

Research Article

Reproducibility Experimentation among Computer-Aided Inspection Software from a Single Point Cloud

Ibtissem Jbira ^{1,2,3}, Antoine Tahan ¹, Serge Bonsaint,⁴ Mohamed Ali Mahjoub,³ and Borhen Louhichi⁵

¹*École de Technologie Supérieure (ÉTS), 1100 Notre-Dame Ouest, Montréal, Québec H3C 1K3, Canada*

²*Institut Supérieur d'Informatique et des Technologies de Communication, G. P. 1 Hammam Sousse-4011, Sousse, Tunisia*

³*Université de Sousse, École Nationale d'Ingénieurs de Sousse, LATIS—Laboratory of Advanced Technology and Intelligent Systems, Sousse, Tunisia*

⁴*Creaform 3D 4700, Rue de la Pascaline Lévis (QC), G6W 0L9, Lévis, Canada*

⁵*Université de Sousse, Institut Supérieur des Sciences Appliquées et de Technologie de Sousse, LMS, Cité Taffala 4003, Sousse, Tunisia*

Correspondence should be addressed to Ibtissem Jbira; ibtissem.jbira.1@ens.etsmtl.ca

Received 21 June 2019; Accepted 9 September 2019; Published 13 October 2019

Academic Editor: Paolo Mercorelli

Copyright © 2019 Ibtissem Jbira et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The ISO GPS and ASME Y14.5 standards have defined dimensional and geometrical tolerance as a way to express the limits of surface part variations with respect to nominal model surfaces. A quality-control process using a measuring device verifies the conformity of the parts to these tolerances. To convert the control measurement points as captured by a device such as a coordinate measurement machine (CMM) or noncontact scan, it is necessary to select the appropriate algorithm (e.g., least square size and maximum inscribed size) and to include the working hypotheses (e.g., treatment of outliers, noise filtering, and missing data). This means that the operator conducting the analysis must decide on which algorithm to use. Through a literature review of current software programs and algorithms, many inaccuracies were found. A benchmark was therefore developed to compare the algorithm performance of three computer-aided inspection (CAI) software programs. From the same point cloud and on the same specifications (requirements and tolerances), three CAI options have been tested with several dimensional and geometrical features.

1. Introduction

These days, 3D computer-aided design software provides various options to fit the measured points to a XYZ coordinate system. Most software features a selection of different “fitting algorithms” that adjust the measured points in various methods. Metrologies have the flexibility to fit a part in multiple ways using postprocessing software. The performance of a measurement system is an estimate of a combination of various measurement errors (random and systematic) that include hardware (equipment error), software (algorithmic error), and operators. Both the software and operator can still strongly correlate. In this paper, algorithmic errors were focused. A novel benchmark model is

proposed in this study to compare the fitting algorithms' performance in different CAI software programs. According to a measurement system analysis (MSA) approach, it is strictly a reproducibility study. The purpose of inspecting is to verify compliance with the design requirements.

To keep track of the process, it is helpful to understand the error levels on the important features. In addition, the inspection report must be useful and easily interpretable to make the necessary corrections if the required tolerances are not respected. Form testing machines allow to measure points to form the profile and estimate circularity errors. Usually, all these machines provided many options for evaluating the circularity errors. Therefore, it is needed to make a decision about the reference form to use for a

particular application. This paper aims to analyze algorithmic errors. A quick review of the available literature on this subject finds that the impact of a measurement strategy (amount of data) and software used (including algorithmic and filtering outliers) is a surprisingly underexplored topic. Recent work in test of fitting algorithms (maximum inscribed, least square, minimum circumscribed, and Chebyshev) indicates that serious problems can exist in the current commercial software packages.

2. Background

Computational metrology includes the implementation of robust algorithms to adjust and achieve many computations on discrete data collected by measuring machines. The computational coordinate metrology is the most pertinent to validate standards that conform to tolerance specifications (see Figure 1). Now many journals consider computational metrology as a separate topic of interest. Computationally, fitting problem is an optimization problem:

$$l_p = \arg \min \left\{ \sum_{i=1}^n |d_i|^p \right\}^{1/p}, \quad (1)$$

where n is the number of points and d_i is the shortest distance between measurements data and the reference form. If $p=2$ the fitting problem is the least square fitting, and if $p \rightarrow \infty$, l_∞ fitting is the Chebyshev or Minimax fitting.

Generally, the fitting problem consists to select an appropriate algorithm to adjust a geometric form (e.g., plane, cylinder, and circle) to data points collected from the inspection of a manufactured part. The perfect form estimation is obtained by a fitting is called a reference feature or substitute feature. The recent publication of ISO 14405-1: 2016 determines size as a fundamental geometric descriptor. It also defines a new set of modifier tools for size [1, 2]; Jbira et al. [3] proposed a methodology to evaluate the algorithmic error of modifiers in the ISO 14405-1 [4] standard. They demonstrated that the noise measurement affects the ISO 14405-1 modifier.

Various methods have focused on the problem of fitting surfaces applied to many fields. Thus, in the absence of recommendations by the ISO about the method adopted to terminate roundness [4, 5], the estimation of circularity errors remains a challenge for the improvement of algorithms to calculate the best center.

