A NON CONTACT CAD BASED INSPECTION SYSTEM

F. PRIETO[£§], T. REDARCE[£], R. LEPAGE[§] AND P. BOULANGER[fl]

[£] INSA, Laboratoire d'Automatique Industrielle, 20 avenue Albert Einstein,
69621 Villeurbanne Cedex, France, -@lai.insa-lyon.fr
[§] ÉTS, Laboratoire d'Imagerie, de Vision et d'Intelligence Artificielle, 1100 rue Notre-Dame Ouest,
Montréal, Québec, H3C 1K3, Canada, lepage@livia.etsmtl.ca
[fl] CNRC, Institut des Technologies de l'Information, Chemin Montréal, édifice M50,
Ottawa, K1A 0R6, Canada, boulanger@iit.nrc.ca

Abstract - We submit an automatic inspection system of industrial parts. Inputs to the system are: an unordered cloud of 3D points of the part and its CAD model in IGES and STL formats. The 3D cloud is obtained from a high resolution 3D range sensor. After registration between the cloud of points and the STL CAD model, the cloud is segmented by computing minimal distance and comparing some local geometric properties between the 3D points and the NURBS surfaces. Segmentation results are used to inspect the surfaces of interest and generate a visual and hardcopy report. Visually, the level of discrepancy between the measured points and the CAD model is highlighted using a color map. The hardcopy report is a statistical analysis of the surface.

1. INTRODUCTION

A. Statement of the problem

The increasing number of manufactured objects showing complex surfaces, either for functional reasons or by design, and technological improvement in manufacturing create a need for automatic inspection of complex parts. This type of apparatus requires a very accurate geometrical definition of the inspected object, an accurate data acquisition system, and clearly defined rules for the inspection of these surfaces. Use of three-dimensional coordinate measuring machines and recent advent of laser sensors combining measurement accuracy and fast acquisition allow obtaining a great number of 3D measurement point. These accurate 3D points allow an explicit description of object surfaces.

Inspection is the process of determining if a product (part or object) deviates from a given set of specifications (tolerances). Coordinate Measuring Machine (CMM) is the industry standard mechanism for part validation, but in spite of its high precision, it has some important limitations such as: the need for mechanical fixturing, low measurement speed and the need to be programmed as new part is inspected. On the other hand, recent advances in non-contact sensor like laser range finder, with significant improvement in speed (about 20 000 points/s) and range precision (about 25 micron), allow them to be used in inspection tasks.

It is most useful to use CAD models in inspection because the model contains an exact specification of an industrial part and they provide a well-defined model for inspection. CAD models provide a mathematical description of the shape of an object, including an explicit parametrization of surface shape and an explicit encoding of inter-surfaces relationships. The database can also be augmented with manufacturing information including geometric tolerance, quality of surface finish, and manufacturing information. An advantage of using CAD representations for inspection is their high flexibility; it is most easy to add a new object to the inspection system even before it is manufactured.

In this paper, we submit an inspection system that uses as input an unordered cloud of 3D points of the part and its CAD model in IGES and STL format. The cloud of 3D samples is digitized by a 3D laser range sensor with a resolution of 25 micron and a depth field of 10 cm. The system registers the cloud of 3D points and the STL CAD model of the part. Most manufactured parts have to be checked using specifications on defined surfaces, so in order to be able to inspect the surfaces of interest, the cloud of points is registered with the IGES CAD model and segmented as many time as the number of surfaces in the part.

In inspection process, each surface of interest is checked against corresponding segmented 3D points. The system then outputs a visual and an hardcopy report.

B. Related Literature

The automatic verification of manufactured object is a fairly recent concern. The main reason is that to carry out this type of task, it is necessary to have contactless sensors. The digitalization of the images from video camera and later from CCD camera made it possible to obtain informations on objects at high speed. Quickly one attains the limits of these sensors for the analysis of 3D parts, at least in industry, because of their limited precision and the difficulty to rebuild up the third dimension. The appearance of sensors combining a laser beam and a CCD camera allows the rebuilding of the third dimension, without however giving the accuracy obtained with a 3D coordinate measuring machine. The laser telemeter sensor made it possible and attains desired speed and precision. It is possible therefore to automate the inspection process. At present, few papers look at the use of depth image for inspection. One reason is the lack, up to now, of powerful systems for to the recovery of depth images.

