



HAL
open science

A coarse to fine 3D acquisition system

V Daval, F Truchetet, Olivier Aubreton

► **To cite this version:**

V Daval, F Truchetet, Olivier Aubreton. A coarse to fine 3D acquisition system. Twelfth International Conference on Quality Control by Artificial Vision 2015, SPIE, Jun 2015, LE CREUSOT, France. hal-01164153

HAL Id: hal-01164153

<https://hal.science/hal-01164153>

Submitted on 16 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A coarse to fine 3D acquisition system

V. Daval and F. Truchetet and O. Aubreton
LE2I, 12 rue de la fonderie, Le Creusot, France

ABSTRACT

The 3D chain (acquisition-processing-compression) is, most of the time, sequenced into several steps. Such approaches result into an one-dense acquisition of 3D points. In large scope of applications, the first processing step consists in simplifying the data. In this paper, we propose a coarse to fine acquisition system which permits to obtain simplified data directly from the acquisition. By calculating some complementary information from 2D images, such as 3D normals, multiple homogeneous regions will be segmented and affected to a given primitive class. Contrary to other studies, the whole process is not based on a mesh. The obtained model is simplified directly from the 2D data acquired by a 3D scanner.

Keywords: 3D Reconstruction, 3D Simplification, 3D Compression, Primitives extraction

1. INTRODUCTION

Currently, 3D data resulting from the acquisition of a 3D scanner will generally be analysed and processed before be exploited. This framework refer usually to the 3D chain as presented in figure 1. This chain can be decomposed into several independent steps: acquisition, computation of additional information, data simplification. Therefore, companies focus on providing commercial scanners with high acquisition resolution in order to provide accurate point cloud. In some industrial cases, this amount of information is required as in reverse engineering. However, in many other applications, computing as many points is rarely necessary. Instead, the resulting amount of data will be very large and more difficult to store and analyse. Therefore, the data will be simplified before being processed, while retaining useful information.

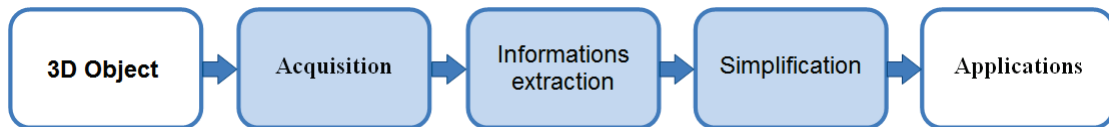


Figure 1. *Classic 3D acquisition-compression-processing chain: from acquisition to compression.*

Figure 1 presents the diagram of the 3D chain encountered in most of the cases. The first step, the acquisition, consists in acquiring a very dense point cloud using a scanner. Then, additional information will be calculated from the point cloud in a second step. Finally, the last step of the 3D chain consists in simplifying the point cloud based on the information calculated in the previous step. This process permits to obtain an amount of data more manageable and easier to analyse. However, this approach leads to the following questions. Knowing the end use of the data, is it necessary to acquire an over-dense amount of data which will have to be simplified afterwards ? Is it not possible to simplify directly the data during the acquisition step ?

In this paper, we propose a methodology to respond positively to this last question. We propose a system to minimize the number of 3D points acquired during the acquisition. This approach provides a simplified point cloud directly to the output of the first step: the acquisition. In this article, we will present the global process of the proposed system (section 2) and the methodology used to identify primitives (section 3). Next, we will present the results obtained with this system on manufactured parts (section 4), then we will conclude in (section 5).