In this section, some of the algorithms cited in the state-of-the-art [6–31] will be briefly described, particularly in the case of geometric elements of circular, cylindrical, and spherical forms. In [6], a reference circle is calculated from the input data in order to minimize the deviation between actual and reference circles. One of the methods to find this circle is a Voronoi diagram used to determine the minimum circumscribed circle (MCC) or maximum inscribed circle (MIC). To obtain a MCC, the points of the diagram that lie farthest from the center of the diagram are used. The MIC has a center on the Voronoi vertex or on the Voronoi edge. The distance between the circle center and the convex vertex

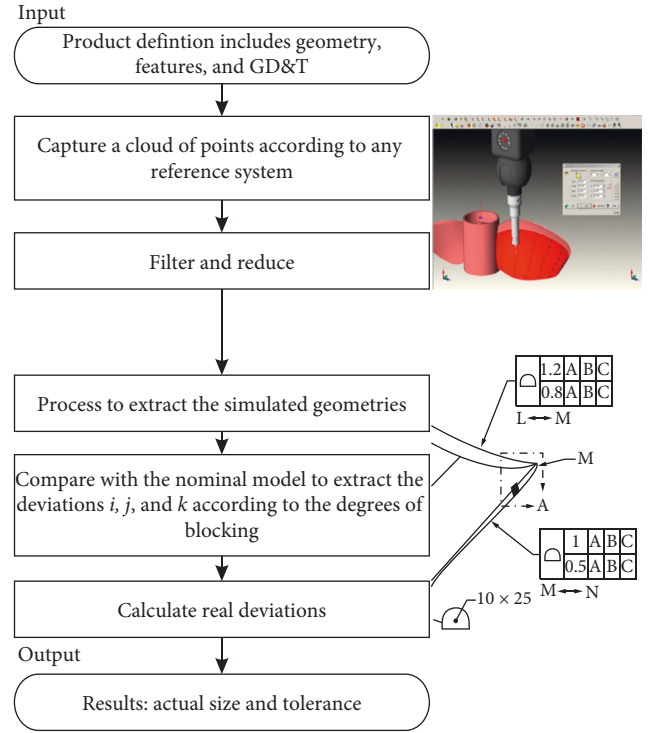


FIGURE 1: Inspection process.

represents the radius. The Delaunay triangulation is used to calculate the convex and Voronoi diagram. In [9], the roundness error from the input 3D points is evaluated using four methods: minimum zone circle (MZC), least squares circle (LSC), MCC, and MIC. For the MZC method, the roundness error is measured using two concentric circles. For the LSC method, it is estimated inside the profile by minimizing the sum of radial squares between the circle and the profile. Using the center of the LSC, a circumscribed circle and an inscribed circle are obtained. The circularity error (out of roundness) value is the radial separation. The MIC method is used to fit the largest circle inside the profile. In the case of a MCC, a center is calculated by finding the circle that has the smallest radius containing the points of the circle. The circularity error is the difference between the radiuses of the inscribed circle drawn using the found center and the circumscribed circle. Tran et al. [12] proposed an algorithm to fit cylinders and to approximate the parameters given by 3D point cloud. The first step is to compute the normal vector of the points. Using curvature information, potential points that could form the cylinder are identified. These points should be updated in a fitting process to check all remaining points that belong to a cylinder. Then, the mean shift clustering method is applied to find the approximated parameters of the valid cylinder. This method is validated using different models featuring various noise and outlier levels. In 2013, Srinivasan et al. [13] proposed a solution to the problems of weighted total least squares fitting of lines, planes, and parallel planes. They demonstrated the need for these algorithms, which is relevant to newer tolerancing standards and instrumentation.

In 2017, Saval-Calvoa et al. [23] presented a novel method based on a random sample consensus (RANSAC). This method estimates multiple planes from a point cloud with noise. The proposed method based on two steps: the first step is to divide the data into planar faces, and in the second step, the plane models are approximated using the multiconstraint RANSAC method. They tested the proposed method with existing methods.

In [14], the authors compared different algorithms to facilitate the choice of the adequate execution method for MCC, MIC, and MZC in order to calculate roundness errors. In addition, they used a new geometric concept based on reflecting a mapping technique to assess roundness errors. They proposed a selected benchmark of algorithms in the literature in order to provide the optimal execution method. It was concluded that no single algorithm provides the best solution. Geometric primitive reconstruction [6, 15–17] is an important problem in the field of computer-aided design (CAD).

Goch and Lübke [16] proposed a new algorithm to approximate geometry elements using Gauss and Chebyshev criteria. Furthermore, Chaperon and Goulette [18] proposed an algorithm to extract a cylinder from unorganized 3D points. The two main steps of their method were the extraction of the constrained plane in the Gaussian image to define the direction of a possible cylinder and extraction of the cylinder of known directions from the set of 3D points. They validated their approach by testing the extraction of pipe objects in industrial environments. In the context of pipeline plant detection and reconstruction, the huge number of points makes the problem extremely difficult. Traditional methods of detection cannot be applied directly because of the high complexity. Moreover, in [19], the authors developed a robust algorithm based on a principal component analysis (PCA) approach to fit cylinders given a set of 3D points. They validated their method using artificial and real point clouds. Most cylinder approximation methods focus on full data. However, the point cloud data that they obtained through laser scanning was incomplete and contained outliers. Nurunnabi et al. [20] described the problem of circle fitting for complete and incomplete data sets with outliers. They proposed a robust approach for circle fitting which had to merge two algorithms: PCA and robust regression. The experimental results confirmed the robustness of the proposed approach with a different percentage of tolerance of clustered outliers. They compared the proposed approach with another exciting method. Guo and Yang [21] proposed a new procedure for circle fitting. They used Taubin's approach to compute the center and radius, and then they identified and removed the outliers by calculating the geometric distances given point cloud to form the adequate circle. Their experiments demonstrated that the iterative procedure could resist against the effect of outliers. More specifically in the case of geometric elements of plane form, Deschaut and Goulette [22] proposed an accurate algorithm to extract planes in noisy point clouds using filtered normal and voxel growing. The first step is the estimation of the better normal at the data points. The second step consists of computing a score of local planes, and then

they apply the growing voxels. Finally, they evaluated the proposed algorithm on different number of points and compared it with existing algorithms. The presented method has a linear algorithmic complexity and it is able to detect large and small planes in very large data sets. In 2017, Nguyen et al. [24] presented a comparative study of the least square plane fitting algorithms with different segmentation methods (e.g., RANSAC, RGPL, Cabo, and RDPCA). They validated the study by two real point clouds collected by a Dynascan S250 scan system. The results demonstrated that the RGPL method gives the best results for planar surface extraction in moving least squares (MLS).