Relating to the inspection process we can quote the article of T.S. Newman and Jain [1], a survey of the question, where the problem is tackled from the point of view of illu-

minance images (gray-level or binary), range images or other sensing modalities. They discuss general benefits and feasibility of automated visual inspection, and present common approach to visual inspection and also consider the specification and analysis of dimensional tolerances and their influence on the inspection task.
The system developed by Newman and Jain [2] permits the detection of defects in range images of castings. This system uses CAD model data for surface classification and inspection. The authors report several advantages for the use of range images in inspection, namely that they are insensitive to ambient light, that the objects can usually be extracted from their background more easily, that depth measurement is accurate, and most important, that range image are explicitly related to surface information. The authors show the interest with the use of the CAD database in order to carry out the task of control. Moreover they show the weakness of the current CAD systems to make automatic check. The authors do not speak about tolerances measurements.
In [3], Tarbox and Gottschlich report a method based on comparing a volumetric model of reference object to a volumetric model of an actual object iteratively created from sensor data. To provide a framework for the evaluation of volumetric inspection, they have developed a system called IVIS (Integrated Volumetric inspection System). They obtain a volumetric image of the defects by using custom comparison operators between the reference model and the model of the analyzed part.
In [4], Marshall and all report a vision system which outputs dense depth data of a part, and the processing algorithm of an array of these data in order to achieve the goal of automatic inspection of mechanical parts by developing general model-based inspection strategies that can be applied to a large set of objects.
We have worked for several years on measurement verification, comparing CAD models and range images [5], [6]. The size of the defects which can be detected depends in a very significant way on the resolution of the sensor. Therefore we experimented with good 3D-measurement sensor [7] having capability to detect as small defects as 0.05 mm of depth.

2. TOLERANCE DEFINITION AND MEASUREMENT

Manufacturing methods are unable to produce parts with perfect shapes, sizes and forms. Therefore, given a physical instance of a part, it is possible to measure dimensional or geometric properties, and determine if deviations are within bounds defined by the tolerance specification. In this section, we will define a few of the tolerance types and give explanation of the algorithms to make those tolerance measurements on an object feature.
A *datum* is a point, line, plane, or other geometric surface from which dimensions are measured when so specified or to which geometric tolerances are referenced. A datum has an exact form and represents an accurate or fixed location, for purposes of manufacturing or measurement. A *datum feature* is a feature of a part, such as an edge or a surface, which forms the basis for a datum or is used to establish its location. We will use datums as obtained from the CAD model (IGES format) of the part. In the IGES CAD model, all the surfaces of the part are defined exactly using a parametric NURBS (Non-Uniform Rational B-Splines) surfaces. As a datum feature, we use the segmented set of 3D measured points related to the datum.
The segmentation process will be detailed later.

Flatness tolerance. A flatness tolerance means that all 3D points on the surface should be contained within a tolerance zone defined as the space between two parallel planes separated by the specified tolerance. These planes may be oriented in any direction to enclose the surface, that is, they are not necessarily parallel to the plane base.

Flatness measurement. From the definition we know that flatness tolerance is not related to any datum. So, we check for flatness of a surface using only the 3D points associated to this surface and obtained from the segmentation process. The 3D points segment is registered with the CAD model, and the perpendicular distance between each point on the segment and the NURBS surface is computed. Because the registration process minimizes the sum of squared distances between the points in the segment and the CAD model, the set of distances will have a gaussian distribution (see Figure 5). We define a tolerance zone by defining two parallels planes to the NURBS surface, at a distance of $\pm 2\sigma$ from the mean distance. If the distance between these two parallel planes (4σ) is less than the specified tolerance, the surface conforms to the flatness specification.

Parallelism tolerance. A parallelism tolerance specifies a tolerance zone defined by two parallel planes or lines parallel to a datum plane or axis within which the surface or axis of the feature must lie.