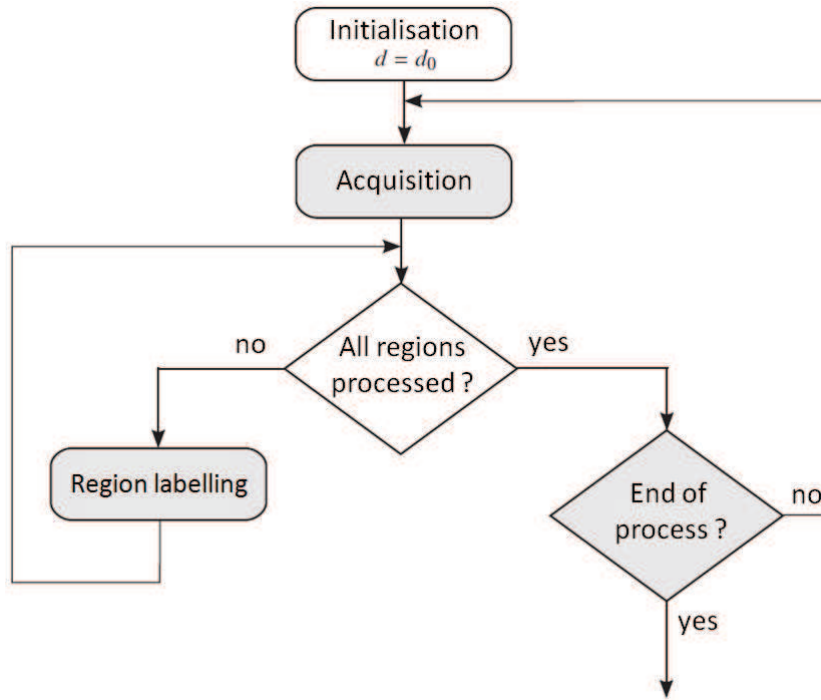


Figure 2. Global flowchart of our system

2. SYSTEM PROPOSED

The method proposed in this paper is based on a coarse to fine approach allowing to simplify data upon acquisition. The principle is to acquire sparse data and then progressively refine the acquisition based on the information extracted from the scanning. The general principle of this approach is synthesized in figure 2.

The objective of this system is to calculate sparse data and then, use this information to identify primitives present in the scene. 3D primitives are simple geometric shapes that can be modelled by a mathematical representation with few parameters. In most industrial applications, manufactured parts are generally composed of planes, cylinders, cones and spheres. Therefore, in our application we have searched to identify these four types of primitive. Because of their simplicity, these forms do not need to be represented by a large number of points. The process proposed is as follow: first, a set of primitives will be identified from a coarse scanning. In a second step, a refinement will be performed on the previously unlabelled regions. Amongst them the works of Yu *et al.* and Bénéière *et al.* can be quoted as being the most significative in the context of our approach¹⁻³.

In order to identify these primitives, the system will compute a sparse point cloud with a density $d = d_0$ from a pair of images obtained by a 3D scanner, more precisely a structured light system. Once the sparse data are obtained, the 3D normals will be calculated from 2D images. The principle of the used method is described by authors⁴ and the application to a structured light system is detailed by authors.⁵ By exploiting these 3D normals, then it is possible to segment the scene in homogeneous regions. The segmentation is based on a exploitation of 3D normals and explained in to details in⁶.

Once the sparse data are segmented, each region will be treated separately in order to identify primitives (region labeling). Then, the regions identified as primitives are stored and labelled. All non-labelled regions will be refined in at each iteration by calculating denser data. The system will continue until a stopping criteria is achieved. This iterative system allows to revisiting the classic 3D chain. In fact, it is possible to obtain simplify data during the acquisition step using a coarse to fine approach based on identifying primitives. In this article, the focus will be on the main part of the system, mainly the identification of primitives (Region labelling). A more complete description of the complete process of this system is described in.⁶