In 2018, Marriott et al. [25] presented an unsupervised extraction planar method. They proposed to adjust the data with a piecewise-linear Gaussian mixture regression model whose components were skewed over planes. In [28], the problem of fitting full and half-geometrical primitives (e.g., circular, spherical, and cylindrical) is addressed. The authors use the Levenberg–Marquardt (LM) method to approximate these geometries. They also proved that using the chaos optimization method improves the initial algorithm estimation. In fact, the chaos-LM algorithm provides efficient results even when the input data points are incomplete and noisy.

The literature review demonstrates the importance of geometric elements of circular, cylindrical, plane, and spherical forms fitting problem to industrial designers and inspectors. Therefore, the quality of the deformed geometrical is evaluated by the fitting algorithms. The literature supplied with different algorithms for the cylinder fitting is used to quantify roundness error. Despite different algorithms being proposed, the optimal solution has not been produced. The designers should carefully select the appropriated algorithm. Through this literature review of current software programs and algorithms, many inaccuracies were found.

3. Experimental Protocol

This paper is interested in the reproducibility estimation between three computer-aided inspection software programs that originate from the same point clouds. According to the industrial guide put out by the Automotive Industry Action Group [AIAG], reproducibility traditionally refers to the variation in average measurements made by different operators using the same gauge and parts.

Typically, the term is defined as the average measurement made by different appraisers using the same measuring instrument when measuring identical characteristics on the same part. In our study, one part was used (same drawings and same specifications) with the same measurement point clouds and appraiser but three inspections software programs were used. For each software, all proposed options by each software are tested (e.g., Best-fit and Minimax). Therefore, variability expresses the algorithmic differences between software programs. Voluntarily, we have not publicized the names to maintain certain confidentiality. The study was not conducted to provide a complete judgment on software. It has the focus only to estimate the reproducibility

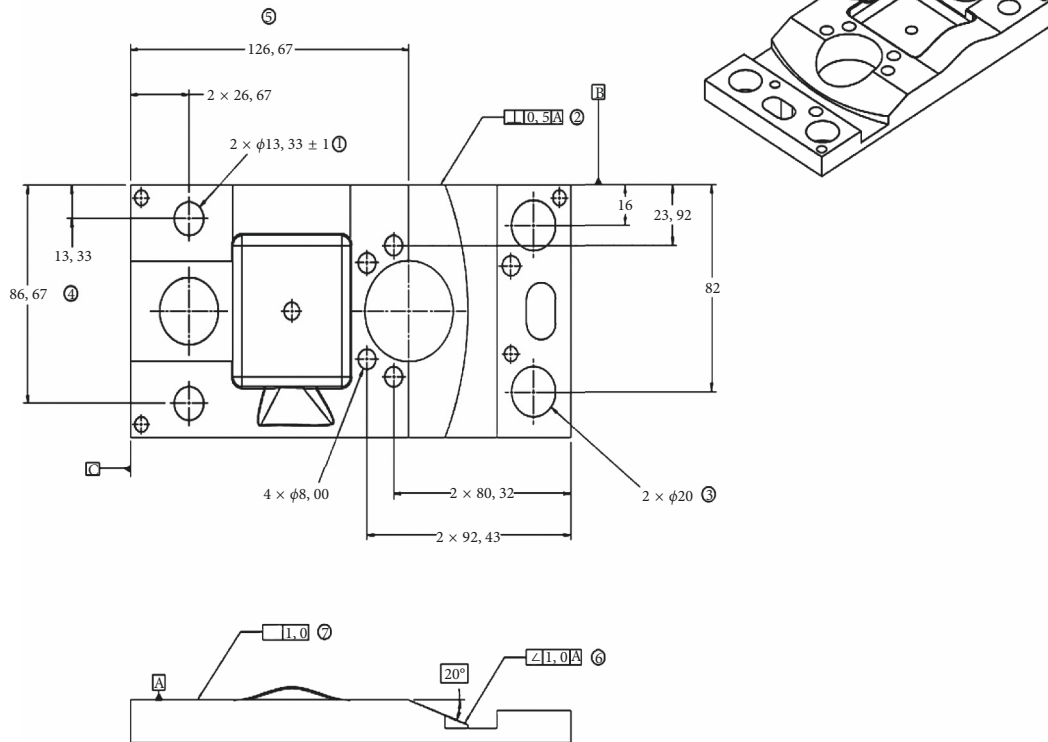


FIGURE 2: GD&T of the sample part with numbering items.

of the used algorithms. Their choice of these three was dictated by the following point: they are commercial software widely used in the industry (>100 costumers for each software).

A piece is designed using basic feature geometries (e.g., planes and cylinders). Dimensional and geometrical requirements are presented in Figure 2. One part is made out of aluminum and the piece is digitized by a HandySCAN portable 3D scanner (with a resolution of 0.05 mm and an accuracy of up to 0.04 mm). This part is to be used for fitting performance evaluation and inspection algorithms while considering dimensional and geometrical tolerances. In this paper, the standard ASME GD&T Y14.5-2009 [27] was used. In total, five different features with dimensional and geometrical tolerances were selected on the test pieces (plane, circle, and oblong hole). Only one operator was involved in the study. Three software programs were tested with all of their options.

The reproducibility in this study (expressed as a standard uncertainty u_A) represents variations due to the difference between algorithms and how the software deals with outliers and registration fit.

Alignment tools can define a part reference frame if you know the nominal location of point-reducible features on the part (e.g., points, circles, and spheres). Before performing a datum alignment, the operator has to provide a close initial part reference frame. The operator can use any available part alignment tool (e.g., CAD—cloud point

alignment with the ICP algorithm [20] or 3-2-1 alignment) to perform this initial part alignment. The geometric plane line point (PLP) alignment enables us to align a part to the CAD model using a plane (primary datum), a line (secondary datum), and a point (tertiary datum) to create the nominal coordinate reference system. In this paper, for each software, the operator has systematically carried out the PLP alignment.

4. Methodology Analysis

In the case of circle features, the results are the measured diameter, the localization of the center, and the roundness error. In the case of plane features, the result is only the flatness error (Figure 3). As mentioned, three software programs have been tested with all options (Table 1).