Parallelism measurement. Because parallelism tolerance is related to a datum plane, we check this tolerance by using two segmented point sets, the first one called datum surface (S_D) and associated to the datum and the other one associated to the surface to inspect (S_I). S_D is registered with the CAD model datum and the rigid transformation from registration is applied to S_I. We compute the perpendicular distance between each point on (S_D) and the NURBS surface, and define a tolerance zone by putting two parallels planes to the CAD model datum (a NURBS surface), at a distance of $\pm 2\sigma$ from the mean distance. If the (4σ) distance is less than the specified tolerance, the surface conforms to the parallelism specification.

In a similar way, we can check for other geometric tolerances like: perpendicularity tolerance, angularity tolerance, cylindricity tolerance and profile tolerance of any surface.

Figure 1 shows the tolerance specifications (in mm) of a

part. As this information is not present in the CAD model, we have to add it in an independent file included in the database.

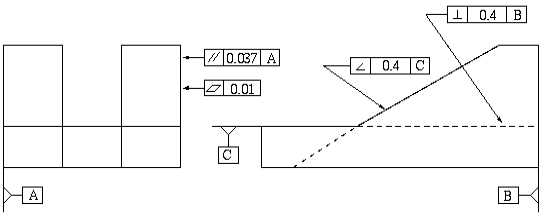

Figure 1. Tolerance specifications

3. THE REGISTRATION METHOD

The registration of two shapes is defined as finding the 3D rigid transformation (rotation + translation) to be applied over one of the shape to bring it with the other one, into one common cartesian coordinate system. The registration process in this paper relies on the well-known work of Besl and McKay [8] who in 1992 developed a general-purpose representation method for the accurate and computationally efficient registration of 3D shapes, including free-form curves and surfaces.
The method is based on the Iterative Closest Point (ICP) algorithm, which requires only to find the closest point from a geometric entity to a given point. The rigid transformation is computed using a unit quaternion. But as the transformation estimation is done by a Mean Square (MS) distance computation, this method is not robust to outliners points, obtained either by noise or by the presence of other parts in the scene.
As a solution to this problem, Masuda and Yokoya [9] estimate the rigid motion between two range images in a robust way by fusing the ICP algorithm with random sampling and Least Median of Squares (LMS) estimation. They demonstrated that registration between two images can be achieved with a high level of robustness (up to 50%) to occlusion and noise.
Moron [10] implemented an algorithm for registration between an unordered cloud of 3D points and a CAD model in STL or IGES format. In the registration process, we use the CAD model in STL format rather than IGES, so that few precision is lost but computation time is largely improved.

The registration method can be decomposed into three main steps:
First, the algorithm randomly selects Ns 3D points from the original 3D data set, and then computes a rigid transformation by using an ICP algorithm on the subset. This process is repeated Nt times. For find a solution at this non-linear problem, we take just a sample of Ns points. The probability of finding a solution increases as Ns decreases or Nt increases. After each ICP execution, the quality of the estimated rigid transformation is evaluated by computing the median square error.
Second, the best estimated rigid transformation corresponding to the least median square error is applied over the whole 3D data, and the original 3D data set is segmented into inlier and outlier point sets.
Finally, a standard mean square ICP algorithm is then applied on the inlier set of points to find the optimal rigid transformation solution.
In order to find a global solution, it may be necessary to apply this method several times, with different initial conditions. From now, we only consider the solution corresponding to the best estimation.
The registration error [11] results from thresholding the error distance in order to stop the iterative registration process.