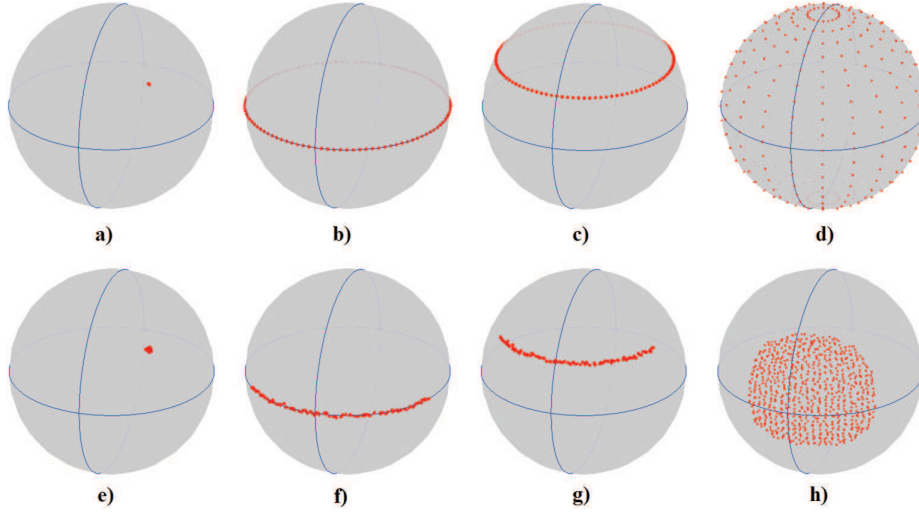


Figure 3. *Gaussian representation of primitives. a) to d) perfect primitives: plan, cylinder, cone, sphere. e) to h) partial and noisy primitives.*

3. PRIMITIVE IDENTIFICATION

In most of the cases, primitives are extracted directly from the 3D data^{7,8}. In^{9,10}, the authors present a method for extracting cylinders and cones from a point cloud using a Gaussian image¹¹. Yu *et al.*¹ propose a similar approach, also based on Gaussian image to identify concave and convex shapes. The Gaussian image is useful for representing the shape of surface. The principle consists in placing the 3D normals of the surface (normalized), to the center of a unit sphere (figure 3). In this section, we propose to generalize this approach for the four types of primitive studied in our application (planes, cylinders, cones, spheres). The Gaussian representation is useful when we consider primitives. Indeed, some primitives have a particular representation that can be exploited to identify and model them. Figure 3 shows that the Gaussian representation is very different between the different types of primitive, even in the case of noisy and incomplete data (figure 3-e to 3-h).

To account for the noisy nature of the data, we propose to use a statistical analysis method, the principal components analysis (PCA), to identify primitives from the Gaussian representation. In the case of the Gaussian image, data are the 3D normals to the surface of the object $n_i = (n_{1i}, n_{2i}, n_{3i})^T$, that can gather in a matrix N . Generally, in the computation of the PCA, data are centred relative to the mean of each component:

$$\bar{n}_1 = \frac{\sum_{i=1}^K n_{1i}}{K}, \quad (1)$$

which gives:

$$N' = \begin{bmatrix} n_{11} - \bar{n}_1 & \dots & n_{1i} - \bar{n}_1 & \dots & n_{1K} - \bar{n}_1 \\ n_{21} - \bar{n}_2 & \dots & n_{2i} - \bar{n}_2 & \dots & n_{2K} - \bar{n}_2 \\ n_{31} - \bar{n}_3 & \dots & n_{3i} - \bar{n}_3 & \dots & n_{3K} - \bar{n}_3 \end{bmatrix}. \quad (2)$$

Calculates the PCA returns to compute the covariance matrix

$$C = N'N'^T, \quad (3)$$

which calculates three eigenvectors X and three eigenvalues λ . Due to the great difference between the primitives, we propose to use these eigenvectors and eigenvalues in order to identify and model the four types of primitives treated in our case.