The alignment performed is a PLP-type alignment using the measured ABC datum using the least squares algorithm. For the CPU time, and for all tested options, below one second in all cases on a PC with a 2.4 GHz Intel Core i3 processor and a physical memory of 4.0 GB manual intervention time: friendly and very similar in the three cases.

5. Results

In the measured part, two circles called circle #1 and circle #2 have been extracted, and an oblong hole for evaluating each option for each one of the three software programs. In

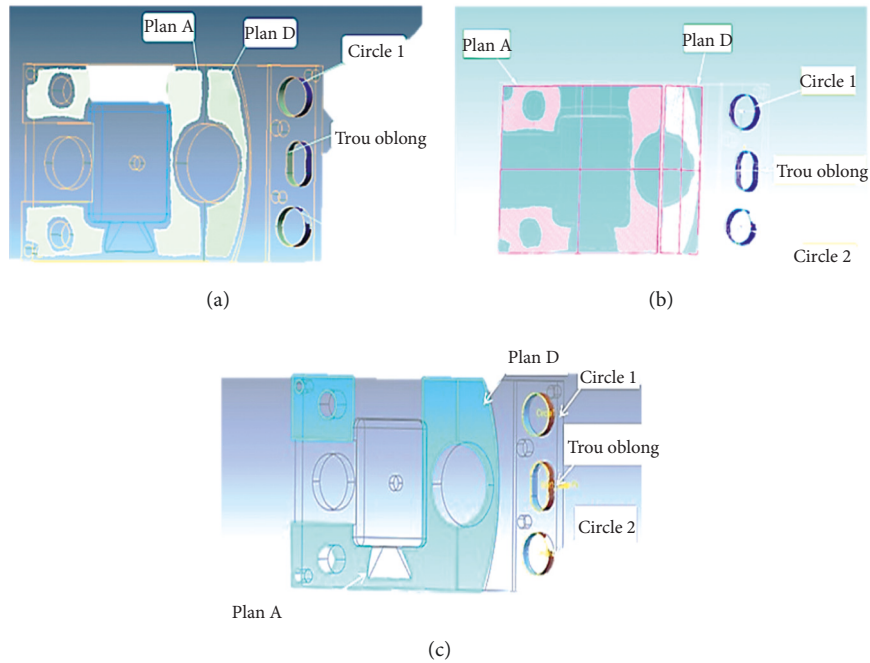
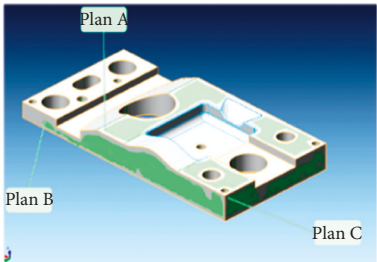
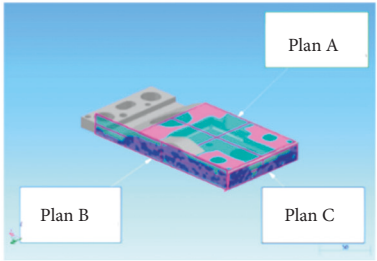
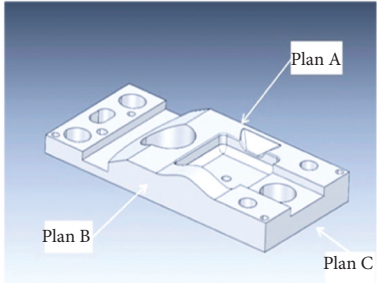


FIGURE 3: Elements measured for analysis. (a) Software #1. (b) Software #2. (c) Software #3.

TABLE 1: Options for three software programs.

Described options	Software
<p><i>Best-fit</i>: an option button that specifies the use of the usual “Best-Fit” fit algorithm that performs a least squares approach.</p> <p><i>Min</i>: an option that specifies the use of the minimum fit algorithm. A primitive Min is the largest primitive that does not encompass any element.</p> <p><i>Max</i>: an option that specifies the use of the maximum fit algorithm. A Max primitive is the smallest primitive that contains all the elements.</p> <p><i>Least square</i>: to create a circle that best fits the probed points. The circle is calculated by minimizing the sum of the squares of the gaps between the circle and at each point probed.</p> <p><i>Maximum inscribed</i>: to create the largest circle that fits in the probed points.</p> <p><i>Circumscribed minimum</i>: to create the smallest circle that contains all the probed points.</p> <p><i>Minimax</i>: to create the circle by averaging between the maximum inscribed circle and the circumscribed minimum, which have the same center (minimizing a circularity error).</p> <p><i>Best-fit</i>: find geometry using the least squares method from contours or surfaces.</p> <p><i>Min</i>: find the maximum circumscribed geometry using the minimum separation method from the contours or surfaces. Used to find the maximum circumscribed.</p> <p><i>Max</i>: find the minimum circumscribed geometry using the minimum separation method from the contours or surfaces. Used to find the minimum circumscribed geometry that encounters the furthest point to the positive direction of the normal.</p>	 <p>#1</p>  <p>#2</p>  <p>#3</p>

addition, two surface planes have been extracted called plane A and plane D to calculate the flatness found in each software.

5.1. Analysis of Circle #1. In the first test, the deviation of the center was computed on the circle in the x and y directions (δ_x , δ_y) and the measured diameter (\emptyset) was calculated, the diameter deviation ($\delta_\emptyset = 20 - \emptyset$) and the roundness error of the circle #1 (Table 2). Results with the least square option are shown in bold, results with the Min option are shown with an underscore, and results with the max option are shown with a double underscore and, if applied, results with the Minmax option are shown in italics. According to Table 2, CAI software #1 and #3 offer three adjustment algorithms for circle #1 (least square circle (LSC), minimum circumscribed circle (MCC), and maximum inscribed circle (MIC)).

Four algorithms are available in the CAI software #2 (LSC, MIC, MCC, and Minimax). The LSC algorithm provides the same value of measured diameters for the three software programs (19.877 mm). It is an expected result, as the LSC is a deterministic algorithm. Software #1 and #2 with the MCC algorithm provide the same values of measured diameters (20.054 mm), but software #3 gives a bit of a different value (20.059 mm). The range (r) between results is equal to 0.005 mm. Again, the MIC algorithm gives the same values for the software #1 and #2 (19.755 mm), but software #3 gives a little different value (19.745 mm), $r = 0.01$ mm.

