference frames with respect to the reference frame of the original CAD model. The virtual Reverse Engineering process ends by generating noise and by inserting outliers.

Table 1 characterizes the different modules with respect to their ability to incorporate the above-mentioned artifacts. It clearly shows the complementarity of the different modules to achieve the desired results, i.e. being able to generate as-scanned point clouds incorporating artifacts appearing when scanning real-life physical objects. The approach is modular and each module is controlled by parameters which are detailed in the next section.

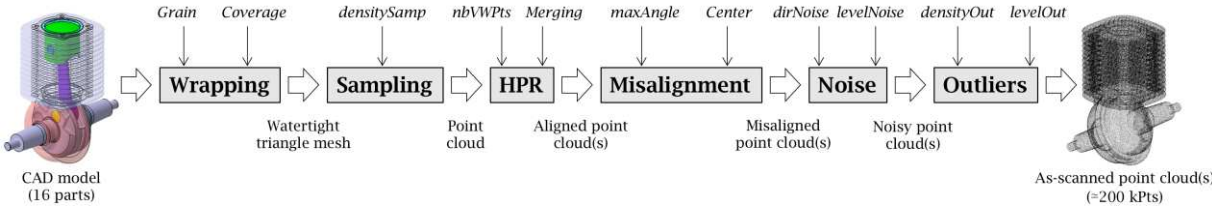

Figure 3: Virtual Reverse Engineering modular framework generating as-scanned point clouds from CAD assembly models.

		Modules					
		Wrapping	**Sampling**	**HPR**	**Misalignment**	**Noise**	**Outliers**
Artifacts	**Non-uniform sampling**	Yes	Yes	Yes	-	-	-
	Missing data	Yes	Yes	Yes	-	-	-
	Misaligned scans	-	-	-	Yes	-	-
	Noisy data	-	-	-	-	Yes	-
	Outliers	-	-	-	-	-	Yes

Table 1: Characteristics of the modules with respect to their ability to incorporate artifacts appearing when scanning real-life objects.

3 MODULES AND CONTROL PARAMETERS

This section briefly introduces the principles underpinning the proposed modules as well as the different control parameters which can be used to reflect as much as possible real scanning conditions.

3.1 Wrapping

The initial CAD model is wrapped to produce a watertight triangle mesh. As a result, some internal parts are not captured and some details of the resulting envelop can be simplified. This depends on the *Grain* accuracy and *wrapCoverage* ratio. The *Grain* characterizes the average distance between connected points of the resulting mesh. It can be set up according to the accuracy of the acquisition device to be simulated, for instance 50µm. The *wrapCoverage* determines the wrapping representation. A lower ratio will result in a thinner wrapping coverage. In the current implementation, the wrapping module makes use of CATIA V5 by Dassault Systèmes [4].

3.2 Sampling

The watertight triangle mesh resulting from the wrapping is then sampled to get a more realistic distribution of points. The amount of points is defined by the *densitySamp* expressing the number of points per square units. It can be set up according to the accuracy of the acquisition device to be simulated. The current implementation makes use of CloudCompare [3] to perform the sampling. Fig 4 illustrates the impact of the *densitySamp* parameter.

Figure 4: Sampling module: (a) watertight triangle mesh, (b) *densitySamp = 2.5, (c) densitySamp = 5.*

3.3 Hidden Point Removal (HPR)

To simulate multiple acquisition viewpoints, a simple and fast HPR operator can be run from several viewpoints. Following the approach of Khalfaoui et al. [7], a set of *nbVWPts* viewpoints are selected from a predefined list of positions around the object to be virtually reverse engineered. The removal of hidden points is performed using the approach of Katz et al. [6]. The adopted HPR operator determines the visible points of the point cloud, as viewed from a given viewpoint (Fig. 5.b and 5.c). An optimized version of this algorithm has been used to speed up the identification of hidden points using an octree-based decomposition. Through the *Merging* output option, the user can decide whether he/she wants to get multiple incomplete point clouds (*Merging* = 0) or if the point clouds are merged to get a single output point cloud (*Merging* = 1). However, this module does not try to filter overlapping areas which may result from multiple viewpoints. This module is optional.