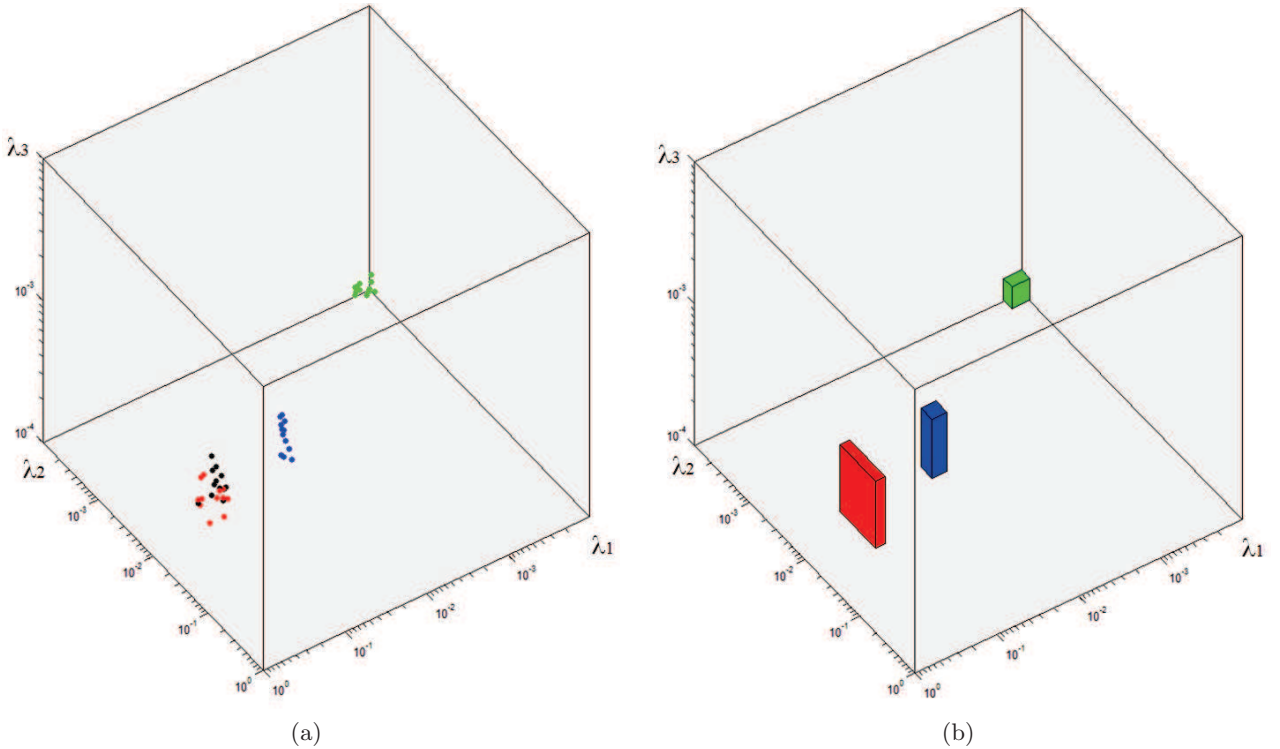


Figure 4. *Three-dimensional representation of eigenvalues. a) eigenvalues for each type of primitives (red: cylinders, black: cones, blue: spheres, green: planes. b) classes corresponding to a type of primitives.*

To classify the different primitives by exploiting the eigenvalues of the covariance matrix C , we have modelled different geometric primitives (planes, spheres, cylinders and cones) with different parameters (radius, orientation, position). For each primitive modelled, we have computed the 3D normals and add a noise. This noise have similar properties than the noise of 3D scanner used. Thus, we obtain a Gaussian representation for each primitive modelled and therefore it is possible to calculate the covariance matrix C for each primitive. Figure 4-a represents the eigenvalues obtained for each primitive in a three-dimensional space. The three axis of this space are λ_1 , λ_2 and λ_3 (the three eigenvalues obtained for each primitive). Red points represent the eigenvalues of the cylinders, green points represent the plans, blue points represent the spheres and black points represent the cones. From this representation, we can define three classes (figure 4-b) by exploiting the eigenvalues of the C matrix. The red class corresponds to the cylinders/cones, blue class contains the spheres and the green class corresponding to the plans. We exploit this 3D representation to identify if a segmented region correspond to a certain primitive. Thus, we compute the covariance matrix C from the Gaussian representation of 3D normals for each segmented region. After, we compute the eigenvalues from C , and by exploiting the 3D representation of these eigenvalues (figure 4), we can identify if the regions correspond to a primitive or not. If a region is identified as belonging to a primitive type, it will be stored and will not be refined by the system. Otherwise, this region will be refined in the next iteration with a scanning with a higher density d .

4. RESULTS

This section presents the results obtained with our system on an industrial object (figure 5-a). Figure 5 presents the results obtained during the different steps of the presented system in this paper. The first step of the system is the acquisition. This step consists in computing sparse data and to segment the object in to many homogeneous regions. Figure 5-b represents the sparse data obtained with a density $d = d_0$ and figure 5-c the different regions segmented using the approach described in section (2) of this article.

The next step of the system corresponds to the identification of the primitives. To identify the primitives, we have propose a method based on the exploitation of eigenvalues of the Gaussian representation (section 3).