Finally, the Minimax algorithm is available only by software #2; the value of diameter is 19.903 mm. According to Figure 4(a), software #1 and #3 provide the same circularity values for all the methods, and software #2 demonstrates different results for each method used. In the case of size in Figure 5(b), the LSC shows the same size for the three software programs despite the fact that the Max and Min algorithms display different results.

5.2. Analysis of Circle #2. According to Table 3, results and conclusions are very similar to Table 1 (circle #1). The LSC algorithm provides the same value of measured diameters for software #1 and #2 (19.874 mm). In addition, software #1 and #2 with an MCC algorithm provide the same values of measured diameters (20.054 mm), but software #3 shows a bit of a different value (20.054 mm). The range (r) between results is equal to 0.005 mm. Again, the MIC algorithm shows the same values for software #1 and #2 (19.760 mm), but software #3 provides a slightly different value (19.721 mm) with $r = 0.039$ mm. Finally, with the Minimax algorithm, available only in software #2, the diameter is equal to 19.907 mm. According to Figure 4(a), software #1 and #3 provide the same circularity values for all methods (Max, Min, Best-fit, and Minimax), and software #2 provides different results for each method. Regarding the size in Figure 4(b), the LSC gives the same size for the three software programs, despite the Max and Min algorithms displaying different results.

According to the circle #1 and circle #2 results, in the case of diameter measurements, there is a small variation. However, in the case of more complex GD&T, there are greater variations (circularity cases).

5.3. Analysis of the Oblong Hole. In the second test, the deviation was computed on the hole location on the axis x (δ_x), and the measured length and width (β , γ), the length deviation ($\delta_\beta = 22.666 - \beta$), and the width deviation ($\delta_\gamma = 13.333 - \gamma$) were calculated (Table 4). According to Table 4, CAI software #1 and #3 offers three adjustment algorithms for oblong holes (LSC, MIC, and MCC). Four algorithms are available in CAI software #2 (LSC, MIC, MCC, and Minimax). The LSC algorithm displays different values for the measured width and length on software #1, #2, and #3 (see Figures 6(a) and 6(b)) (13.139 mm, 13.152 mm, and 13.151 mm). This is abnormal, since the LSC is a deterministic algorithm that normally should provide the same value of measured width. According to the Figure 6(b) software #1 and #3 with a MCC algorithm displays the same value of measured width (13.320 mm), but software #2 shows a different value (13.316 mm). The range between results is equal to 0.004 mm. Again, the MIC algorithm gives different values of measured width for software #1, #2, and #3 (12.989 mm, 13.048 mm, and 13.320 mm). Finally, the Minimax algorithm is available only by software #2, where the value of the width is equal to 13.164 mm.

5.4. Analysis of Plane A. In the third test, the flatness was computed of two planes, Plane A and Plane D (Table 5). Results with the least square option are shown in bold; results with the Min option are shown with an underscore, and the max option is displayed using a double underscore (Figure 7).

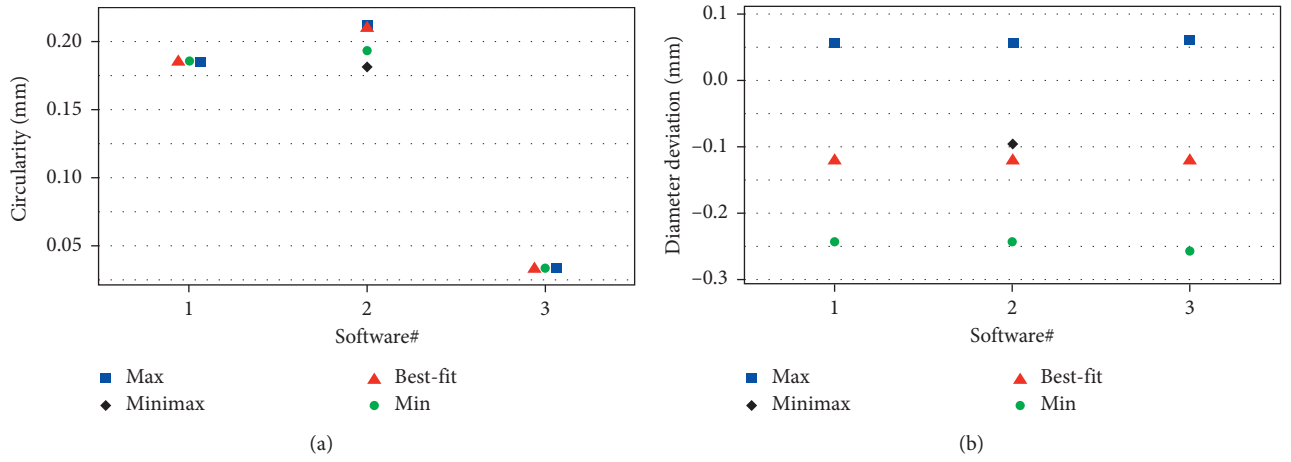
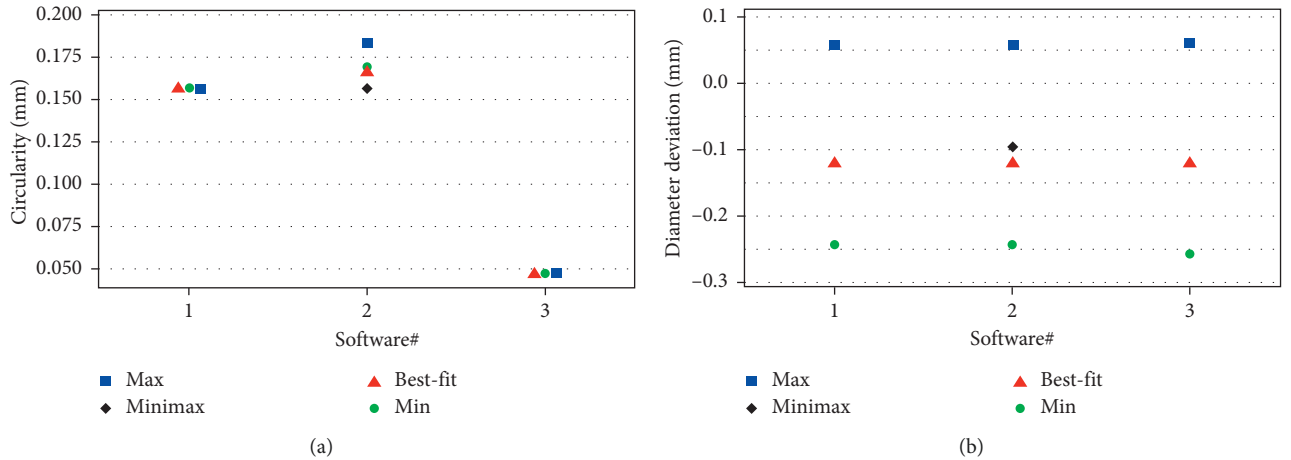
According to Table 5, CAI software #1 and #3 offer three adjustment algorithms for Plane A, the least square plane (LSP), minimum plane (MP), and maximum plane (MIP). The CAI software #2 only offers one algorithm (LSP) to fit Plane A. Only the best fit is available. The software #1 gives the same flatness value (0.107 mm) for the three algorithms (LSP, MC, and MIP). In the case of software #2, only the best fit is available (0.119 mm).