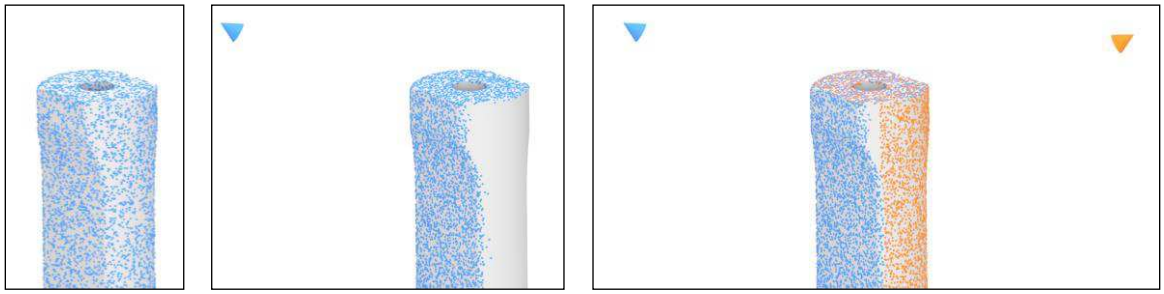

Figure 5: Hidden Point Removal module: (a) initial point cloud, (b) visible points after the HPR operator running from a given viewpoint shown with a cone, (c) from two viewpoints without merge (*Merging = 0*).

3.4 Misalignment

Since the HPR operator simply flags the visible points, the resulting point clouds perfectly fit in each other and there is no need to run an ICP algorithm. This differs from a real scan for which the point clouds acquired from different viewpoints appear in different reference frames. Thus, to simulate the fact that point clouds acquired with a real scanner would never fit perfectly, this module slightly rotates the point clouds one after the others. Of course, it can only be used if several (at least two) point clouds have been generated by the HPR module. Actually, a slight rotation is performed between two point clouds PC_i and PC_{i+1} generated from two successive viewpoints VP_i and VP_{i+1}. More precisely, PC_{i+1} rotates of an angle α_{i+1} that is defined by randomly selecting a value smaller than a user-specified *maxAngle*. As a default, the axis of rotation goes through the barycenter of the reverse engineered CAD model, and its direction is given by the plan's normal, defined by three points: the barycenter, the two viewpoints VP_i and VP_{i+1}. This misalignment procedure is run (*nbVWPts* – 1) times starting with the rotation of PC_2. The barycenter used to define the axis of rotation can optionally be substituted by another user-specified *Center*. This can be of interest when the CAD model has widely varying main dimensions. Fig 6 illustrates the misalignment with three viewpoints.

3.5 Noise

Being able to insert noise is an important feature of the proposed virtual Reverse Engineering framework [10]. Noise depends on many factors (e.g. type of acquisition device, material of the object, orientation of the sensor with respect to the surface) but it can be characterized by three main parameters: the type of distribution law, the amplitude of the noise, and the direction of the noise. In this work, the noise distribution law is a uniform one.

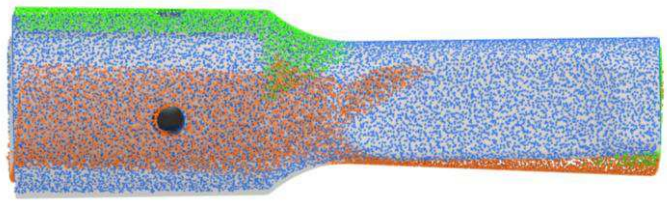

Figure 6: Misalignment module with three viewpoints, the barycenter as a rotation center and with the *maxAngle* = 3.5.

The remaining control parameters *levelNoise* and *dirNoise* (either along the normal to the surface, or along the line of sight) describe the way points are moved. The amplitude of the noise introduced at a given point is equal to 2 × *levelNoise* × (rand – 0.5) where rand returns a single uniformly distributed random number in the interval [0, 1]. Fig 7 illustrates the impact of the *levelNoise* parameter.