4. 3D DATA SEGMENTATION

In the registration process, we have superposed the CAD model with the 3D data of the part. But because we are interested in inspecting some specific surfaces, we need to segment the piece into its different surfaces.
The segmentation of the 3D cloud is done by computing the distance between every 3D point and all of the surfaces in the CAD model (IGES format), and by comparing some local geometric properties between each 3D point in the cloud and its closest point on the surface.
In the IGES CAD model, all the surfaces of the part are defined as a parametric NURBS surfaces. The problem of computing the distance from a 3D point to a NURBS surface can be formulated as finding a point on the parametric surface such that the distance between the 3D point and the point on the surface is minimal in the perpendicular direction to the tangent plane at the point on the surface.The problem is solved as a minimization problem.
The local geometric properties that we estimate are: the normal surface, the gaussian curvature and the mean curvature. For the point on the parametric surface, those properties are estimated using the surface parameters (NURBS). For the 3D point, we use a parametric second order polynomial computed across a neighborhood of points. A 3D point is labelled with the name (number) of the closest NURBS surface, if the local geometric properties on the 3D point are similar to those on the parametric surface.

A. Point/NURBS surface distance computation

The distance of a point to a NURBS surface can be computed as follow: find a point in the parametric space of the surface (u_0, v_0) such that the distance between the surface $\vec{S}(u_0, v_0)$ and the point $\vec{r}$ is minimum in the perpendicular direction to the tangent plane at the point location (see Figure 2).

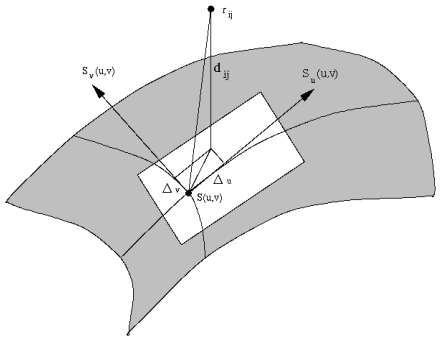

Figure 2. Point / NURBS surface distance

The function to be minimized is the following: $min_{u_0, v_0}\left\|\vec{r} - \vec{S}(u, v)\right\|^2$. If one performs the Taylor expansion of the parametric surface $\vec{S}(u, v)$ we obtain:

$$\vec{s}(u, v) = \vec{S}(u_0, v_0) + \frac{\partial}{\partial u}\vec{S}(u_0 - u) + \frac{\partial}{\partial v}\vec{S}(v_0 - v))$$

Using this expansion, the minimization problem becomes:

$$min_{u_0, v_0}\left\|\vec{r} - \vec{S}(u_0, v_0) + \frac{\partial}{\partial u}\vec{S}(u_0 - u) + \frac{\partial}{\partial v}\vec{S}(v_0 - v)\right\|^2$$

This can be expressed in matrix from as: $min_{u_0, v_0}\left\|J\vec{w} - \vec{d}\right\|^2$

where J is the Jacobien matrix of $\vec{S}(u, v)$ and given by

$$J = \begin{bmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \\ \frac{\partial z}{\partial u} & \frac{\partial z}{\partial v} \end{bmatrix} \quad \text{and} \quad \vec{w} = \begin{bmatrix} u_0 - u \\ v_0 - v \end{bmatrix}$$ is equal to the variation of the parametrization.

If $\vec{d}(u, v)$ is the error vector for the initial parametrization (u_t, v_t), i.e. the initial closest point to the triangulated CAD format, let: $\vec{d}(u, v) = \vec{r} - \vec{S}(u, v)$, then the solution to the minimization problem is equal to: $\vec{w}(J^T J)^{-1} J^T \vec{d}$. Using an iterative procedure, one can compute the distance from the point to the surface in less than four or five iterations.

B. Geometric properties comparison

Let P be a point from the 3D range data, and Q the closest point to P on the surface. To terminate the segmentation process, we estimate and compare some local geometric properties around of P and Q. Q points are estimated by using the NURBS CAD model.