5.5. Analysis of Plane D. According to Table 6, the results are similar to Plane A. software #1 gives the same flatness value (0.070 mm) for the three algorithms (LSP, MC, and MIP). In the case of software #2, only the best fit is available (0.0312 mm), but software #3 gives different flatness values for each of the three algorithms (LSP, MC, and MIP) (Figure 8).

In this paper, an investigation of reproducibility estimation was presented. Similar point clouds, operators, CAD, and many fitting algorithms with different shapes (circle, plane, and oblong hole) and features (size, circularity, and flatness) were evaluated. The achieved experiences demonstrated that there exists a different variation between the three CAI software programs. Using the example of a LS

TABLE 2: Results of inspection circle #1 ($\varnothing 20$ mm).

Algorithm	δ_x	δ_y	\varnothing	δ_{\varnothing}	Circularity
#1					
Best-fit	0.060	-0.050	19.877	-0.123	0.157
Min	0.056	-0.043	19.755	-0.245	0.157
Max	0.058	-0.088	20.054	0.054	0.157
#2					
Least square	0.058	-0.047	19.877	-0.123	0.166
Max inscribed	0.055	-0.039	19.755	-0.245	0.168
Min circumscribed	0.056	-0.085	20.054	0.054	0.184
Minimax	0.066	-0.054	19.903	-0.097	0.157
#3					
Best-fit	0.0629	-0.0354	19.877	-0.123	0.047
Min	0.0705	-0.0433	19.745	-0.255	0.047
Max	0.0705	-0.0433	20.059	0.059	0.047

FIGURE 4: Results of the circle #2 analysis. (a) Circularity tolerance. (b) Feature of size deviation δ_{\varnothing} .FIGURE 5: Results for circle #1. (a) Circularity tolerance. (b) Feature of size deviation δ_{\varnothing} .

algorithm, the deterministic algorithm and the analytical solution are well known. The aforementioned results demonstrate that there exists a small (and great) variation between CAI software programs. These deviations can be due to filtering and smoothing operations, a reduction in the

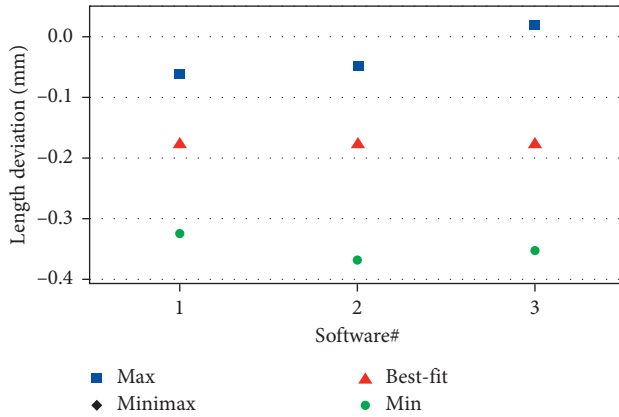
point density, the treatment of outliers, a calibration of the point cloud to the CAD model, and optimization operations when the tolerances allow 10 for degrees of freedom. All these variation sources have an effect on variations that we call “algorithmic errors.”

TABLE 3: Results of inspection circle #2 ($\varnothing 20$ mm).

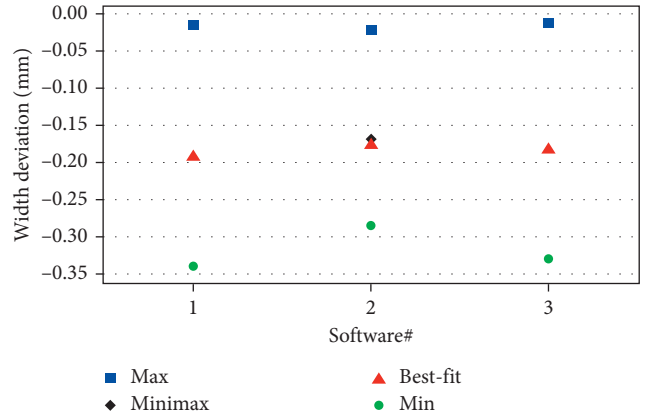
Algorithm	δ_x	δ_y	\varnothing	δ_{\varnothing}	Circularity
#1					
Best-fit	0.049	-0.046	19.874	-0.126	0.186
Min	0.046	-0.053	19.760	-0.240	0.186
Max	0.046	-0.010	20.091	0.091	0.186
#2					
Least square	0.048	-0.043	19.874	-0.126	0.205
Max inscribed	0.046	-0.049	19.760	-0.240	0.208
Min circumscribed	0.045	-0.007	20.091	0.091	0.195
Minimax	0.058	-0.007	19.907	-0.093	0.185
#3					
Best-fit	0.0635	-0.0312	19.873	-0.126	0.034
Min	0.0725	0.0049	19.721	-0.279	0.034
Max	0.0725	0.0049	20.093	0.093	0.034

TABLE 4: Results for the inspection of the oblong hole (width = 13.333 mm and length = 22.666 mm).