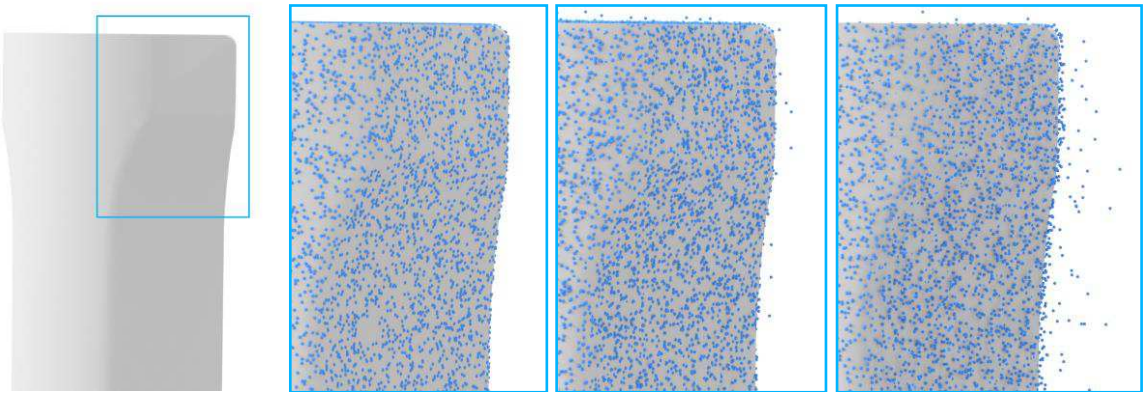

Figure 7: Noise module moving points along the normal to the surface (*dirNoise*): (a) zoom window, (b) *levelNoise* = 0, (c) *levelNoise* = 0.03, (d) *levelNoise* = 0.1

3.6 Outliers

Outliers are commonly due to structural artifacts in the acquisition process. In some instances, outliers are randomly distributed in the volume, where their density is smaller than the density of the points that sample the surface. Outliers can also be more structured, however, where high density clusters of points exist far from the surface. This can occur in multi-view stereo acquisition, where view-dependent specularities can result in false correspondences [2]. In the proposed approach, the idea is to identify a restricted set of points which can be duplicated and then moved randomly along the three directions of the space. Thus, the positioning of the outliers is driven by two control parameters: *densityOut* and *levelOut*. The first one is a percentage of points to be duplicated and moved. Points are uniformly selected among the points of the point cloud(s). Outliers then result from a duplication of those selected points which are then moved in the three dimensions of the space using three distinct amplitudes each of them being computed using the formula: 20 × *levelOut* × (rand – 0.5).

4 RESULTS

This section presents several tests which have been performed to validate the proposed approach on several CAD assembly models to be virtually reverse engineered.

4.1 Tested configurations

To analyze the influence of the previously introduced control parameters and to show the potential of the proposed approach, four configurations have been tested. The parameters associated to those configurations are summarized in Table 2. Each configuration can be considered as a setting characterizing a scanning session (e.g. type of acquisition device, material of the object, lightning conditions, expertise of the operator) to be virtually simulated using the proposed approach. The links between the characteristics of a real scanning session and those parameters are not detailed in this paper. Due to space limitation, all the parameters are not tested.

Steps	Parameters	Configurations			
		1	2	3	4
Wrapping	Grain	0.5mm	0.5mm	0.5mm	0.5mm
	wrapCoverage	0	0	0	0
Sampling	densitySamp	5	5	8	1M pts
HPR	nbVWPts	1	4	2	6
	Merging	1	1	1	1
Misalign-ment	maxAngle	0°	0°	0°	0°
	Center	Default	Default	Default	Default
Noise	dirNoise	Normal	Normal	Normal	Normal
	levelNoise	0	0.03	0.1	0.02
Outliers	densityOut	0.1%	0.1%	0.1%	0.1%
	levelOut	0	0.03	0.1	0.02

Table 2: Parameters of four configurations characterizing different scanning sessions to be simulated using the proposed virtual Reverse Engineering framework.