We estimate the local geometric properties of P by using the method proposed by Boulanger [12]. This method is viewpoint invariant because the surface estimation process minimizes the distance between the NURBS surface S and the 3D data point in a direction perpendicular to the tangent plane of the surface at this point. The surface normal $\vec{n}(u, v)$, the gaussian curvature $K(u, v)$ and the mean curvature $H(u, v)$ from the parametric surface $\vec{\eta}(u, v)$ can be estimated by:

$$\vec{n}(u, v) = \frac{\vec{r_u}(u, v) \times \vec{r_v}(u, v)}{\left\|\vec{r_u}(u, v) \times \vec{r_v}(u, v)\right\|}$$

$$K(u, v) = \frac{\left[\vec{r}_{uu} \cdot \vec{r}_u \cdot \vec{r}_v\right]\left[\vec{r}_{vv} \cdot \vec{r}_u \cdot \vec{r}_v\right] - \left[\vec{r}_{uv} \cdot \vec{r}_u \cdot \vec{r}_v\right]^2}{\left|\vec{r}_u \times \vec{r}_v\right|^4}$$

$$H(u, v) = \frac{A + B - 2C}{2D^3} \quad \text{where}$$

$$A = (\vec{r}_u \cdot \vec{r}_v)\left[\vec{r}_{uu}\vec{r}_u\vec{r}_v\right], \quad B = (\vec{r}_u \cdot \vec{r}_v)\left[\vec{r}_{vv}\vec{r}_u\vec{r}_v\right]$$

$$C = (\vec{r}_u \cdot \vec{r}_v)\left[\vec{r}_{uv}\vec{r}_u\vec{r}_v\right], \quad D = \left|\vec{r}_u \times \vec{r}_v\right|$$

$$\vec{r}_u = \frac{\partial}{\partial u}\vec{r}, \ \vec{r}_v = \frac{\partial}{\partial v}\vec{r}, \ \vec{r}_{uu} = \frac{\partial^2}{\partial u^2}\vec{r}, \ \vec{r}_{vv} = \frac{\partial^2}{\partial v^2}\vec{r} \text{ and}$$

$$\vec{r}_{uv} = \frac{\partial^2}{\partial u \partial v}\vec{r}.$$ We need to estimate the first and second partial derivatives at the point P, by using a parametric second order polynomial. It is obtained by using a N x N neighborhood, where $\vec{r}(u, v) = (x(u, v), y(u, v), z(u, v))^T$ is the measured point from the range sensor.

Let $\vec{\eta}(u, v) = \sum_{i=0}^{2}\sum_{j=0}^{2} \vec{a}_{ij} u^i v^j = (h_x(u, v), h_y(u, v), h_z(u, v))^T$

where $\vec{a}_{ij}$ is the coefficient for each component of $\vec{\eta}(u, v)$ and equal to zero if $i+j > 2$. Using this polynomial the partial derivatives at the point P are:

$\vec{\eta}_u = \overrightarrow{a_{10}} + 2\overrightarrow{a_{20}}u_0 + \overrightarrow{a_{11}}v_0$, $\vec{\eta}_v = \overrightarrow{a_{01}} + \overrightarrow{a_{11}}u_0 + \overrightarrow{2a_{02}}v_0$, and

$\vec{\eta}_{uu} = 2\overrightarrow{a_{20}}$, $\vec{\eta}_{vv} = 2\overrightarrow{a_{02}}$, $\vec{\eta}_{uv} = \overrightarrow{a_{11}}$ where (u_0, v_0) are the parametric coordinates in the center of the neighborhood. These parameters are found by using the least-square method.

Finally, we compare the local geometric properties of Q estimated from the NURBS surface to P from the 3D range data. Let α_{tol} be the permissible angle between the surface normal $\overrightarrow{N_S}$ and 3D data normal $\overrightarrow{N_r}$ at point P. Then the condition $\left|Angle(\overrightarrow{N_S}, \overrightarrow{N_r})\right| \le \alpha_{tol}$ has to be respected. Let K_{tol} and H_{tol} be the defined variation of the gaussian and the mean curvatures, then the conditions: $K_S - K_r| \le K_{tol}$, and $|H_S - H_r| \le H_{tol}$ have to be respected.

5. INSPECTION RESULTS

We have used a Biris[1] sensor to digitize the mechanical parts. The result of this digitalization is an unordered set of 3D points describing the scanned object (see figure 3).

1. Sensor Biris is manufactured by the Vitana company (Ottawa, Ontario, Canada) under CNRC license.

Figure 3. 3D points cloud resulting from the digitalization of a mechanical piece

Our goal is to check the cloud of 3D points against the CAD model of the part. Registration of the cloud with the CAD model is the first step illustrated in figure 4.