Software	Algorithm	δ_x	β	δ_{β}	γ	δ_{γ}
#1	Best-fit	-0.143	22.487	-0.179	13.139	-0.194
	Min	-0.145	22.345	-0.321	12.989	-0.344
	Max	-0.134	22.605	-0.061	13.320	-0.013
#2	Least square	-0.143	22.478	-0.188	13.152	-0.180
	Max inscribed	-0.145	22.306	-0.36	13.048	-0.284
	Min circumscribed	-0.138	22.607	-0.059	13.316	-0.016
	Minimax	-0.144	22.479	-0.187	13.164	-0.168
#3	Best-fit	-0.132	22.478	-0.187	13.151	-0.182
	Min	-0.132	22.321	-0.345	12.994	-0.339
	Max	-0.132	22.646	0.020	13.320	-0.013



(a)



(b)

FIGURE 6: Results for the oblong hole. (a) Length deviation. (b) Width deviation.

TABLE 5: Results for the inspection of Plane A.

Software	Algorithm	Flatness
#1	Best-fit	0.107
	Min	0.107
	Max	0.107
#2	Least square	0.119
	Best-fit	0.051
#3	Min	0.051
	Max	0.051

6. Conclusion

The adjustment of measured point cloud is an essential step in measuring machine software. A correct application of adjustment algorithms can be employed to a comparative tool not just a control tool. Some fitting algorithms optimize the conformance of geometric tolerances and can be applied in the manufacturing analysis. It is important that the correct algorithm be applied in the specific goal; otherwise, an error will occur and the results will not be optimal. An

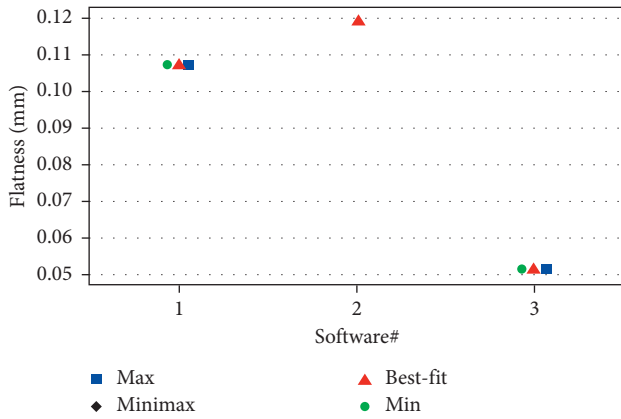


FIGURE 7: Results of flatness analysis (Plane A).

TABLE 6: Results of inspection of Plane D.

Software	Algorithm	Flatness
#1	Best-fit	0.070
	Min	0.070
	Max	0.070
#2	Best-fit	0.031
#3	Min	0.035
	Max	0.035
	Least square	0.073

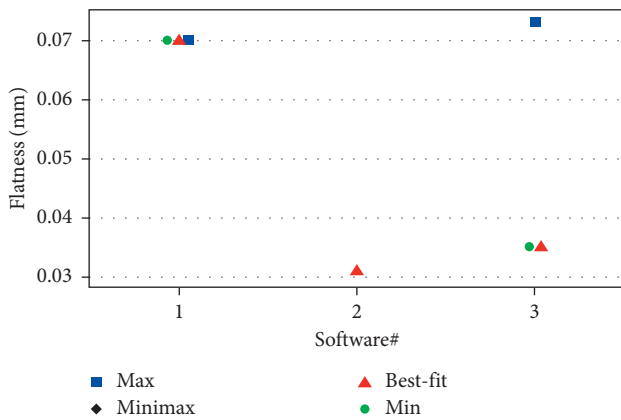


FIGURE 8: Result of flatness analysis in Plane D.

experimental investigation among three inspection software programs to evaluate algorithmic errors was proposed by the same operator, the same CAD, and the point cloud. This experience shows that there are indeed variations (large and small) among the three software programs. The proposed methodology is set to be an important tool to help industrial designers and inspectors select the appropriate algorithm for size and form evaluation. As such, it is the responsibility of the designer and inspector to carefully select the corresponding algorithm in order to avoid errors. Therefore, each specification method has its own field application (functional requirement). Once the specification method for a special application has been assigned, it becomes important to select the most optimal specifications. The benchmark and

experimental case studies demonstrate the influence of the algorithm choice.

Data Availability

The data are given by Creaform, an industry of measure.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to thank the Institut Supérieur d'Informatique et des Techniques de Communication Hammam (Sousse, Tunisia) and the École de Technologie Supérieure (Montréal, Canada) for their support.