4.2 Evaluation criteria and results

The point clouds resulting from the newly developed virtual Reverse Engineering framework are evaluated according to three criteria: the number of generated points, the deviation to the original CAD model (mean and max), and the coverage ratio. Actually, the coverage ratio represents the amount of available information when compared to what is available in the original CAD model. In our implementation, it corresponds to the ratio between the area of the final triangle mesh generated from the point cloud and the overall area of the initial CAD model. The coverage does not depend on the density of points but on the number of viewpoints which are to be used. Coverage has been computed with respect to the area of the entire CAD models (including hidden part) and the coverage with respect to the wrapping results (which could in this case reach 100%). The first coverage factor can be used also to analyze how complex the CAD models are, i.e. if they do have a lot of internal and hidden parts. The second coverage factor is used to analyze if a sufficient number of viewpoints has been used.

Following the configurations of Table 2, the virtual Reverse Engineering technique is applied on three first CAD models. Results are presented in Table 3 and in Figures 1, 8, 9. Points are represented by spheres. To be able to visualize the points in a proper way, i.e. without having fully overlapping spheres, the number of points has been reduced in each figure.

Models	Parts (#)	Config.	Points (#)	Deviation (mm)	Coverage wrt wrapping (%)	Coverage wrt assembly (%)
Sonotrode	1	1	18 936	0.000 (0.0)	30.8	30.8
		2	61 327	0.033 (2.2)	76.9	76.9
		3	44 631	0.127 (7.5)	53.8	53.8

Compressor	16	1	54 831	0.00012 (1.35)	67.4	38.8
		2	91 950	0.00302 (2.36)	14.6	8.4
		3	163 241	0.07916 (8.48)	17.4	10.0
Engine	82	1	106 475	0.00023 (1.14)	24.1	11.7
		2	109 173	0.01145 (3.85)	29.1	14.2
		3	158 042	0.10563 (8.27)	32.9	16.0

Table 3: Results of the virtual Reverse Engineering process applied on three first CAD models (sonotrode, compressor, engine) and following three different scanning configurations.

Figure 8: CAD model of a compressor reversed engineered with configurations 1 to 3.

To further validate the efficiency and robustness, the proposed approach has been tested on additional CAD parts and CAD assemblies as shown in Fig 10. Again, points are represented by spheres, and the number of points has been reduced in each figure to improve the visualization. To keep the consistency in the results, the same evaluation criteria have been used. The parameters of the framework have been tuned using configuration 4 in Table 2, with a slight change in the way of sampling. Actually, to keep the same number of points for each example, parts are sampled with a fix number of points (1 million points). Viewpoints are also fixed for HPR by freezing one viewpoint on every face of the CAD part bounding box (front, back, top, bottom, right and left). Performing HPR from these six viewpoints ensures maximum coverage of point cloud with less viewpoints. Level of noise and outliers are taken same for these new examples to see the resulting behavior. Due to the different sizes of the parts and assemblies, the level of noise and outlier may affect the results differently. This can be observed in the Fig 10 where more noise can be seen on the clutch plate as compared to the connecting rod which has larger dimensions. Actually, in real scanning, smaller parts are difficult to handle as compared to bigger parts due to level of accuracy of laser scanners.

Table 5 represents the result of 9 examples shown in Fig 10. It can be seen that number of points have been reduced from 1 million, as there are many empty spaces in the resulting cloud due to HPR. It is not necessary that these six viewpoints will always cover the whole cloud, for full coverage we

may need many viewpoints as per complexity of CAD parts and assemblies. The values of coverage ratio also vary from product to product due to topologic configurations.