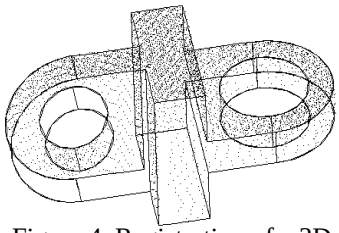

Figure 4. Registration of a 3D cloud and its CAD model

After registration, in order to be able to check for geometric tolerance, the cloud of points is segmented as many times as the number of surfaces in the part. Figures 6 and 7 shown, some segmented surfaces.

In section 2 we have described the algorithms used for checking the geometric tolerance. Visually, the level of discrepancy between the measured points and the CAD model is shown using a color map. We checked against the tolerance specifications of the part given in Figure 1.
Figure 5 shows the distribution of the 3D point to CAD model distance, for the surface segment with a flatness tolerance. From this figure, a gaussian distribution can be approximated.

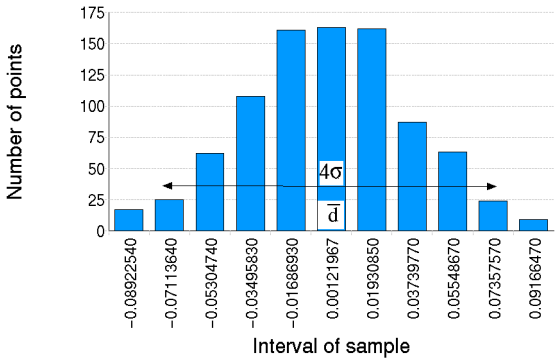

Figure 5. Distribution of 3D points to CAD model distance

Rigorously, to measure the flatness of the surface we would place the parallel planes to the NURBS surface at the distances: *max* and *min* from $\bar{d}$ (see figure 5).

But we will locate them at a distance of $\pm 2\sigma$, guarantying that the 95% of measured points are between the two parallel planes. We pose that the 5% of remaining points are noisy.

We computed the mean distance and the standard deviation of Figure 5, as: $\bar{d}$ = -0.000455264 mm and σ = 0.0365971 mm. The distance 4σ is bigger than the specified tolerance (0.01mm) so, *a priori* the surface is not conforms to the flatness specification.

Figure 6 shows a datum surface and a surface with a perpendicularity tolerance specification.

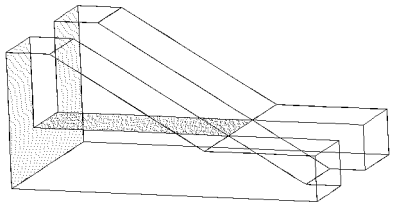

Figure 6. 3D points of a two perpendicular surfaces

We have computed the mean distance and the standard deviation as: $\bar{d}$ = -0.00242857 mm and σ = 0.0436967 mm. For this surface, 4σ = 0.1747868 mm is less than the specified tolerance (0.4mm), so we can say that the surface true to the perpendicularity specification.

Figure 7 shows a datum surface and a surface with an angularity tolerance specification, we have compute σ = 0.0310764 mm.

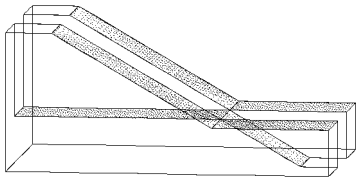

Figure 7. 3D points of two inclined surfaces

So 4σ = 0.1243056 is less than (0.4 mm) and the surface conforms to the angularity specification.

In the figure 8, we show a visual inspection of a hole (parts viewed in the figure 4). It has a tolerance cylindricity specification of 0.0165 mm, and we computed 4σ = 0.118914 mm.
A problem of this kind of sensor appears when we want to measure an inner surface. It is then more difficult to have a large number of points. That is due to occlusion problems or to the high incidence angle between the beam and the surface. Actually, accuracy in the 3D points is dependent on viewpoint at the time of digitalization.