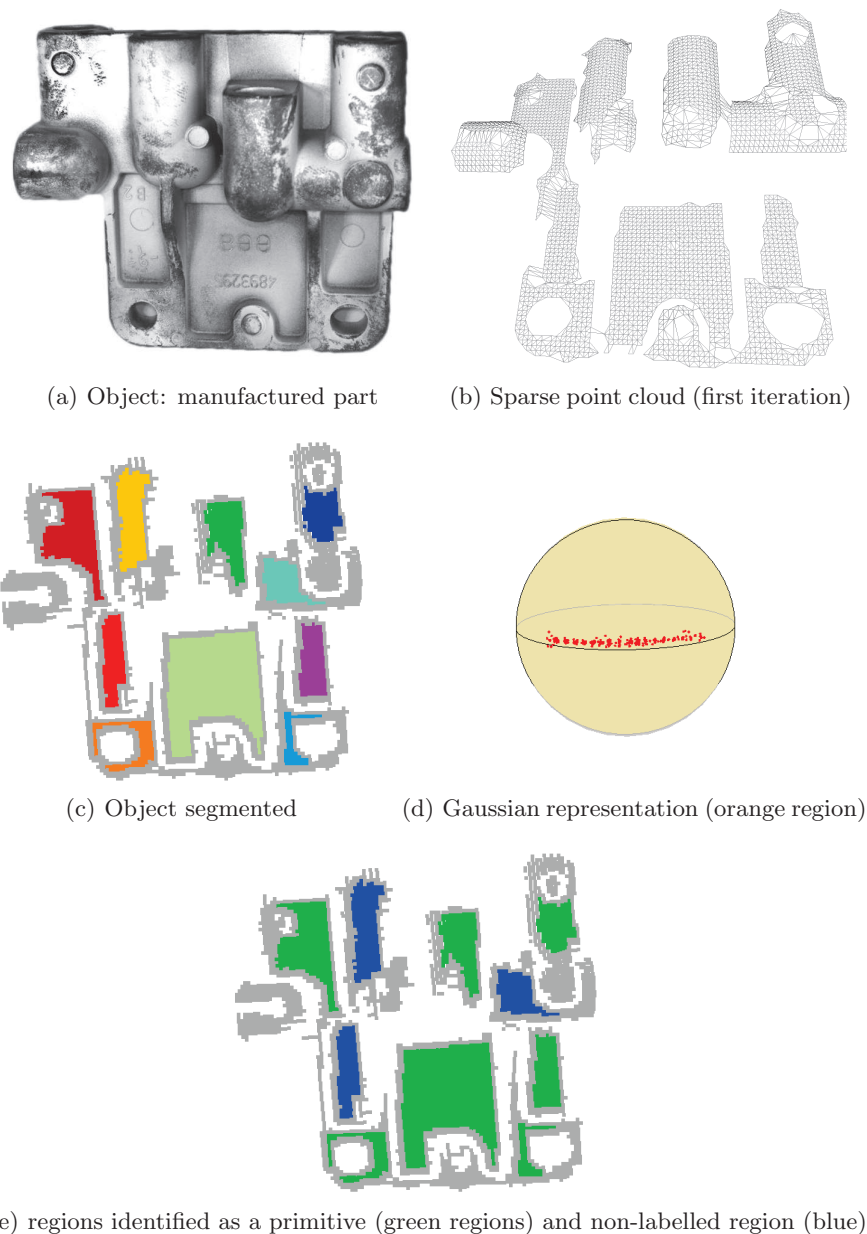


Figure 5. Results obtain by our system step by step.

Figure 5-d shows the Gaussian representation of one segmented region. For this example, the eigenvalues of the covariance matrix C correspond to a cylinder. Therefore, this region will be labelled. Figure 5-e present the result obtained for this object after the first iteration. Green parts correspond to objects identify as primitives, and the blue parts represent the non-labelled regions. During the second iteration of the system, only the blue regions will be refined. Finally, figure 6 depicts the final result obtained with the following stopping criteria:

- The system reaches the maximum density scanner $d = d_{max}$
- All regions are labelled.