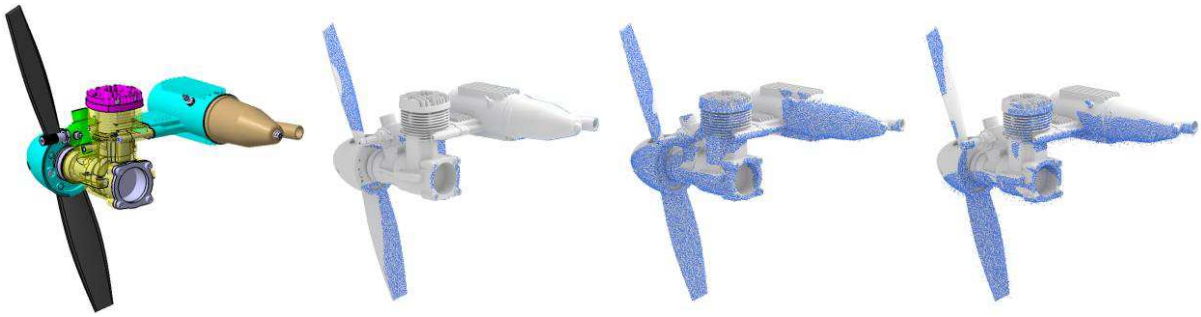

Figure 9: CAD model of a RC engine reversed engineered with configurations 1 to 3.

Figure 10: Results of the virtual Reverse Engineering process applied on CAD models following configure 4 of Table 2, from left to right and top to bottom: clutch plate, connecting rod, clutch plate assembly, radial engine rod, hand wheel, wheel assembly, vice, machine element and stop valve assembly.

As already explained coverage ratio corresponds to the ratio between the area of the final triangle mesh generated from the point cloud and the overall area of the initial CAD model, so it can never exceed 100 percent value. Due to same configuration of parts deviations are very close to each other. Deviations are calculated as mean distance of points in resulted point cloud to the mesh part.

Models	Parts (#)	Con-fig.	Points (#)	Deviation (mm)	Coverage wrt wrap-ping (%)	Coverage wrt assem-bly (%)
Clutch plate	1	4	525,822	0.0347 (1.98)	73.5	73.5
Hand wheel	22	4	945,190	0.0185 (3.11)	90.6	42.9
Machine element	14	4	873,182	0.0195 (3.91)	82.6	47.7
Radial engine rod	1	4	791,716	0.0196 (3.58)	79.1	79.1
Clutch plate assem-bly	36	4	939,037	0.0190 (4.75)	89.7	23.9
Stop valve assembly	35	4	880,576	0.0192 (2.77)	87.2	48.5
Connecting rod	6	4	703,292	0.0196 (5.56)	68.3	59.4
Wheel assembly	6	4	748,718	0.0190 (2.74)	71.2	66.8
Vice	8	4	895,662	0.0182 (2.76)	90.3	62.9

Table 5: Results of the virtual Reverse Engineering process applied on CAD models according to the fourth scanning configuration of Table 2.

4.3 Comparison with real-scanned point clouds

After performing tests on different examples, a comparison has been also carried out to analyze the deviations between real-scanned and as-scanned point clouds. In this way, the parameters of the proposed approach have been tuned so that the deviations match. ROMER Absolute Arm 7520 SI (7 axis and 2m volume, absolute encoders, RS1 laser sensor 30000pts / s, volumetric accuracy of 61μm) was used for data acquisition of the sonotrode (Fig 1). The point cloud obtained from the real scanning of the sonotrode has been compared with the point cloud generated with our method. Recorded deviation values were used further to tune the parameters of proposed approach (e.g. noise, sampling density and level of outliers). After series of tests and adjustments in different parameters, one set of parameters resulted with a minimized deviation. The values of those tuned parameters are summarized in Tab. 6. These parameters correspond to the defaults values that can be directly specified when the user chooses to simulate such a ROMER Absolute Arm 7520 SI. Of course, for a different device, the values could be different.