References

- [1] E. P. Morse and V. Srinivasan, "Size tolerancing revisited: a basic notion and its evolution in standards," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 227, no. 5, pp. 662–671, 2013.
- [2] E. P. Morse, Y. Peng, V. Srinivasan, and C. Shakarji, "Metrology challenges introduced by new tolerancing standards," in *Proceedings of the 11th International Symposium of Measurement Technology and Intelligent Instruments*, Aachen, Germany, July 2013.
- [3] I. Jbira, M. Tlija, B. Louhichi, and A. Tahan, "CAD/Tolerancing integration: mechanical assembly with form defects," *Advances in Engineering Software*, vol. 114, pp. 312–324, 2017.
- [4] ISO 14405-1, *Geometrical Product Specifications (GPS) Dimensional Tolerancing, Part 1: Linear Sizes*, International Organization for Standardization, Geneva, Switzerland, 2016.
- [5] I. Jbira, A. Tahan, M. A. Mahjoub, and B. Louhichi, "Evaluation of the algorithmic error of new specification tools for an ISO 14405-1: 2016 size," in *Proceedings of the ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Quebec City, Canada, August 2018.
- [6] E. S. Gadelmawla, "Simple and efficient algorithms for roundness evaluation from the coordinate measurement data," *Measurement*, vol. 43, no. 2, pp. 223–235, 2010.
- [7] H. Noura and P. Bourdet, "Evaluation of roundness error using a new method based on a small displacement screw," *Measurement Science and Technology*, vol. 25, no. 4, Article ID 044012, 2014.
- [8] M. S. B. Gapinski, M. Grzelka, and M. Rucki, "The accuracy analysis of the roundness measurement with coordinate measuring machines," in *Proceedings of the 18th Imeko World Congress Metrology for a Sustainable Development*, Rio de Janeiro, Brazil, September 2006.
- [9] W. Sui and D. Zhang, "Four methods for roundness evaluation," *Physics Procedia*, vol. 24, no. 1, pp. 2159–2164, 2012.
- [10] A. Mohamed, A. H. Esa, and M. A. Ayub, "Non-contact approach to roundness measurement," in *Proceedings of the IEEE 7th International Colloquium on Signal Processing and Its Applications*, IEEE, Penang, Malaysia, March 2011.
- [11] L. Xiuming, Z. Jingcai, and L. Hongqi, "Determination of the minimum zone circle based on the minimum circumscribed circle," *Measurement Science and Technology*, vol. 25, no. 1, Article ID 017002, 2014.

- [12] T. T. Tran, V. T. Cao, and D. Laurendeau, "Extraction of cylinders and estimation of their parameters from point clouds," *Computers & Graphics*, vol. 46, pp. 345–357, 2014.
- [13] V. Srinivasan, C. M. Shakarji, and E. P. Morse, "On the enduring appeal of least squares fitting in computational coordinate metrology," *Journal of Computing and Information Science in Engineering*, vol. 12, no. 1, Article ID 011008, 2012.
- [14] P. T. Rhinithaa, P. Selvakumar, S. Nikhil, A. Vysyaraju, L. K. Deepak, and M. Jose, "Comparative study of roundness evaluation algorithms for coordinate measurement and form data," *Precision Engineering*, vol. 51, pp. 458–467, 2017.
- [15] M. Petřík, J. Ková, P. Kaťuch, L. Bednarčíková, R. Hudák, and J. Živčák, "Roundness: determining the reference circle for MCCI and MICI system," in *Proceedings of the 7th International Conference Measurement 2009*, Smolenice, Slovakia, 2009.
- [16] G. Goch and K. Lübke, "Tschebyscheff approximation for the calculation of maximum inscribed/minimum circumscribed geometry elements and form deviations," *CIRP Annals*, vol. 57, no. 1, pp. 517–520, 2008.
- [17] A. Nurunnabi, Y. Sadahiro, R. Lindenberg, and D. Belton, "Robust cylinder fitting in laser scanning point cloud data," *Measurement*, vol. 138, pp. 632–651, 2019.
- [18] T. Chaperon and F. Goulette, "Extracting cylinders in full 3D data using a random sampling method and the Gaussian image," in *Proceedings of the Vision Modeling and Visualization Conference*, pp. 35–42, Stuttgart, Germany, November 2001.
- [19] A. Nurunnabi, Y. Sadahiro, and R. Lindenberg, "Robust cylinder fitting in three-dimensional point cloud data," *ISPRS—International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-1/W1, pp. 63–70, 2017.
- [20] A. Nurunnabi, Y. Sadahiro, and D. F. Laefer, "Robust statistical approaches for circle fitting in laser scanning three-dimensional point cloud data," *Pattern Recognition*, vol. 81, pp. 417–431, 2018.
- [21] J. Guo and J. Yang, "An iterative procedure for robust circle fitting," *Communications in Statistics—Simulations and Computation*, vol. 48, no. 6, pp. 1872–1879, 2018.
- [22] J. E. Deschaud and F. Goulette, "A fast and accurate plane detection algorithm for large noisy point clouds using filtered normals and voxel growing," in *Proceedings of the International Conference on 3D Data Processing, Visualization, and Transmission*, Paris, France, May 2010.
- [23] M. Saval-Calvo, J. Azorín López, A. Fuster-Guillo, and J. Garcia-Rodriguez, "Three-dimensional planar model estimation using multi-constraint knowledge based on k -means and RANSAC," *Applied Soft Computing*, vol. 35, pp. 572–586, 2015.
- [24] H. L. Nguyen, D. A. Belton, and P. Helmholtz, "Comparative study of automatic plane fitting registration for MLS sparse point clouds with different plane segmentation methods," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. IV-2/W4, pp. 115–122, 2017.
- [25] R. T. Marriott, A. Pashevich, and R. Horaud, "Plane-extraction from depth-data using a gaussian mixture regression model," *Pattern Recognition Letters*, vol. 110, pp. 44–50, 2017.
- [26] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.
- [27] ASME GD&T Y14.5-2009, *Dimensioning and Tolerancing*, American Society of Mechanical Engineers (ASME), New York, NY, USA, 2009.
- [28] G. Moroni, W. P. Syam, and S. Petrò, "Performance improvement for optimization of non-linear geometric fitting problem in manufacturing metrology," *Measurement Science and Technology*, vol. 25, no. 8, Article ID 085008, 2014.
- [29] L. Truong-Hong and D. F. Laefer, "Quantitative evaluation strategies for urban 3D model generation from remote sensing data," *Computers & Graphics*, vol. 49, pp. 82–91, 2015.
- [30] L. Truong-Hong and D. Laefer, "Validating computational models from laser scanning data for historic facades," *Journal of Testing and Evaluation*, vol. 41, no. 3, pp. 481–496, 2013.
- [31] I. Jbira, A. B. Makhlof, B. Louhich, A. Tahan, M. A. Mahjoub, and D. Deneux, "A comparative study of extraction cylinder features in industrial point clouds," in *Proceedings of the 23rd International Conference Information Visualisation (IV)*, IEEE, Paris, France, July 2019.