Compared to the conventional scanning which provides 38328 points with this manufactured object, our approach provides a point cloud composed of 19141 points (compression rate of 49.94%). To validate our

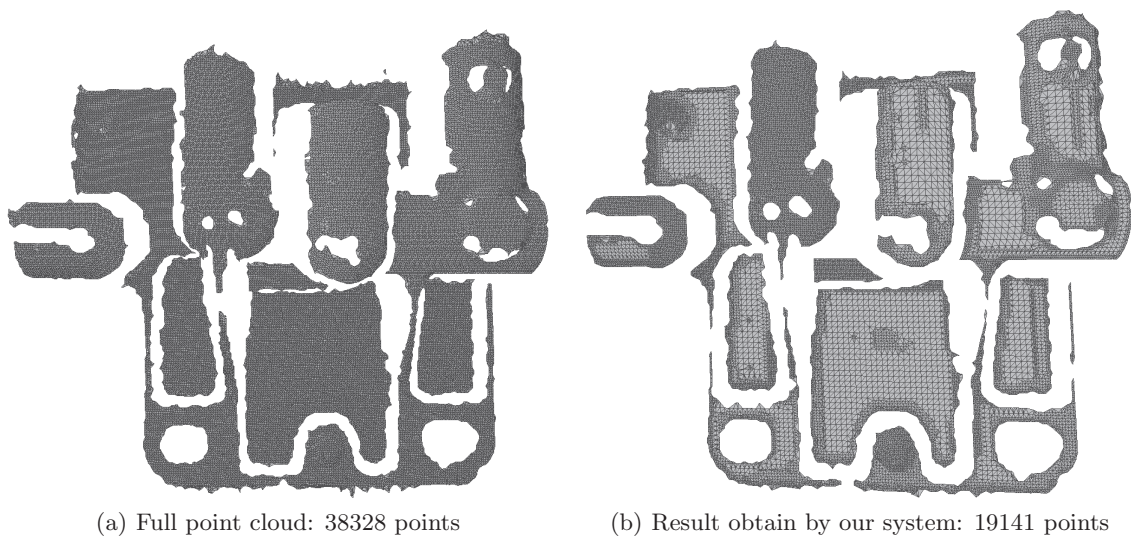


Figure 6. *Final result of the coarse to fine acquisition system.*

method, we also compared the results obtained with traditional methods, through all stages of the 3D chain (dense acquisition, information retrieval, simplification). Two methods have been chosen to be compared with our work: ACVD¹² and Qslim¹³ (figure 7), which are well-known and have exploitable toolbox. In this comparison we get similar results with both methods but the simplified point cloud is obtained directly from the acquisition, directly from the output of the scanner.