Steps	Parameters	ROMER Absolute arm 7525 SI
Wrapping	Grain	0.1mm
	wrapCover-age	0
Sampling	densitySamp	10%

HPR	nbVWPts	6
	Merging	1
Misalign-ment	maxAngle	0°
	Center	Default
Noise	dirNoise	Normal
	levelNoise	0.006
Outliers	densityOut	0.1%
	levelOut	0.005

Table 6: Default values of the framework's parameters so that the generated as-scanned point clouds fit the ones obtained by a ROMER Absolute Arm 7520 SI.

5 CONCLUSION

This paper has introduced a new virtual Reverse Engineering technique able to generate as-scanned point clouds from CAD models. The method is very fast when compared to the traditional Reverse Engineering process. It does not require any tedious and time-consuming post-processing steps. It is controlled by several parameters which values can be used to insert artifacts commonly encountered when dealing with real acquisition devices. This technique has been tested on several examples from single parts to assemblies. Different configurations are used to represent the variety in data acquisition as if the point clouds are obtained from different scanners. Parameters used for generating these results can be fine-tuned with the parameters of scanners for sensitivity and accuracy. The CAD assemblies considered here are perfect (perfectly fitted interfaces, sharp edges etc) which of course is not the same for product that has been manufactured. Effectively this is a limitation that the digital mockups of CAD assemblies used for this framework do not contain defects caused by manufacturing processes. Similarly, type of material is also not directly considered in this framework but rather indirectly through other parameters introducing artifacts which can be ascribed to surface properties. The proposed technique can be used to develop database of point clouds for educational and research purposes. This is a modular based approach and new capabilities can also be added to enrich the data. The next steps concern the definition of pre-defined configurations of the parameters so as to help the user instantiating them, the labeling of the point clouds while developing mechanisms able to capture and propagate the information from the CAD models, the generation of huge databases of as-scanned point clouds to be used for Artificial Intelligence applications.

Jérémy Montlahuc, http://orcid.org/0000-0003-2441-6812
Ghazanfar Ali Shah, http://orcid.org/0000-0003-3740-9945
Arnaud Polette, http://orcid.org/0000-0002-8572-6454
Jean-Philippe Pernot, http://orcid.org/0000-0002-9061-2937

REFERENCES

[1] Bagci, E: Reverse engineering applications for recovery of broken or worn parts and re-manufacturing, Three case studies, Advances in Engineering Software, 40(6), 2009, 407-418 https://doi.org/10.1016/j.advengsoft.2008.07.003

[2] Berger, M.; Tagliasacchi, A.; Seversky, L.; Alliez, P.; Guennebaud, G.; Levine, J. A.; Sharf, A.; Silva, C. T.: A Survey of Surface Reconstruction from Point Clouds, Computer Graphics Forum, 36(1) 2017, 301-329. https://doi.org/10.1111/cgf.12802

[3] Cloudcompare. http://www.cloudcompare.org

[4] Dassault Systèmes, https://www.3ds.com

[5] Huang, J.; You, S.: Point cloud labeling using 3D Convolutional Neural Network, International Conf. on Pattern Recognition, 2016, 2670-2675. https://doi.org/10.1109/ICPR.2016.7900038

[6] Katz, S.; Tal, A.; Basri, R.: Direct Visibility of Point Sets, Association for computing machinery Transactions on Graphics, 26(3), 2007, 24. https://doi.org/10.1145/1276377.1276407

[7] Khalfaoui, S.; Aigueperse, A.; Seulin, R.; Fougerolle, Y.; Fofi, D.: Fully automatic 3D digitization of unknown objects using progressive data bounding box, Three-Dimensional Image Processing and Applications II, 2012, 829011. https://doi.org/10.1117/12.909164
[8] Kumar, A.; Jain, P. K.; Pathak, P. M: Reverse Engineering in Product Manufacturing, 2013, 665-678. https://doi.org/10.2507/daaam.scibook.2013.39
[9] Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J.: PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation, IEEE Conf. on Computer Vision and Pattern Recognition, 1(2), 2017, 77-85.
[10] Sun, X.; Rosin, P. L.; Martin, R. R.; Langbein, F. C.: Noise in 3D Laser Range Scanner Data, Shape Modeling and Applications, 2008, 37-45.