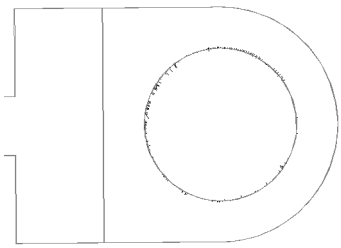

Figure 8. Visual result for cylindricity checking

During the digitalization process, some noise is added to the measured points as a function of the laser camera position. Because we didn't take the noise value into account in tolerance conformity computations, an out-of-tolerance result cannot guarantee a lack of conformity for sure. We are presently modelizing the noise formation process in order to enhance tolerance conformity computation.

6. CONCLUSION

We have submitted an inspection system for manufactured parts. The system first registers a cloud of 3D points with a STL CAD model of the part, and then segments the 3D points in different surfaces by using the IGES CAD model. Results of inspection are available in two ways: visually, using a color map to display the level of discrepancy between the measured points and the CAD model, and a hardcopy report with a statistical description of the surface.

The segmentation process is not dependent on the part geometry. It depends basically on the 3D points precision and in a most important way on the density of points on a segmented surface, in order to obtain a good estimate of the local geometric properties.

The precision of the inspection results is mainly function of the precision of the 3D points. At present we find some range sensors with a high precision, but in order to approach the precision of a Coordinate Measuring Machine a lot of work in the digitalization process has to be done.

We have defined and implemented a methodology to check for geometric tolerances, using a cloud of 3D points and a CAD model of the part. From the CAD model we know the exact description of the part, so we can use this information to make the inspection of parts with complex surfaces.

7. REFERENCES

[1] Newman, T.S., and Jain, A.K. "A survey of automated visual inspection", *Computer Vision and Image Understanding*, 61(2), March 1995, pp. 231-262.

[2] Newman, T.S., and Jain, A.K. "A system for 3D CAD-based inspection using range images", *Pattern Recognition*, vol. 28, No10, 1995, pp. 1555-1574.

[3] Tarbox, G.H., and Gottschlich, S.N. "IVIS: An integrated volumetric inspection system", *Computer Vision and Image Understanding*, 61(3), May 1995, pp. 430-444.

[4] Marshall, A.D., Martin, R.R. and Hutber, D. "Automatic inspection of mechanical parts using geometric models and laser range finder data", *Image and Vision Computing*, 9(6), December 1991, pp. 385-405.

[5] Moron, V., Boulanger, P., Masuda, P., Redarce, H.T. "Automatic inspection of industrial parts using 3-D optical range sensor", *SPIE Proceedings, Videometrics IV*, vol. 2598, 1995, pp. 315-325.

[6] Prieto, F., Redarce, H.T., Lepage, R., and Boulanger, P. "Visual system for the fast and automated inspection of 3D part", *7th European Conference on Rapid Prototyping*, Paris, France, 19-20 November 1998.

[7] Beraldin, J.-A., El-Hakim, S.F., and Blais, F. "Performance evaluation of three active vision systems built at the NRC". *Proceedings of the Conference on Optical 3-D Measurement Techniques*, Vienna, Austria, 1995, October 2-4, pp. 352-362.

[8] Besl, P.J., McKay, N.D. "A method for registration of 3-D shapes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, No 2, 1992, February, pp. 239-256.

[9] Masuda, P., Yokoha M. "A robust method for the registration and segmentation of multiple range images", *Computer Vision and Image Understanding*, vol 61, N^{o} 3, May 1995, pp.295-307.

[10] Moron, V., Boulanger, P., Redarce, H.T. "Mise en correspondance du modèle CAO d'un objet avec son image 3D: Application à l'inspection", *10ème Congrès de Reconnaissance de Formes et Intelligence Artificielle (RFIA)*, Rennes, 16-18 janvier 1996, Vol. 2, pp. 913-922.

[11] Moron, V. "Mise en correspondance de données 3D avec un modèle CAO: Application à l'inspection automatique", Thèse de doctorat, INSA de Lyon décembre 1996.

[12] Boulanger, P., "Extraction multiéchelle d'éléments géométrique", PhD Dissertation: Génie Electrique, Université de Montréal, Canada, 1994.