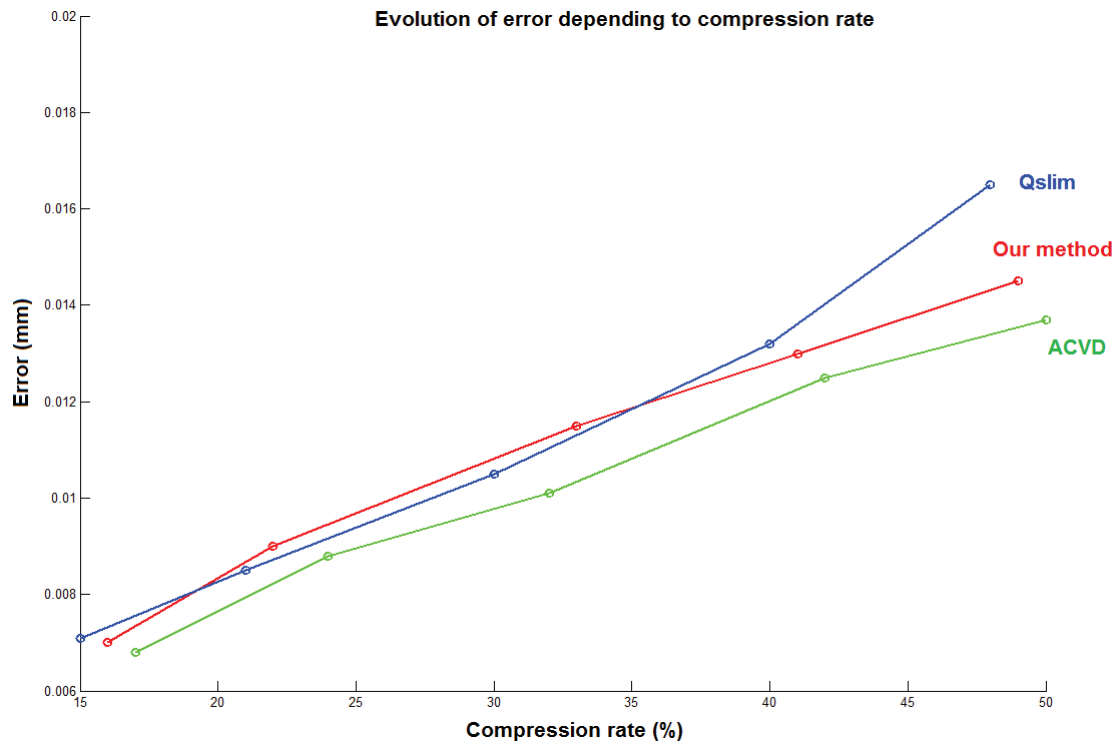


Figure 7. *Comparison of our method with ACVD¹² and Qslim.¹³*

5. CONCLUSION

In this paper, we present a coarse to fine acquisition system permitting to revisit the classic 3D chain. This system computes sparse data and refines the acquisition progressively by respecting the surface geometry. The goal of our system is to identify if the primitives present in a point cloud can be used for point cloud compression. In fact, a large set of points can be replaced by a small number of parameters.

However, this system is totally dependent of the numbers of primitives of the object itself. For these reasons, this system is only useful for mechanical parts. An improvement of this system, could be the extension to other forms using more sophisticated primitives (nurbs, superquadrics, ...).

REFERENCES

1. L. Yu and X. Youlun, "Automatic segmentation of unorganized noisy point clouds based on the gaussian map," *Computer-Aided Design* **40**, pp. 576 – 594, 2008.
2. R. Bénéière, G. Subsol, G. G., G. Le Breton, and P. W., "A comprehensive process of reverse engineering from 3d meshes to cad models," *Computer Aided Geometric Design* **45**, pp. 1382–1393, November 2013.
3. H. Song and H. Feng, "A progressive point cloud simplification algorithm with preserved sharp edge data," *The International Journal of Advanced Manufacturing Technology* **45**, pp. 583–592, 2009.
4. Z. Song and C. Chung, "Determining both surface position and orientation in structured-light-based sensing," *Pattern Analysis and Machine Intelligence* **32**, pp. 1770–1780, 2010.
5. V. Daval, F. Truchetet, and O. Aubreton, "Primitives extraction based on structured-light images," *QCAV 2013 - 11th International Conference on Quality Control by Artificial Vision*, p. 45, 2013.
6. V. Daval, F. Truchetet, and O. Aubreton, "Coarse to fine: toward an intelligent 3d acquisition system," *SPIE Electronic Imaging 2015*, 2015.
7. C. Beder and W. Forstner, "Direct solutions for computing cylinders from minimal sets of 3d points," *Computer Vision*, pp. 135–146, 2006.
8. R. Schnabel, R. Wahl, and R. Klein, "Efficient ransac for point-cloud shape detection," *Computer Graphics* **26**, pp. 214–226, 2007.
9. R. Bénéière, G. Subsol, G. G., G. Le Breton, and P. W., "Recovering primitives in 3d cad meshes," *Proc. SPIE* **7864**, pp. 78640R–78640R–9, 2011.
10. T. Chaperon and F. Goulette, "Extracting cylinders in full 3d data using a random sampling method and the gaussian image," *Vision Modeling and Visualization Conference*, pp. 35–42, 2001.
11. K. Gauss, "General investigations curved surfaces surfaces," *English translation*, 1965.
12. S. Valette, J.-M. Chassery, and R. Prost, "Generic remeshing of 3d triangular meshes with metric-dependent discrete voronoi diagrams," *Visualization and Computer Graphics* **14**, pp. 369–381, March 2008.
13. M. Garland and P. S. Heckbert, "Surface simplification using quadric error metrics," pp. 209–216, 1997.